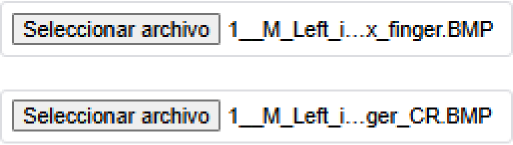
# 🧾 Operation Manual: Fingerprint Biometric Comparison

### 1. Uploading Images

In the form, upload **two fingerprint images**:

- 📥 **Requester**: the fingerprint image of the person to be verified.
- 📥 **Candidate**: the fingerprint image to compare against.

The images must be in .bmp or .png format and should contain **complete or partial fingerprint scans**, such as those provided in the sample link below.



Seleccionar archivo 1__M_Left_i...x_finger.BMP

Seleccionar archivo 1__M_Left_i...ger_CR.BMP

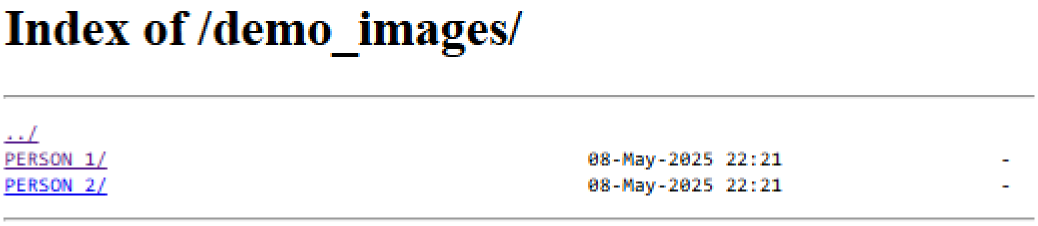
### 2. Sample Images for Testing

To run real tests using valid data, download sample images from the following link:

🔗 [Download Sample Images]

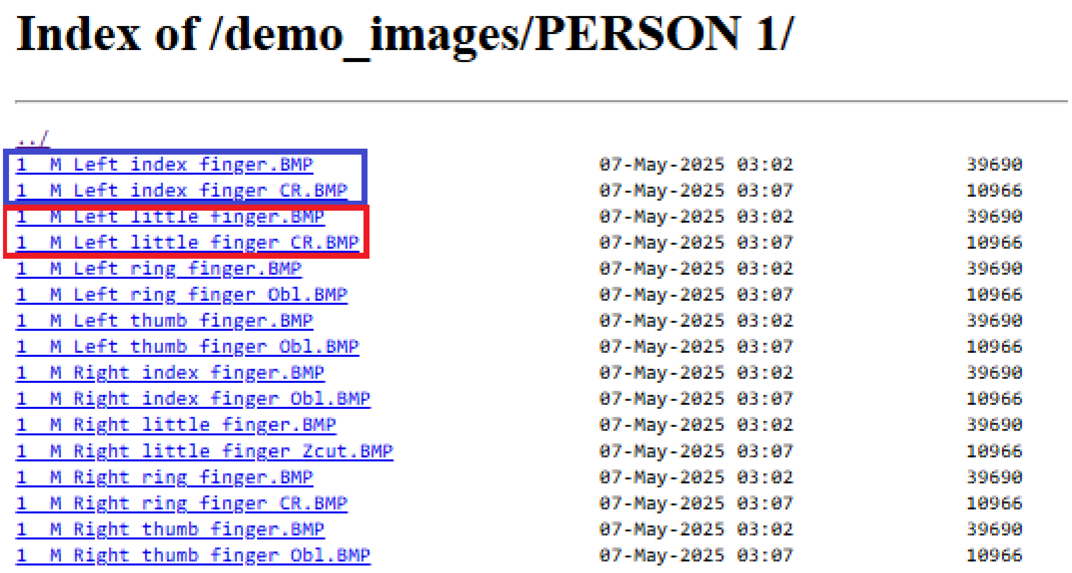
The dataset is organized as follows:

- Each individual has **10 fingers**, and each finger includes **two images**:
  - A **clean** (undistorted) version.
  - A **distorted** version (with rotation, pressure, blur, or cropping).



# Index of /demo_images/

```
../
PERSON 1/                                          08-May-2025 22:21                   -
PERSON 2/                                          08-May-2025 22:21                   -
```



```
Index of /demo_images/PERSON 1/

../
1__M_Left_index_finger.BMP                         07-May-2025 03:02               39690
1__M_Left_index_finger_CR.BMP                      07-May-2025 03:07               10966
1__M_Left_little_finger.BMP                        07-May-2025 03:02               39690
1__M_Left_little_finger_CR.BMP                     07-May-2025 03:07               10966
1__M_Left_ring_finger.BMP                          07-May-2025 03:02               39690
1__M_Left_ring_finger_Obl.BMP                      07-May-2025 03:07               10966
1__M_Left_thumb_finger.BMP                         07-May-2025 03:02               39690
1__M_Left_thumb_finger_Obl.BMP                     07-May-2025 03:07               10966
1__M_Right_index_finger.BMP                        07-May-2025 03:02               39690
1__M_Right_index_finger_Obl.BMP                    07-May-2025 03:07               10966
1__M_Right_little_finger.BMP                       07-May-2025 03:02               39690
1__M_Right_little_finger_Zcut.BMP                  07-May-2025 03:07               10966
1__M_Right_ring_finger.BMP                         07-May-2025 03:02               39690
1__M_Right_ring_finger_CR.BMP                      07-May-2025 03:07               10966
1__M_Right_thumb_finger.BMP                        07-May-2025 03:02               39690
1__M_Right_thumb_finger_Obl.BMP                    07-May-2025 03:07               10966
```

These variations are designed to test the **accuracy of the comparison algorithm**, determining whether it can still match fingerprints from the **same person** despite differences in capture conditions.

### 3. Performing the Comparison

After uploading both images:

1. Click the **"Compare"** button.
2. The system will process the fingerprints, identify **relevant matching regions**, and return:
   - ✅ **Match** (if both belong to the same person),
   - ❌ **No Match** (if they belong to different people),
   - 🧠 Optionally, a visual explanation highlighting the areas that influenced the decision (e.g., using Grad-CAM or overlay visualization).

## 🧾 Technical Detail

### 🚀 Fingerprint Matcher Using Siamese Neural Network + Grad-CAM

Over the past few days, I've developed an educational project focused on **fingerprint comparison using deep learning techniques**. The solution is built on a **Siamese Neural Network architecture**, using **MobileNetV2** as a lightweight feature extractor — and incorporates a **Grad-CAM visualization module** to highlight the most relevant fingerprint regions used during biometric verification.

---

### 🔬 💡 Technical Overview

This project offers an alternative approach to traditional biometric verification systems. While classical methods rely on heuristic algorithms for **minutiae extraction** (e.g., ridge bifurcations and endings), this solution leverages a neural network that **automatically learns vector representations (embeddings)** directly from image pixels.

These embeddings are then compared using **Euclidean distance**, providing a similarity measure that is robust to noise, rotation, distortion, and low-quality captures — all common challenges in real-world biometric environments.

---

### 📊 🧪 Training Dataset — SOCOFing

The SOCOFing dataset was used for training, comprising over **6,000 BMP-format fingerprint images** with artificially applied variations (cuts, noise, rotation, dirt). This allowed for realistic simulation of forensic and administrative scenarios.

---

### 🛠️ Training Architecture

- 📐 Image size: 96×96 px (RGB)
- 🔁 Batch size: 32
- 🔄 Epochs: 5
- 📉 Loss function: Binary Crossentropy
- ⚙️ Optimizer: Adam

- 🧠 Output model: model/siamese_model.h5

Custom lightweight CNN with:

- 2 Conv2D layers
- 2 MaxPooling2D layers
- 1 Flatten + Dense (128-dim embeddings)
- Lambda layer for Euclidean distance

---

### 🧩 Web Inference + Grad-CAM

The app.py application was developed using **Flask**, with a simple and interactive web interface for uploading .bmp images. The system:

- Processes input image pairs
- Generates embeddings
- Returns a similarity score + **MATCH / NO MATCH**
- Optionally displays a **Grad-CAM heatmap** over the image, highlighting the neural activation regions most responsible for the decision.

The heatmap can be enabled via a checkbox and enhances interpretability of the matching process.

---

### ☁️ Execution Environment (Oracle Cloud)

- Ubuntu 20.04 (Canonical)
- Instance: VM.Standard.E2.1 (1 vCPU, 8 GB RAM)
- TensorFlow CPU-only (no GPU)
- NGINX + Gunicorn + systemd

---

### 📦 Tech Stack

- TensorFlow 2.x + Keras
- Flask, OpenCV, Matplotlib, Pillow

- Custom Grad-CAM implementation
- Automated deployment via setup.sh + NGINX reverse proxy

---

📂 **Repository + Live Demo**

🔗 **Live demo:** http://164.152.33.142/
🔗 **GitHub repository:** https://github.com/algodas/BiomatchML.git

Includes:

- Full codebase (app.py, train.py, cam_utils.py)
- Demo fingerprint images
- Auto-install script (setup.sh)
- Grad-CAM integration and web-based reprocessing

---

💬 If you work with biometrics, computer vision, or are interested in applying Siamese Networks in contexts like signature verification, face recognition, document comparison or voice matching — let's connect!

#DeepLearning #Biometrics #SiameseNetwork #ComputerVision #GradCAM #Flask #TensorFlow #Keras #MobileNetV2 #AI #MachineLearning #GitHub #OpenSource #EducationalProject #NeuralNetworks #FingerprintRecognition