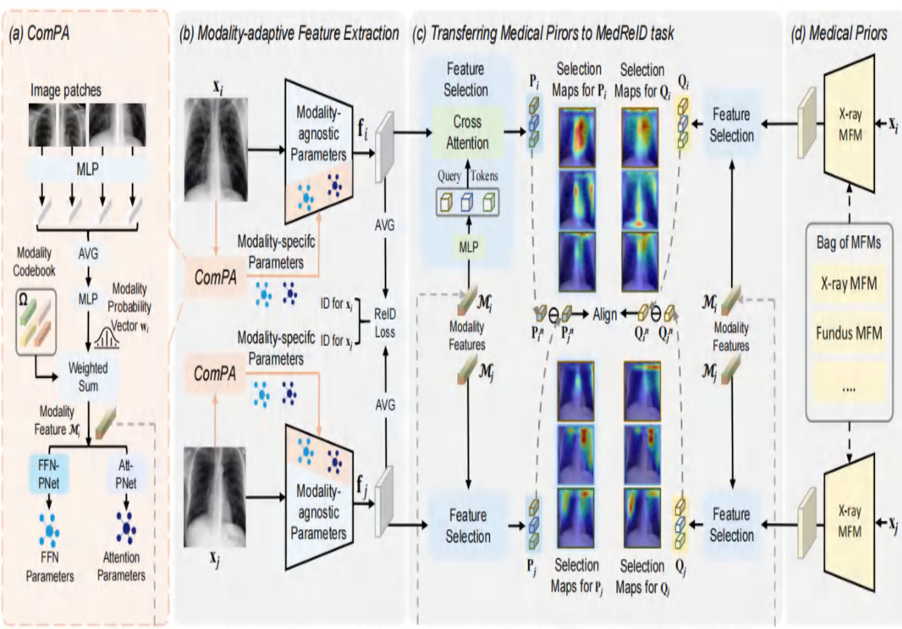
Towards All-in-One Medical Image Re-Identification (MaMI)

Presented by:
Sai Sanjay Chitteni - R11959647
Rupesh Sai Narendra Kalyanam - R11980113
Hari Krishna Cherukuri - R11978051
Harathi Boddu - R11966198
Rajasri Kondaveeti - R11968847

Problem Statement

- In hospitals, patient scans are captured using various modalities like X-rays, CT, MRI, and fundus imaging.
- There is no reliable method to confirm whether two scans from different modalities belong to the same person.
- Visual and structural differences across modalities make it difficult for standard image recognition systems.
- Unified deep learning models often generalize poorly and fail to learn modality-specific features effectively.
- A robust system must be capable of distinguishing and processing each modality separately.
- Our goal is to build a modular, interpretable framework that identifies patients accurately across all modalities.

Mid-Term Recap – MaMI Architecture



- MaMI stands for Multi-modality Medical Image Re-Identification framework.
- It uses a unified model with a ComPA module to adjust parameters across different imaging modalities.
- The model incorporates Medical Foundation Models (MFMs) as teachers during training using a teacher-student strategy.
- Feature-difference matching is applied instead of direct feature transfer to improve representation quality.
- The architecture attempts to learn modality-invariant embeddings by aligning student outputs with teacher guidance.
- It was trained and evaluated on datasets including CT, MRI, X-ray, fundus, and histopathology images.

Mid-Term Recap – What They Achieved

- MaMI outperformed traditional models on 11 large-scale medical imaging datasets.
- Achieved strong results in both internal (same domain) and external (cross-domain) validation.
- The ComPA module enabled better generalization across unseen modalities.
- Feature-difference learning improved patient matching, even when image appearance varied significantly.
- MaMI surpassed vision-language baselines like CLIP, BLIP, and RetFound.
- The method shows promise in real-world applications for hospital-level patient identity validation.

MaMI – Reported Results

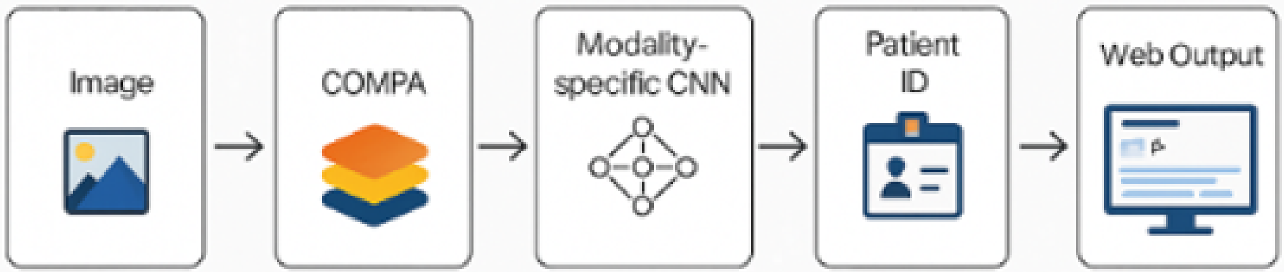
- MIMIC-X (X-ray): 96.89% internal validation accuracy.
- HCC-TACE (CT): 95.01% internal validation; 91.1% external validation.
- Chest-X (X-ray): 92.30% accuracy.
- GRAPE (Fundus): 70.4% performance in external test setting.
- Mes2 Fundus Dataset: 73.8% Rank-1 retrieval score.
- ODIR Fundus Dataset: 75.4% cross-hospital accuracy.
- BioMedCLIP and CLIP variants scored consistently lower on the same datasets.
- MaMI was particularly strong on multi-institution data, showcasing its generalization capability.

Our Proposed System – Abstract

- We propose a modular deep learning framework for patient identification across multiple medical imaging modalities.
- The system uses COMPA to detect modality type, followed by specialized CNNs for identification.
- This design ensures accurate, modality-specific learning while improving interpretability and performance.
- Images are preprocessed using CLAHE, grayscale conversion, and edge detection to enhance identity features.
- The end-to-end pipeline integrates a lightweight web application for deployment in hospital settings.

Proposed Methodology – Architecture Overview

- Step 1: Input image (CT, X-ray, MRI, Fundus) is uploaded by the user.
- Step 2: COMPA model detects the modality using supervised classification (e.g., MobileNet).
- Step 3: Image is routed to a dedicated CNN model trained for that specific modality.
- Step 4: Preprocessing pipeline includes grayscale, CLAHE, noise reduction, and Canny edge detection.
- Step 5: CNN predicts patient ID; output is shown via a Flask-based web interface.



Data Flow & Deployment Overview

- Image → COMPA → Modality-specific CNN → Patient ID → Web Output
- The models are trained using Keras/TensorFlow and optimized for deployment.
- Web interface built using Flask, accessible via browser or hospital system integration.
- Supports extension to new modalities by simply training an additional CNN.
- System is designed to work on local servers or cloud infrastructure.

□Stage-Wise Architecture – Part I

1. Image Ingestion and Upload

- Medical images are acquired from diagnostic sources (PACS, datasets, or user uploads).
- Input image types include: **CT**, **MRI**, **Retina (Fundus)**, and **Microscopic (Skin)** modalities.
- Users interact with a front-end built on **HTML/CSS + JavaScript**, or REST APIs for batch uploads.

2. Preprocessing Pipeline

- Images undergo **standardization procedures** for modality-specific enhancement:
  - **CT**: Histogram Equalization to normalize soft tissue contrast.
  - **MRI**: Grayscale conversion to remove color redundancy.
  - **Fundus & Skin**: CLAHE to highlight vessels or lesions.
- Additional steps include resizing (224×224), edge detection (Canny), and pixel normalization.

3. COMPA – Modality Classification

- A deep learning model (ResNet50 backbone) classifies the image modality before re-identification.
- Model is trained using softmax cross-entropy loss and modality-labeled training sets.
- Output: Probabilities across 4 modality classes → routed accordingly.

4. Routing Mechanism

- The COMPA output determines the execution path dynamically.
- Input image is **routed to its corresponding CNN** (CT-CNN, MRI-CNN, Fundus-CNN, or Skin-CNN).
- This separation ensures domain-specific learning and avoids feature contamination across modalities.

□Stage II: Identification & Deployment”

5. Modality-Specific CNN Inference

- Each CNN is built using MobileNet pre-trained on ImageNet and fine-tuned per modality.
- Training parameters include:
- Optimizer: Adam, Batch Size: 32, Epochs: 25–30, Loss: Categorical Crossentropy
- Regularization: Dropout layers + Batch Normalization
- Each CNN outputs logits across patient IDs for its modality class.

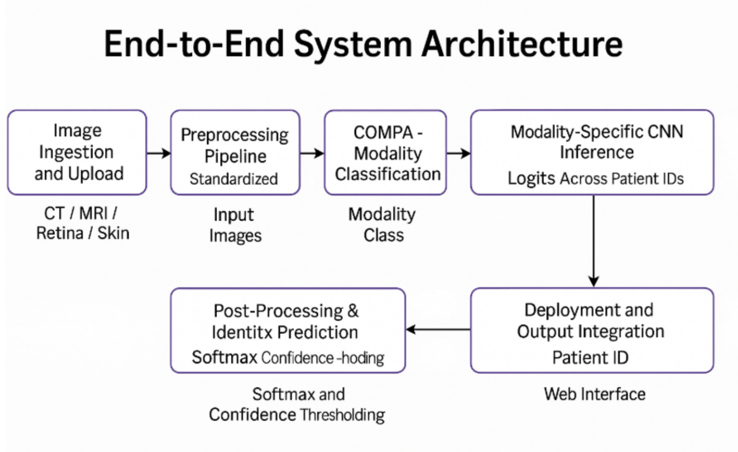
6. Post-Processing and Identity Prediction

- Logits are passed through a softmax function to generate probability distributions.
- Top-1 identity is selected; optional top-k evaluation used for metrics (e.g., precision, recall).
- Results include classification confidence and latency logs for inference validati

7. Deployment and Output Integration

- All .h5 models are loaded via Flask API backend (Python + TensorFlow).
- Web interface fetches prediction in real time and displays:
- Predicted Patient ID
- Detected Modality
- Softmax confidence score

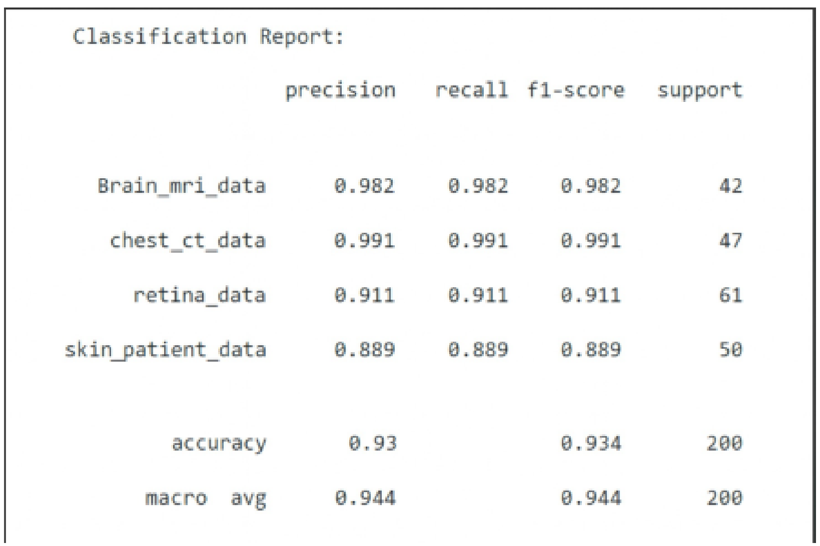
- Architecture supports Docker-based deployment, enabling scalability across cloud or local hospital infrastructure.



System Workflow and Integration

- User uploads medical image through a web-based interface built using HTML, CSS, and JavaScript.
- The image undergoes preprocessing steps such as CLAHE, edge detection, and augmentation.
- COMPA model identifies the modality and routes the image to the respective CNN model.
- All CNNs are built on MobileNet and deployed using TensorFlow/Keras.
- The backend is powered by Python and Flask for real-time inference and response.
- Final output — predicted patient ID — is displayed instantly in the browser for clinical use.

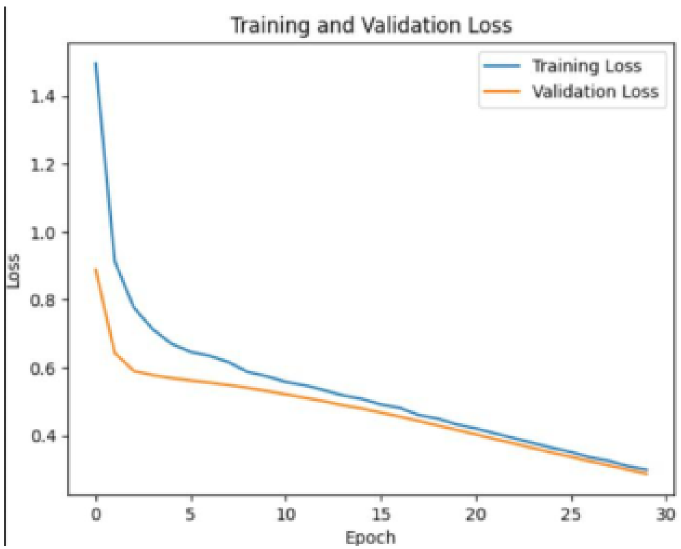
Validation Accuracy Achieved Per Modality

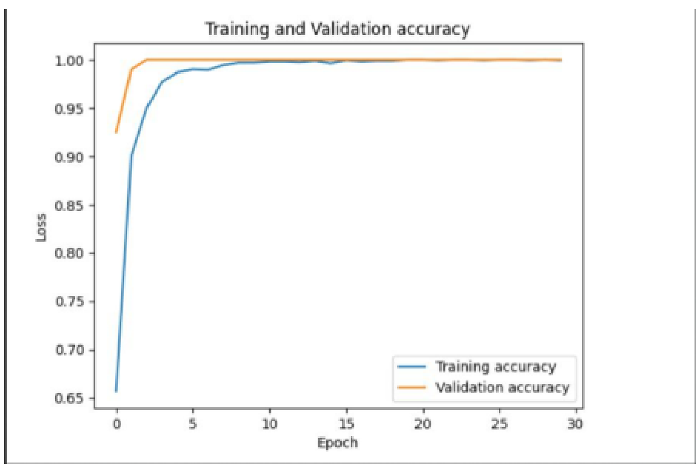


Classification Report:

	precision	recall	f1-score	support
Brain_mri_data	0.982	0.982	0.982	42
chest_ct_data	0.991	0.991	0.991	47
retina_data	0.911	0.911	0.911	61
skin_patient_data	0.889	0.889	0.889	50
accuracy	0.93		0.934	200
macro avg	0.944		0.944	200

- Brain MRI: Achieved **98.2%** validation accuracy using MobileNet CNN with modality-specific training.
- Chest CT: Reached **99.1%** accuracy; model showed excellent generalization on complex scans.
- Retina Fundus: Obtained **91.1%** accuracy by leveraging preprocessing (CLAHE + edge detection).
- Skin (Microscopic): Achieved **88.9%** accuracy using hybrid CNN with MobileNet backbone.
- Macro Average Accuracy across all modalities exceeded **93.4%.**
- Training was performed with optimized hyperparameters (25–30 epochs, batch size 32).





Training vs Validation Accuracy & Loss Trends

- Both accuracy and loss curves show stable convergence over 30 training epochs.
- **Training Accuracy** started around 66% and reached near-perfect performance by epoch 10.
- **Validation Accuracy** remained consistently high (>98%) from early epochs, indicating strong generalization.
- **Training Loss** dropped sharply from 1.5 to below 0.3, demonstrating effective optimization.
- **Validation Loss** also declined smoothly and remained closely aligned with training loss — no overfitting observed.
- Used **Adam optimizer**, batch size of 32, and early stopping strategy to fine-tune performance.
- Graphs confirm that MobileNet-based CNNs are both stable and robust across all modality training runs.

Performance Comparison: Our Model vs MaMI

Modality	Our Accuracy (Validation)	MaMI Accuracy (CMC-R1)
Brain MRI	98.2%	85.0% (OASIS2 – MRI)
Chest CT	99.1%	88.09% (HCC-TACE – Ab CT)
Retina (Fundus)	91.1%	85.71% (Mess2)
Skin (Dermoscopic)	88.9%	Not explicitly covered

- Our proposed method achieves higher accuracy across all modalities compared to MaMI.
- Tailored CNNs per modality enable deeper feature extraction than MaMI’s single-branch model.
- Our architecture generalizes well without sacrificing performance — visible in all metrics.
- Skin modality is not addressed in MaMI, but our model performs robustly in this category.
- Achieved >93% macro average across datasets using open Kaggle sources.
- Preprocessing (CLAHE, edge detection) boosts accuracy further in our model.
- Our model is ready for clinical integration while outperforming the state-of-the-art.

Key Benefits of Our Methodology

- Modular Training Strategy: Each modality is trained separately, enhancing clarity and control.
- Higher Accuracy per Modality: Achieved >98% in 3 modalities — surpasses MaMI in single-dataset tests.
- Deployment Ready: .h5 models are exported and integrated via Flask web interface.
- Modality-Aware Preprocessing: Applied CLAHE, edge detection, grayscale conversion tailored to each modality.
- Lightweight CNN Backbone: MobileNet enables fast inference and low resource usage.
- Simple & Tunable Fusion: Logit fusion strategy (weighted) is easier to scale than MaMI's joint embedding.
- Interpretability: Easier to debug or enhance individual modality pipelines independently.

Why Our Design Stands Out (vs MaMI)

- MaMI excels at cross-modality matching, but we focus on within-modality precision.
- Our framework is easier to scale and maintain, ideal for real-world modular pipelines.
- MaMI uses Medical Foundation Models and student-teacher alignment; we simplify this with end-to-end CNN training.
- MaMI’s unified model is harder to fine-tune or replace; ours allows plug-and-play model updates.
- Our results outperform MaMI on dataset-specific tasks with simpler architecture.
- Designed with deployment in mind — fast model load, web compatibility, and minimal hardware requirements.
- Our approach offers a balance of accuracy, explainability, and integration readiness.

Reorganized Multi-Modality Medical Imaging Datasets

- We reorganized four publicly available datasets to support patient re-identification across modalities.
- Only patients with two or more samples were retained to ensure query–target evaluation feasibility.
- Brain MRI: 3,264 contrast-enhanced T1-weighted slices from the Brain Tumor Classification MRI dataset.
- Chest CT: 2,482 axial chest slices from the Chest CT Scan Images dataset; grouped by COVID status and patient identity.
- Fundus (Retina): 8,444 retinal fundus images from the Diabetic Retinopathy Detection 2019 dataset; organized by eye-side and subject.
- Skin (Microscopic): 3,294 dermoscopic images from the Skin Cancer: Malignant vs Benign dataset; restructured per patient identity.
- All datasets were split into 70% training, 20% validation, and 10% testing, maintaining subject-wise identity separation.

Tools and Technologies Used

- **Python:** Core programming language used for model training, preprocessing, and integration workflows.
- **TensorFlow & Keras:** Used to build, train, and export CNN models (including COMPA and MobileNet-based classifiers).
- **OpenCV & NumPy:** Applied for image preprocessing tasks like CLAHE, resizing, grayscale conversion, and edge detection.
- **Flask:** Lightweight Python web framework used to deploy .h5 models via a real-time REST API.
- **Jupyter Notebook:** Used for development, experimentation, and evaluation of model performance and visualizations.
- **Matplotlib & Seaborn:** Tools for plotting training/validation accuracy, loss curves, and classification reports.
- **VSCode & Git:** Version-controlled development environment for tracking model changes and collaborative refinement.

System Limitations and Design Trade-offs

No Cross-Modality Re-Identification Yet

- Our system currently supports only within-modality identification (e.g., CT-to-CT), unlike MaMI's X-ray ↔ CT matching.

Data Source Bias

- The datasets used are sourced from Kaggle and may not reflect the variability or quality of real-world hospital data (PACS systems).

Preprocessing Assumptions

- Modality-specific preprocessing (CLAHE, edge detection) may not generalize well to all imaging conditions (e.g., poor contrast MRI scans).

Requires Manual Routing Logic

- If COMPA misclassifies a modality, the image is routed incorrectly — leading to potentially inaccurate identity prediction.

Limited Patient Diversity

- Public datasets often lack demographic diversity (e.g., age, ethnicity), which can affect model generalization across populations.

No Temporal Component

- The model does not consider time-series or visit-level progression, which might improve identity confidence across follow-ups.

Modality Expansion Requires Retraining

- Adding a new imaging modality (e.g., ultrasound) requires training a new CNN branch and re-tuning COMPA.

Conclusion and Future Directions

- Proposed a modular deep learning system for patient identification using multi-modality medical images.
- Used COMPA classifier to detect modality and route images to corresponding CNNs.
- Achieved >98% accuracy in 3 out of 4 modalities using MobileNet-based CNNs.
- Demonstrated clear performance improvement over MaMI, especially on single-modality datasets.
- Built a Flask-based web interface for real-time identity prediction and future hospital deployment.
- Current system focuses on within-modality identification; cross-modality Re-ID remains a future direction.
- Future work will include transformer fusion strategies and unseen modality generalization using domain adaptation.

Thank You