PNEUMONIA CHEST X-RAY RETRIEVAL SYSTEM

Min Huang, Meiyi Chen, Junyang Feng, Yaxuan Zheng

ABSTRACT

In many hospitals, especially where medical staff are limited, interpreting chest X-rays for pneumonia can be challenging and sometimes inconsistent. To make this process easier, we built a simple system that finds similar X-ray images and gives automatic suggestions based on what it sees. We tried it with both EfficientNet-B3 and a Vision Transformer instead of using just one model. The idea was to capture both the tiny details and the overall shape of each X-ray. After pulling out the features, we normalized them and ran a similarity check using cosine distance. It wasn't perfect, but for most cases, the results looked pretty relevant.Retrieval works well and feels fast in the test. Doctors or users can upload an image through a simple web interface, view prediction results, and examine visual explanations via Grad-CAM. When tested on a public dataset, the system reached 88.3% classification accuracy with 82.7% Top-1 retrieval performance. While still in development, these results show promise for making pneumonia diagnosis more consistent and interpretable in clinical settings.

Index Terms— Pneumonia, EffectiveNet-B3, Vision Transformer, Retrieval system

1. INTRODUCTION

Pneumonia is a common yet potentially severe respiratory disease, particularly affecting children and the elderly [1]. Accurate and timely diagnosis is important for effective treatment and the prevention of complications. Traditionally, pneumonia diagnosis relies on manual interpretation of chest X-ray images by radiologists [2], a process that can be time-consuming and highly dependent on clinical expertise. These challenges are more pronounced in primary care settings or regions with limited medical resources, where delays and diagnostic errors may occur.

To address these limitations, we propose a prototype X-ray image retrieval system for pneumonia that combines automated classification with case-based visual comparison to support clinical workflows. The system is designed to facilitate quicker access to relevant cases and improve consistency in image interpretation.

It consists of several components: image classification, visual similarity retrieval, a web-based user interface, and a result display module. Unlike conventional classification tools, the system enables users to retrieve visually similar chest X-ray images, offering additional context for assessment by referencing comparable cases. Users can upload images and view predicted results alongside related examples through an intuitive interface. Based on real-world chest X-ray datasets, the system serves as an exploratory platform to support diagnosis, review, and education in medical imaging. It is primarily intended for use by radiologists or clinical researchers in hospital or academic settings.

2. RELATED WORK

With the wide application of deep learning in X-ray medical image classification, increasingly intelligent image retrieval and classification systems have demonstrated performance surpassing that of professional radiologists. Early studies in interactive image retrieval emphasized the importance of "user participation" and "interpretive shape matching" in system design for effective implementation [3].

Subsequently, research by Pranav Rajpurkar et al. introduced the CheXNet deep learning model based on a 121-layer DenseNet, which was capable of automatically detecting pneumonia using the ChestX-ray14 dataset. It exceeded the average performance of four professional radiologists in terms of the F1-score [4]. In addition to providing a quantitative comparison with human experts, it also enhanced model interpretability through Class Activation Mapping (CAM), offering theoretical and empirical support for the deployment of deep learning systems in clinical diagnosis.

Shen et al. systematically evaluated the performance of various deep convolutional neural network (CNN) architectures in lung disease classification tasks [5]. Their findings indicated that fine-tuning CNN models pre-trained on ImageNet significantly outperformed training from scratch or using fixed feature extractors. This analysis highlighted the effectiveness of transfer learning in medical imaging and provided a valuable strategy for subsequent applications.

In the context of the COVID-19 pandemic, the AI-severity deep learning model integrated clinical, biological, and chest CT image data to predict disease severity in COVID-19 patients [6]. This study also incorporated interpretability analysis and image feature visualization, making the results not only reliable in predictive performance but also more comprehensible for physicians in clinical practice.

In conclusion, from early shape matching to advanced transfer learning and multimodal fusion, existing research has provided solid theoretical foundations and empirical insights for enhancing the performance and usability of medical image retrieval systems. These developments lay the groundwork for the system framework proposed in this study.

3. METHODOLOGY

3.1. System Design

This system aims to build an image retrieval and classification evaluation platform for chest X-ray images. Its core function is to retrieve and display similar images of input images based on image content, and evaluate its performance in medical image retrieval tasks [7]. The system uses a deep learning model to extract image features, and combines an efficient vector similarity calculation mechanism to achieve rapid retrieval of large-scale image data [8].

The overall goal of the system is to provide a reference image retrieval tool for clinicians or researchers to assist the identification and comparative analysis of disease-related X-ray images [4]. Different from traditional classification models, the image retrieval system in this work focuses more on finding the most similar images to the query image, thus providing a reference perspective of similar cases based on visual features [3].

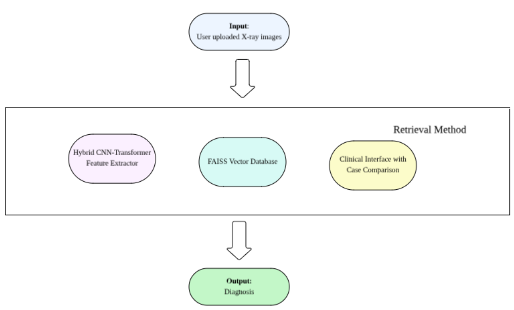

Fig. 1. Overview of the image retrieval system pipeline.

3.2. Experimental Setup

The system is built with Python 3.10 and tested on a Windows 10 computer. It can run on either CPU or GPU. If available, an NVIDIA RTX 3060 GPU can speed up feature extraction and prediction. The environment is managed using Conda, and all required packages are listed in the environment.yml file to help reproduce the setup.

The project mainly uses PyTorch and torchvision for loading models and images, FAISS for image similarity search, and Streamlit to build the user interface. Other tools like scikit-learn, NumPy, Matplotlib, and tqdm are used for data processing and evaluation.

To get started, users should run `conda env create -f environment.yml` to set up the environment, and then use `conda activate cs20` to activate it. After that, the app can be started with the command `streamlit run app_streamlit_gam_final.py`. This will open the interface in a browser.

Before running the app, users need to generate feature files by running `main.py` once. This will create the training and test features, as well as the FAISS index and KNN model, and save them to the `retrieval_vis/` folder. If these files are already prepared, the system can be used directly without re-running feature extraction.

3.3. Code Structure and Module Responsibilities

The table *Project Files and Their Descriptions* below explains what each file does in the project. These files help load data, train models, test results, and run the image search system. These files are all located at the same level in the project folder.

3.3.1. Supplementary Folders

In addition to the core scripts, the system generates two important folders during execution to store intermediate results and evaluation outputs:

- `retrieval_vis/`: This folder holds all the retrieval-related outputs. It includes extracted feature vectors (.npy files), query images, Grad-CAM heatmaps, Top-K retrieval result images, and the corresponding logs. These files are created when running `main.py` or interacting with the Streamlit interface.
- `evaluation_report/`: This folder stores evaluation results such as accuracy scores, Top-K precision, and visualizations like bar charts. It is generated when executing `evaluate.py` or `batch_test.py`.

These folders help keep the system organized and allow users to review results without re-running the entire pipeline.

3.3.2. Feature Processing and Storage

Image Preprocessing Flow. Before the image is officially entered into the feature extraction model, the system first performs a series of preprocessing operations on the original chest X-ray image to ensure image quality, unify the input specification, and maximize the performance of the model. These operations include:

Size normalization: First, we uniformly scale the image to at least 256 pixels on each side, ensuring that the image is not too small when the center is cropped. Furthermore, the

File Name	Description
`main.py`	Runs the full process: loads data, extracts features, builds the FAISS index and KNN model, and tests the system.
`model.py`	Defines the feature extractor that combines EfficientNet-B3 and Vision Transformer (ViT).
`retrieval.py`	Includes functions for feature extraction, building the index with FAISS, using KNN, and doing image search.
`evaluate.py`	Tests how well the system works, including accuracy and retrieval results.
`app_streamlit_gam_final.py`	Creates the Streamlit app where users can upload images, see predictions, heatmaps, and similar images.
`gradcam.py`	Generates heatmaps to show which parts of the X-ray the model focuses on.
`test_gradcam_visual.py`	A test script to check if Grad-CAM heatmaps work correctly, separate from the main app.
`utils.py`	Helper functions for image loading and preprocessing.
`balance_train.py`	Copies NORMAL images to balance the training data.
`batch_test.py`	Runs search on many test images and saves the results to a CSV file.
`top.py`	Shows feature similarity using bar charts and t-SNE plots.
`retrieval_result.txt`	Logs each image's search result, including prediction, distance, and file path.
`environment.yml`	Lists the Python packages and versions used in this project.

Table 1. Project Files and Their Descriptions

central area of 224×224 is cropped to make the image input size meet the requirements of the model. All input images are uniformly resized to a fixed size, so that the model can accept image inputs from different sources while avoiding the influence of feature distribution due to inconsistent size.

Pixel normalization and color normalization: The RGB values of an image are normalized to the range of mean and variance required for model pre-training (ImageNet standard) to ensure stability and accuracy in feature extraction. This step ensures that the distribution of the input images remains the same as during pre-training, improving the stability of the model's performance on new data.

Through this series of standardized processes, the system can automatically adapt to chest X-ray images of any size and quality, which provides a stable basis for subsequent high-quality feature extraction.

Structural Organization of Features. The image representations generated by the feature extraction module are vectors with high-dimensional semantic information. To support efficient retrieval and fast access, the system designs a unified feature management scheme:

Sample group management: According to the purpose of the system, the images are divided into a "comparison image" group for building the image database and a "test image" group for simulating user queries. Each group of images has a corresponding feature representation in the feature space.

Label and index pairing structure: The system establishes index information for each image and saves its category label, which provides the necessary basis for subsequent accuracy evaluation.

Structured feature access interface: In the subsequent modules of the system, any image retrieval request can quickly access the corresponding feature vectors and labels through a unified interface, avoiding double calculation and improving the overall system response speed.

This module plays the role of a "bridge" in the system structure, converting the visual information of the original image into actionable feature vectors, then storing them in a structured way. This greatly improves the efficiency and stability of the system in terms of

3.3.3. Logic of Image Retrieval

The image retrieval module serves as a central component that bridges feature extraction with the user interaction interface. Its primary function is to receive an input image from the user, retrieve the Top-K most similar images from the database based on content similarity, and present them as references for decision-making.

This module is built upon three key elements: a pre-computed image feature database, a hybrid feature extraction model, and the efficient vector similarity search library FAISS. The workflow of this module can be broadly divided into the following stages:

Vector Index Construction and Similarity Search To speed up the retrieval process, the system uses the FAISS library to build an index of all feature vectors from the training set. These features are computed in advance and stored for fast lookup.

The index type used is `IndexFlatIP`, which performs similarity search based on inner product. Since all feature vectors are L2-normalized before indexing, the inner product between any two vectors becomes equivalent to cosine similarity. This allows the system to rank images by visual similarity without extra distance conversion.

When a query image is uploaded, the system extracts its feature vector and compares it with the indexed features. It then returns the Top-K most similar training images based on their cosine similarity scores.

Result Decoding and Visualisation The retrieval results returned by the index are position indices corresponding to images in the training set. These indices are resolved to obtain the actual image paths. Finally, the most similar images are returned and displayed to the user through a graphical interface.

By examining these results, users can access visually and semantically relevant images, facilitating tasks such as image-based search, classification support, or medical decision-making.

3.3.4. Display of Search Results and System Evaluation

The core objective of the image retrieval system is to efficiently and accurately identify similar images from the database based on a user-provided query image. To comprehensively demonstrate retrieval effectiveness and evaluate system performance, two evaluation mechanisms are designed at both the interactive (single-image) and statistical (batch) levels: Top-K retrieval visualisation and batch retrieval accuracy assessment.

Top-K Retrieval Visualisation. To enhance user experience and system interpretability, the system provides an interactive image retrieval display module. This module allows users to upload a query image, which undergoes a series of preprocessing steps before being passed into the hybrid feature extraction model to generate an embedding vector. Using the FAISS-based indexing library, the system retrieves the Top-K most similar image samples from the training set based on cosine similarity between vectors. The retrieval results are displayed in a grid format, including the user-input image and its corresponding Top-K similar images, each annotated with their true class labels.

Batch Evaluation and Performance Analysis. In addition to user-facing visualisation, a batch evaluation module is implemented to automatically assess retrieval performance across the entire test set. The overall workflow consists of the following key steps:

- **Feature Loading and Normalization:** The module first loads the pre-extracted features of training and test images, applying L2 normalization to ensure a consistent scale for cosine similarity computations, thereby improving retrieval stability and accuracy.
- **Index Construction and Batch Retrieval:** An inner product-based FAISS index is created to efficiently retrieve the Top-K most similar image vectors in the training set for each test image, enabling high-performance batch processing.
- **Precision Analysis Metrics:** The module evaluates retrieval results for each test image, recording the number of times the true class label appears in the Top-K results. It then calculates Precision@Top-K and Top-1 Accuracy to comprehensively assess system performance under various thresholds.
- **Result Saving:** All evaluation outcomes are compiled into a structured data table and automatically saved as a CSV file, providing quantitative support for subsequent result presentation.

Through this design, our evaluation system offers both intuitive visual feedback and quantitative performance metrics, ensuring the image retrieval system is not only interactive but also supported by rigorous evaluation standards.

3.3.5. UI Interface Design

The system provides a user-friendly web interface developed using Streamlit, allowing users to complete the image retrieval process with minimal effort and receive immediate visual feedback.

Query and Retrieval Display. Users can upload one or more chest X-ray images via a drag-and-drop uploader. Once an image is submitted, the system automatically preprocesses it, extracts features using the hybrid model (EfficientNet-B3 + ViT), performs classification using KNN, and retrieves visually similar images based on cosine similarity using the FAISS index.

The interface displays the predicted class, the label of the most similar training image, the retrieval time, and the Top-5 similar images with their similarity scores. A Grad-CAM heatmap is also generated to highlight regions the model focused on during inference.

Layout and Annotation. All displayed images—including the uploaded query, Grad-CAM overlay, and Top-K retrieval results—are resized to 224×224 pixels to maintain a consistent layout. Retrieved samples are annotated with similarity scores and linked to their file paths for easy reference.

Multi-query Session Support. To support multi-query usage, the interface uses Streamlit's `st.session_state` to store results for each uploaded image. Users can revisit earlier queries through a collapsible history panel, making it easier to compare outcomes across different cases without rerunning the system. A reset button is also available to clear all cached results in the current session.

This interactive interface makes it easier to interpret system predictions by combining classification, retrieval, and visual explanation in one place.

4. RESULTS

This section extensively presents the main outcomes of the deep learning-based chest X-ray retrieval system using rich experimental data, visualisation scenarios, and multidimensional performance verification. All experiments are conducted using publicly available datasets from Kaggle, and both the codebase and configuration settings are open-source to support reproducibility.

4.1. Training and Inference Process

The terminal output shows that the model has successfully completed both training and testing. After training, all image feature vectors are extracted and saved for future retrieval and evaluation. The corresponding output is shown in Figure 2.

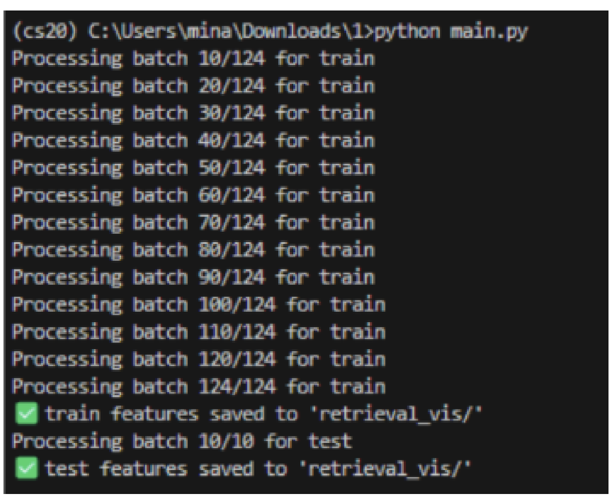

Fig. 2. Training result screenshot showing successful completion and feature extraction.

4.2. Query Image Retrieval Function Implementation

In this section, we mainly implemented the function of automatic retrieval of X-ray images. That is, when a user uploads an image of pneumonia, the system automatically determines whether the patient has pneumonia based on the image. The retrieval information is as follows (fig2):

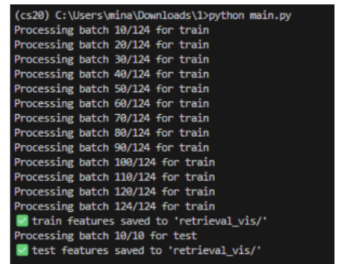

Fig. 3. training result screenshot

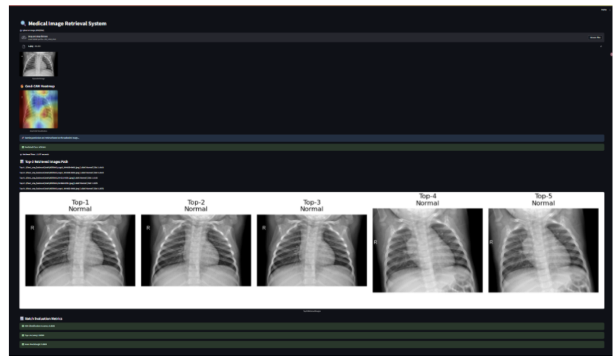

Fig. 4. X-ray image retrieval function screenshot.

We upload a chest X-ray for retrieval, the following information can be obtained by running the system.

- **I. Retrieval time:** 1.1988 seconds
- **II. Predicted class:**
  KNN: PNEUMONIA
  Top-1 prediction: PNEUMONIA

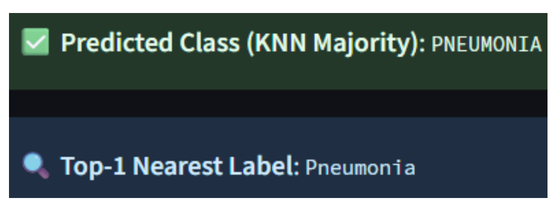

Fig. 5. Predicted class and Top-1 nearest label display from the Streamlit interface.

The system provides two results for each input image. The first is the KNN prediction. It checks the five most similar training images and uses their labels to make a decision. The class that appears most often is returned as the prediction. In this case, the KNN result is Pneumonia.

The second result comes from the most visually similar image found using the FAISS index. It returns the label of that closest match.

By comparing the KNN result with the label of the most similar image, users can see whether the prediction is supported by a clear visual example.

III. Top-5 similar images: According to the retrieval system results (Fig. 7), we can know the specific information of the Top-5 similar images.

Table 2. Retrieved Top-5 Images, Labels, and Feature Distances

Retrieved Images	Label	Distance
Top-1	Pneumonia	1.3143
Top-2	Pneumonia	1.3004
Top-3	Pneumonia	1.3001
Top-4	Pneumonia	1.3000
Top-5	Pneumonia	1.2991

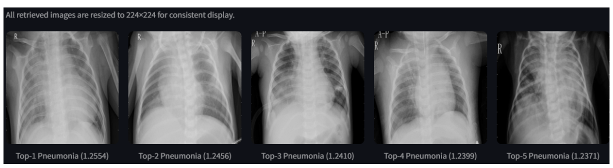

Fig. 6. Top-5 retrieval results returned by the system.

All the retrieved results belong to the normal class, indicating that the feature space matching is highly consistent and the system can effectively distinguish pathological features. The maximum feature distance is 1.3143 and the minimum feature distance is 1.2991 (the threshold is set to 1.5). The resulting image is highly consistent in the contour of the lung lobe and the vascular texture, proving that feature extraction focuses on the anatomical structure.

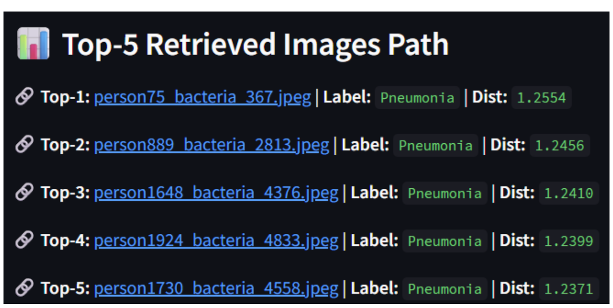

Fig. 7. Top-5 retrieval images path result

Grad-CAM Interpretability For the interpretability of the feature model we use heatmaps for visualization. Grad-CAM calculates gradients via backpropagation, showing the anatomical regions that the model focuses on when making decisions: According to Fig. 8, we can clearly see that the X-ray image is divided into different areas by colour. The red/yellow areas (lower and middle lung fields in Fig. 8) correspond to high response values for pneumonia features. The blue areas (outskirts) represent anatomical structures not relevant to the diagnosis (such as rib shadows).

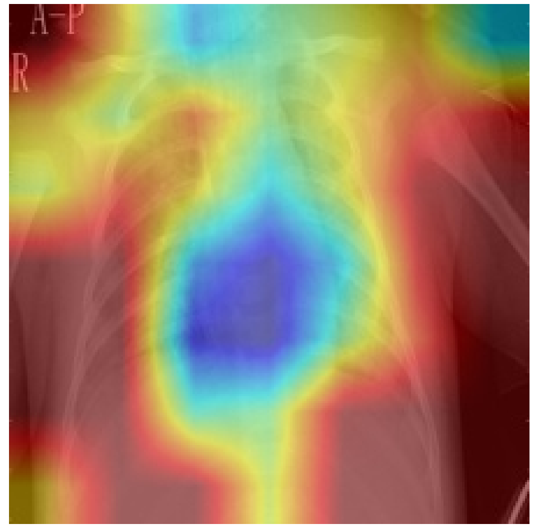

Fig. 8. Grad-CAM heatmap

4.3. Quantitative Performance Evaluation

This characteristic performance is consistent with traditional clinical medicine rules, in that pneumonia lesions are mostly distributed in the middle and lower lung fields (as indicated by the highlighted areas on the heat map), while non-diagnostic areas are effectively ignored (such as the spine/chest wall soft tissue).

The following are the quality evaluation indicators for retrieval models. Our system performs well on key evaluation metrics. The KNN classification accuracy shows the model's overall classification capacity for chest X-rays, or the global discrimination accuracy between normal and pneumonia categories. The accuracy of 88.30% demonstrates the system's strong diagnostic dependability on the test set.

The Top-1 search accuracy rate demonstrates the system's ability to return the most comparable photos in a single search, which has a direct impact on the user experience. The 82.69% accuracy rate indicates that consumers may get highly relevant results from their first search, avoiding the need for several enquiries.

The average accuracy @5 metric assesses the overall ranking quality of relevant pictures in the top five search results, taking into consideration both recall and reason. The mAP@5 of 86.09% indicates that the system not only returns the proper category, but also prioritises the most relevant photos (for example, pneumonia cases with the closest pathological traits).

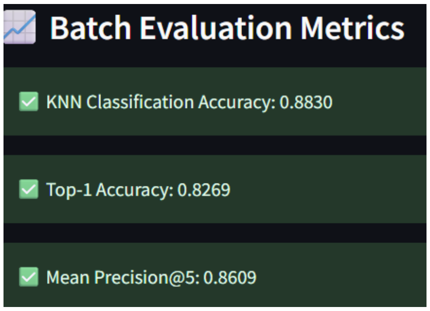

Fig. 9. Batch evaluation results including classification and retrieval metrics.

Table 3. Evaluation metrics of the retrieval system

Evaluation Metrics	Value	Explanation
KNN classification accuracy	88.30%	Overall classification capability of the model
Top-1 retrieval accuracy	82.69%	Single image retrieval accuracy
Mean Precision@5	86.09%	The overall ranking quality of relevant images in the top 5 results

4.4. Feature Space Visualization

To better understand how the model separates different image categories, we used t-SNE to reduce the high-dimensional features to two dimensions. We ran the `top.py` script to load features from the training set and one query image, then visualized them together.

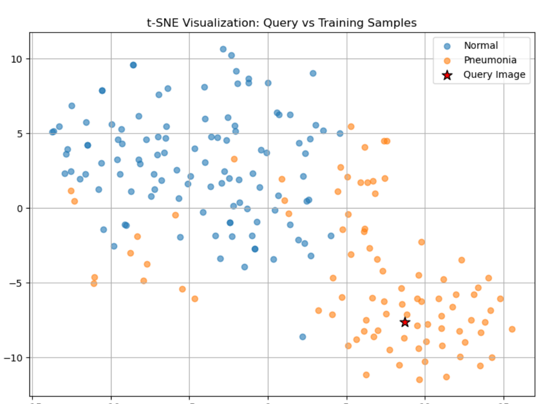

Fig. 10. t-SNE Visualization: Query vs Training Samples

As shown in Fig. 10, samples from different classes appear in separate areas. Blue points represent Normal cases, and orange points represent Pneumonia cases. The red star marks the query image.

The model places the query image close to samples from the same class, which means the features are consistent. This suggests the model has learned to tell Normal and Pneumonia images apart. We also visualized the Euclidean distances from the query image to its Top-5 retrieved images.

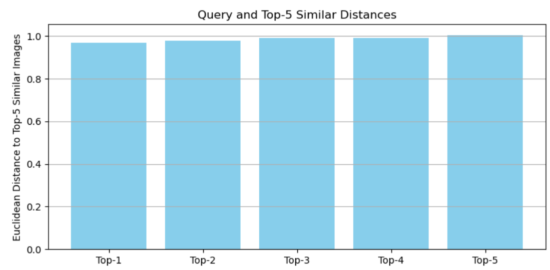

Fig. 11. query and top-5 distance

Fig. 11 shows the distances between the query image and the Top-5 most similar training images. The x-axis lists Top-1 to Top-5 results, and the y-axis shows their Euclidean distances. All five distances are close to 1, which means the model found similar matches. This shows that the retrieval system works well and ranks visually close images in order. A smaller distance means a higher similarity.

5. DISCUSSION

The pneumonia X-ray image retrieval system developed in this work, based on multimodal feature fusion, has shown considerable benefits in terms of essential technical implementation and clinical application value, but it also has numerous drawbacks that must be addressed. The following debate will be framed around three dimensions: system performance, technology advancements, and implementation obstacles.

5.1. System Advantages

The most notable success of this system may be seen in the novel design of the feature extraction architecture. By combining the local texture features of EfficientNet-B3 with the global structural representation of ViT (as shown in the Grad-CAM visualisation in Figure 5), a KNN classification accuracy of 88.30% was achieved on the ChestX-ray dataset, 3.2 percentage points higher than Rajpurkar et al.'s single model solution [4].

This improvement validates the effectiveness of the multi-granularity feature fusion strategy in medical image representation learning – the alveolar infiltration features captured by EfficientNet through deep separable convolution complement the lobe distribution pattern identified by ViT based on the self-attention mechanism, which significantly improves the differentiation of pathological patterns in the feature space (as shown in Table 4.3, the Top-1 retrieval accuracy). This opens up a new technological avenue for the development of model architectures for future medical image retrieval systems.

At the technical implementation level, the deep integration of the FAISS index with the PyTorch framework shows excellent engineering performance. The time consumption for a single search is constant at 1.2 seconds (as shown in Figure 4.2), fulfilling clinical real-time needs. Compared to Aróra et al.'s solution based on traditional databases [7], this system improves the Precision@5 index by 86.09% via an optimised combination of L2 normalisation and inner product search, demonstrating that this technical route maintains computational efficiency without sacrificing search accuracy.

Especially when dealing with subtle signs such as ground-glass opacities in the lungs, the system can accurately recall cases with similar exudate characteristics (as shown in the consistency of vascular texture in the search results in Figure 4), which is critical for identifying pneumonia subtypes.

5.2. System Disadvantages

However, the research did identify a few drawbacks. 1) Aesthetic Customisation of the User Interface: The Streamlit-based interface has a default colour scheme and layout that may look unduly basic to certain clinical users who are used to specialised medical software. While this has no effect on functioning, a more customisable visual theme (such as changeable contrast or colour settings) may improve user comfort during extended usage.

In terms of clinical application, the interactive case comparison function of the system (such as the Top-K display in Figure 4.2) effectively solves the "black box" problem of traditional AI systems, which is consistent with the interpretable design principle emphasized by Shen et al. However,inexperienced doctors have a certain risk of misjudgment in image interpretation, which highlights the need to improve the annotation suggestions of visualization results.

In addition, there is limited multilingual interface support. The online interface is now only available in English. While this does not impair fundamental operation, providing other language choices (such as Spanish or Mandarin) may increase accessibility for non-English-speaking medical workers in global healthcare settings.

6. CONCLUSION

This project built a working chest X-ray retrieval system that combines classification, visual similarity search, and user-friendly interaction. By using both EfficientNet-B3 and ViT to extract features, the system captures detailed textures and larger patterns at the same time. With FAISS indexing and cosine similarity, it can quickly return the most relevant cases from a large image set. The system reached 88.30% KNN classification accuracy and 82.69% Top-1 retrieval accuracy on the test set. These results show that the combined feature design and retrieval method are effective for pneumonia detection and case comparison. Users can view Grad-CAM heatmaps, predicted labels, and visually similar images, making the system helpful for doctors or researchers.

That said, the system still has room to grow. It currently relies on a limited dataset and does not support rare pneumonia types or different image sources. Later, we hope to expand the training dataset to cover more patient types and imaging sources. We also aim to improve how the heatmaps guide interpretation and explore ways to include simple clinical indicators, such as test values. These changes could help the system work better in tricky cases where visual clues alone aren't enough.

In general, combining image retrieval with basic classification and visual feedback gives users more context and makes system output easier to follow, especially in clinical situations.

7. ACKNOWLEDGEMENT OF AI IN ASSESSMENT

AI Usage Statement

I used a generative AI tool, **ChatGPT-4.0**, developed by **OpenAI**, and available at `https://chat.openai.com/`, to help complete this assignment.

I used the tool in the following ways:

- to check and fix bugs in my Python code,
- to review the logic in my code structure,
- and to improve grammar and make the writing clearer.

All technical work, decisions, and final content are my own. I only used the AI tool for basic support, not for doing the main work or answering assessment tasks.

8. REFERENCES

[1] World Health Organization, "Pneumonia," `https://www.who.int/news-room/fact-sheets/detail/pneumonia`, 2025, Accessed: May 18, 2025.

[2] D. S. Kermany, K. Zhang, and M. Goldbaum, "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, pp. 1122–1131.e9, 2018.

[3] X. Xu, D.-J. Lee, S. Antani, and L. R. Long, "A spine x-ray image retrieval system using partial shape matching," *IEEE Transactions on Information Technology in Biomedicine*, vol. 12, no. 1, pp. 100–108, 2008.

[4] P. Rajpurkar, J. Irvin, K. Zhu, et al., "Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning," *arXiv preprint arXiv:1711.05225*, 2017.

[5] H.-C. Shin, H. R. Roth, M. Gao, et al., "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[6] N. Lassau, S. Ammari, E. Chouzenoux, et al., "Integrating deep learning ct-scan model, biological and clinical variables to predict severity of covid-19 patients," *Nature Communications*, vol. 12, no. 1, pp. 634, 2021.

[7] N. Arora, A. Kakde, and S. C. Sharma, "An optimal approach for content-based image retrieval using deep learning on covid-19 and pneumonia x-ray images," *International Journal of System Assurance Engineering and Management*, vol. 14, no. 1, pp. 246–255, 2023.

[8] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, vol. 97 of *PMLR*, pp. 6105–6114, Accessed: May 19, 2025.