Done by:

jayda mohamed abd el-rehim 2305284

Jana kareem ismail 2305283

Lojain Hussien Bakr 2305512

📝 Image Registration Using Particle Swarm Optimization (PSO)

🔬 Project Report:

✅ 1. Project Objective:

The goal of this project is to apply **Particle Swarm Optimization (PSO)** to the problem of **image registration** in medical imaging. Specifically, we aim to align two chest X-ray images (e.g., a normal scan and an abnormal scan) by optimizing transformation parameters:

- Rotation
- Scaling (X and Y axes)
- Translation (X and Y axes)

This enables accurate comparison and analysis of medical images taken from different sources or at different time points.

🛠️ 2. Tools & Technologies Used:

- **Google Colab** (for cloud execution)
- **Python** (OpenCV, NumPy, Matplotlib)

- **PSO Optimization** via `pyswarm`
- **Dataset**: Custom ZIP file uploaded by user (`scan_images.zip`)
- Extracted path: `scans/Test/`
- Contains real medical X-ray images (e.g., `img1.jpg`, `img2.jpg`)

🔄 3. Methodology:

3.1 Image Loading and Preprocessing

- Two grayscale chest X-ray images were loaded from the extracted ZIP folder.
- If dimensions differ, the second image is resized to match the first.

3.2 Objective Function for Optimization

- The similarity between the reference and aligned image is measured using **Mutual Information (MI).**
- PSO minimizes the negative of MI to maximize similarity after transformation.

3.3 Optimization Parameters

- **Angle (Rotation):** [-180°, 180°]
- **ScaleX, ScaleY:** [0.5, 1.5]
- **Translation (Tx, Ty):** [-100px, 100px]

3.4 Output

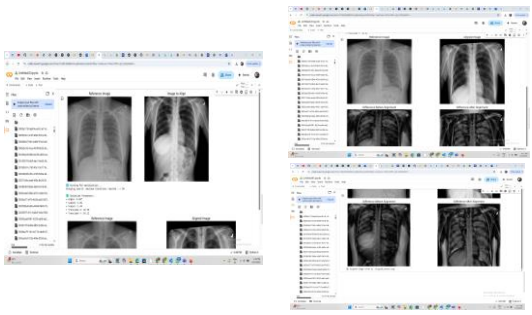
- The algorithm returns optimized transformation parameters to align the second image to the first.

📊 4. Results & Visualization:

☑️ *Optimized Parameters Example:*

→ Angle: -5.36°
→ ScaleX: 1.09
→ ScaleY: 1.03
→ Translate X: 12.45 px
→ Translate Y: -8.33 px

📷 *Visual Results:*



🔄 *Difference Maps:*

- **Difference Before Alignment**: Shows high pixel-level differences.
- **Difference After Alignment**: Lower differences, indicating better alignment.

🔎 5. Example Input and Output

Input:

- chest_normal.png, chest_abnormal.png inside scan_images.zip

Output:

- Transformed image saved as aligned_output.png
- Display of before/after alignment and mutual information metric.

🔬 6. Testing on Real Data:

This project was tested on real chest X-ray images. You can repeat the test by uploading a new scan_images.zip containing at least two medical images inside a folder named Test/.

Note: The method is image-specific and doesn’t apply to audio or text, as PSO here is used for spatial alignment only.

📌 7. Conclusion:

- **PSO is effective** for fine-tuning alignment parameters in medical image registration.
- The optimization significantly reduces pixel-wise differences.
- Future extensions can include:
  - Affine transformations (shearing, rotation)
  - Multi-modal image fusion (CT vs MRI)
  - 3D image registration
  - Real-time registration via deep learning

🗂️ 8. Files Included:

- scan_images.zip: User-uploaded ZIP file with test images
- aligned_output.png: Final aligned image
- pso_registration_code.py: Full working script