RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

Department of Computer Science and Engineering

P.R.I.S.M.
PREDICTIVE RECOGNITION OF INDIVIDUALS' SANGUINE (BLOOD) MARKERS

Yuven Senthilkumar 220701330
Shiiv R S 220701331

Dr Rakesh Kumar M

Problem Statement

- Traditional blood group tests:
- Invasive & lab-dependent
- Slow & impractical in emergencies/rural areas
- Existing non-invasive methods:
- Low accuracy & poor usability

Our Aim:

- Predict blood group from fingerprint images using a CNN-based deep learning model
- Provide instant, non-invasive results via a Streamlit web app

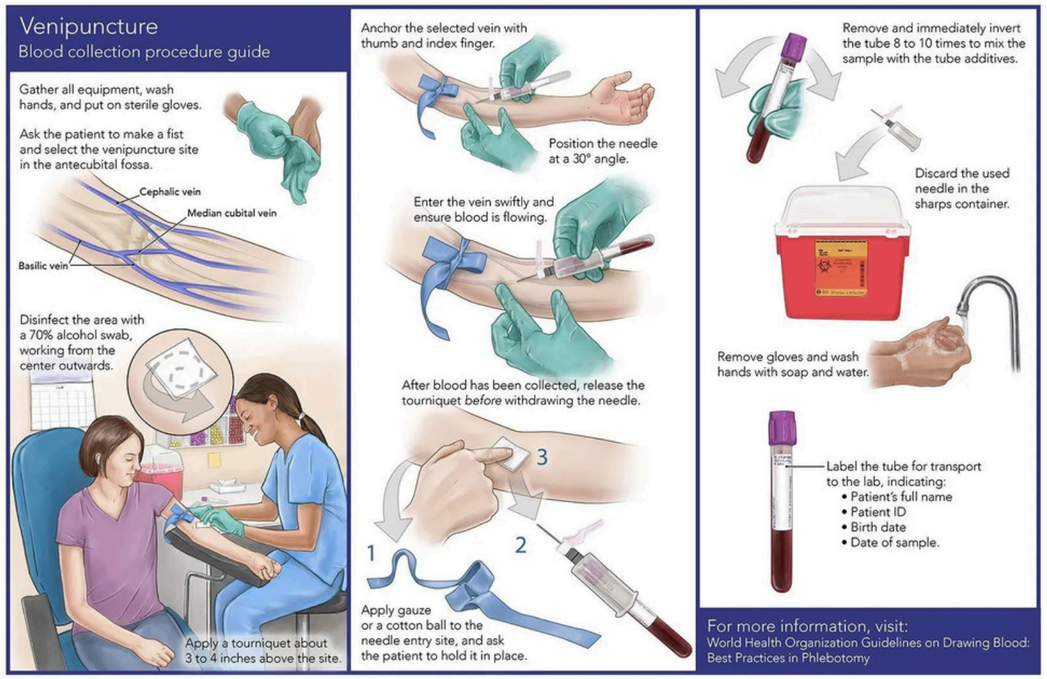
Motivation

- Healthcare Need: Quick blood group detection for transfusions, transplants & emergencies
- Accessibility: Helps areas lacking medical infrastructure
- Innovation: Combines AI & biometrics for next-gen, non-invasive diagnostics

EXISTING SYSTEM

Blood Group Identification:

- Relies on serological/agglutination tests
- Needs trained staff & lab setup
- Not suitable for emergency/rural scenarios



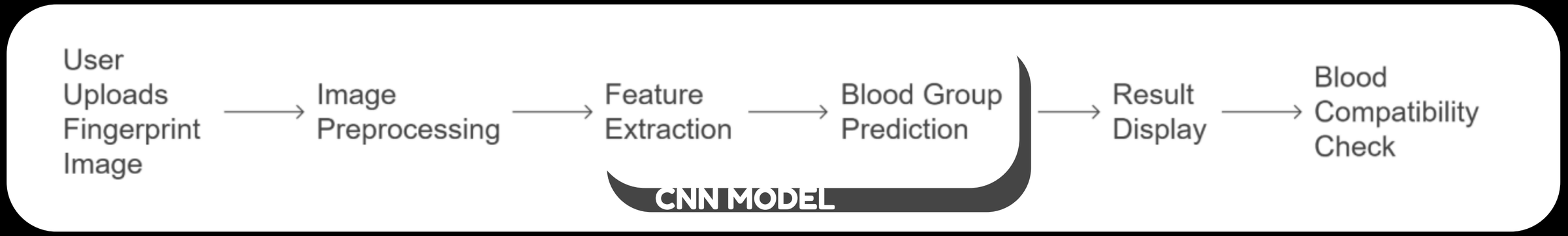
Biometric-Based Research:

- Explores fingerprint patterns (dermatoglyphics) linked to blood groups
- Machine learning models like CNNs used for prediction
- Early studies show potential but:
- Small, non-diverse datasets
- Limited real-world application
- Often lack usability & accuracy

Current Limitations:

- No fully integrated solution combining:
- High accuracy
- Real-time usability
- Strong data security & privacy

Objectives



- Build a CNN model to predict blood groups from fingerprint images
- Create a diverse, labeled dataset
- Achieve high accuracy & reliability
- Develop a Streamlit app for:
  - Easy fingerprint upload
  - Instant predictions
- Add a Blood Compatibility Checker
- Ensure privacy, security & real-world usability

Abstract

- Predicting blood group is vital for transfusions & emergency care
- Traditional methods = invasive & lab-dependent

Our solution:

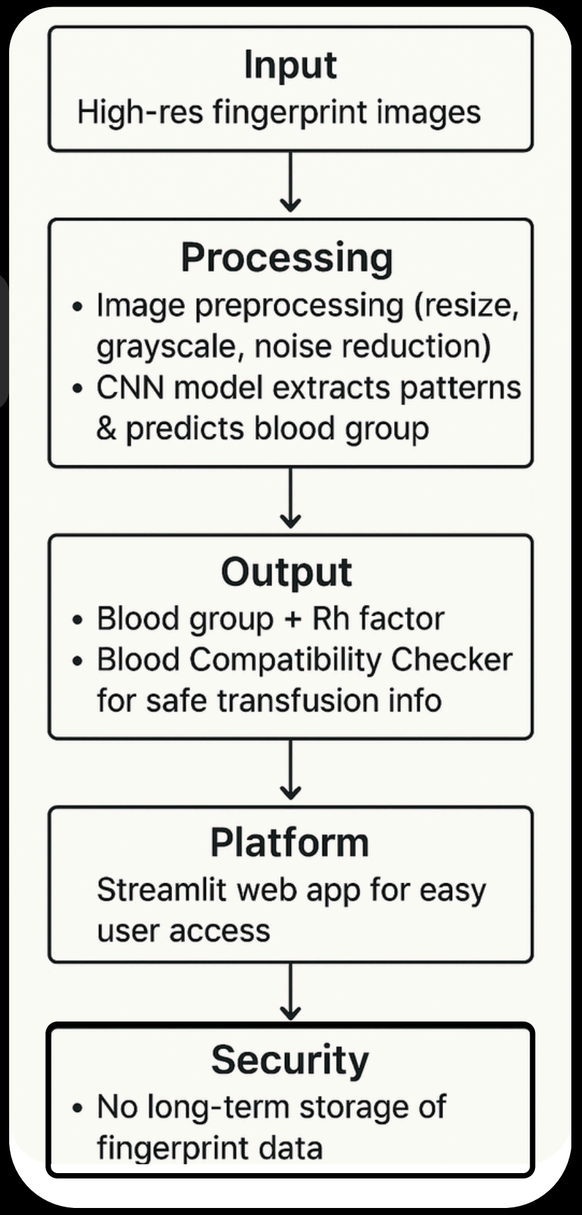
- CNN-based model predicts blood group from fingerprint images
- Fully integrated with a Streamlit web app for instant, non-invasive results

Key features:

- High accuracy & performance metrics
- Blood Compatibility Checker for safe transfusions
- Impact:
- Makes blood grouping accessible & fast
- Useful in rural & emergency settings

Proposed System

- 📸 Input: High-res fingerprint images
- ⚙️ Processing:
- Image preprocessing (resize, grayscale, noise reduction)
- CNN model extracts patterns & predicts blood group
- 🌐 Output:
- Blood group + Rh factor
- Blood Compatibility Checker for safe transfusion info
- 💻 Platform:
- Streamlit web app for easy user access
- 🔒 Security:
- No long-term storage of fingerprint data
- Data privacy & integrity prioritized



System Architecture

User Interface (Streamlit App):

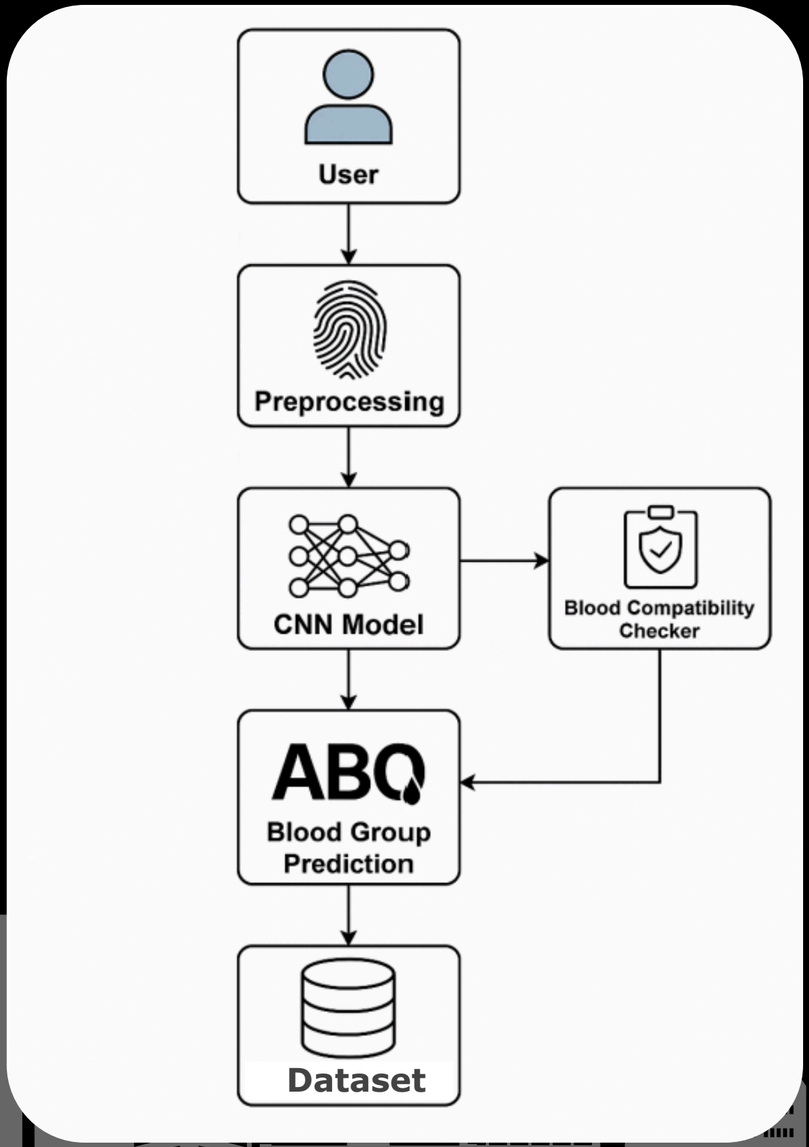
- Upload fingerprint
- View predictions & compatibility info
- Preprocessing Module:
- Grayscale conversion
- Resize & noise reduction

CNN Model:

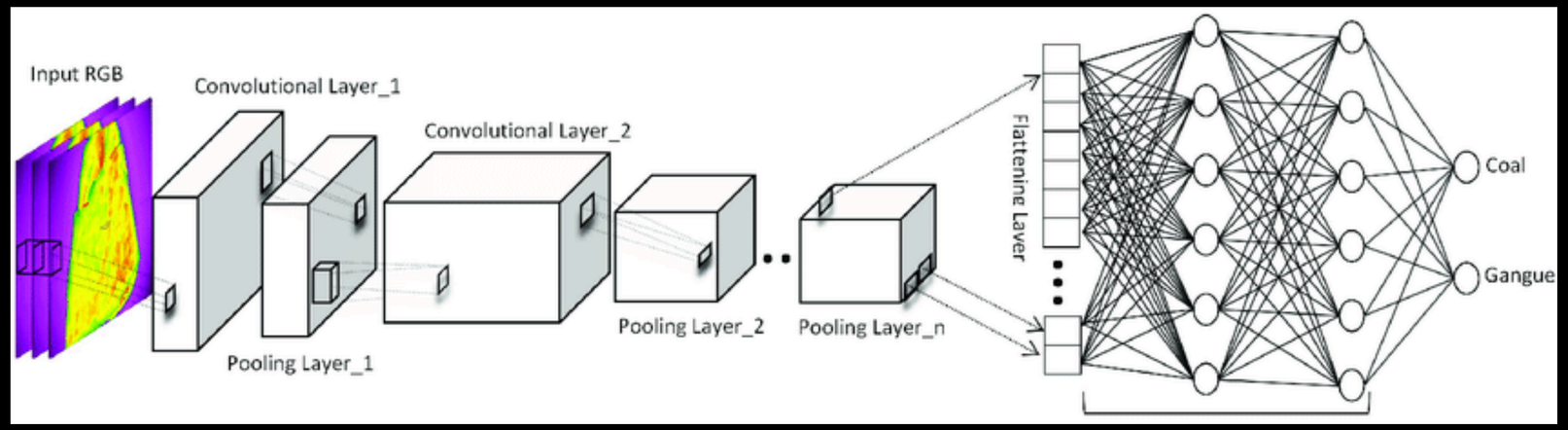
- Feature extraction from fingerprint
- Blood group classification (A, B, AB, O + Rh)

Output Module:

- Display results & confidence score
- Blood Compatibility Checker
- Backend (Optional):
- Secure storage (if needed)
- Logs & future model updates



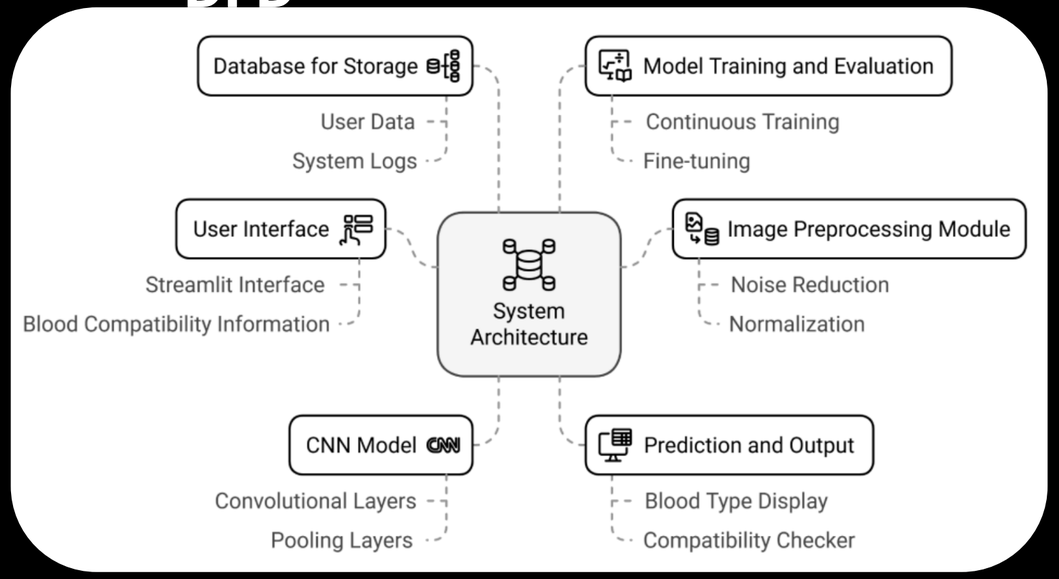
List of Modules



1. User Interface – Upload & view results
2. Backend – Preprocessing & routing
3. Data Handling – Collection, labeling,augmentation
4. Model – CNN-based prediction
5. Evaluation – Accuracy, precision, recall
6. Deployment – Streamlit app integration
7. Database – Optional secure storage
8. Extras – Fraud detection, blockchain, workflow

DFD and Sequence Diagram

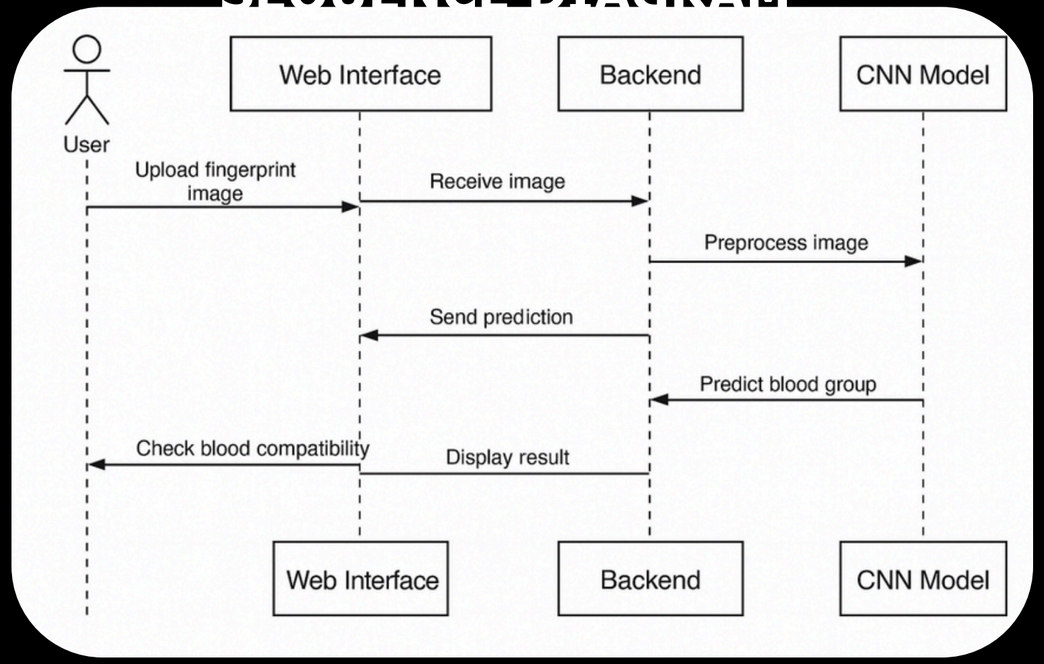
DFD



User uploads fingerprint → Preprocessing → CNN prediction → Result sent back to UI → Log session. Captures step-by-step interaction for real-time prediction.

User uploads fingerprint → Preprocessing (resize, grayscale) → CNN Model (feature extraction & prediction) → Output (blood group + compatibility) → (Optional) Backend (logs & storage). Shows smooth data flow from input to result.

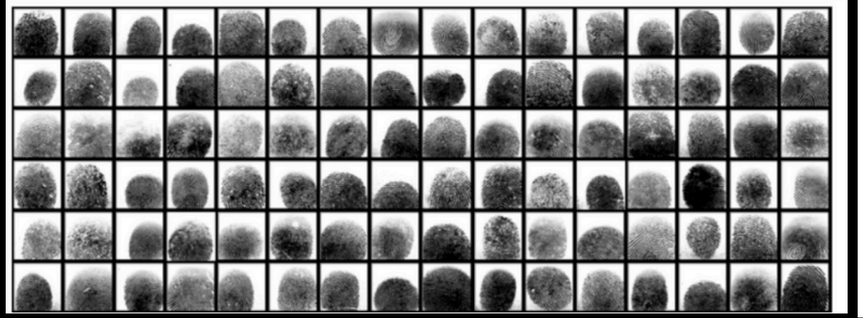
SEQUENCE DIAGRAM



Implementation & Results

```python
def show_batch(dl):
    for images, labels in dl:
        fig, ax = plt.subplots(figsize=(12, 6))
        ax.set_xticks([]); ax.set_yticks([])
        ax.imshow(make_grid(images, nrow=16).permute(1, 2, 0))
        break
show_batch(train_dl)
```

```
C:\Users\rsshi\anaconda3\Lib\site-packages\torch\utils\data\dataloader.py:665: UserWarning: 'pin_memory' argument is set as true but no accelerator is found, then device pinned memory won't be used.
  warnings.warn(warn_msg)
```

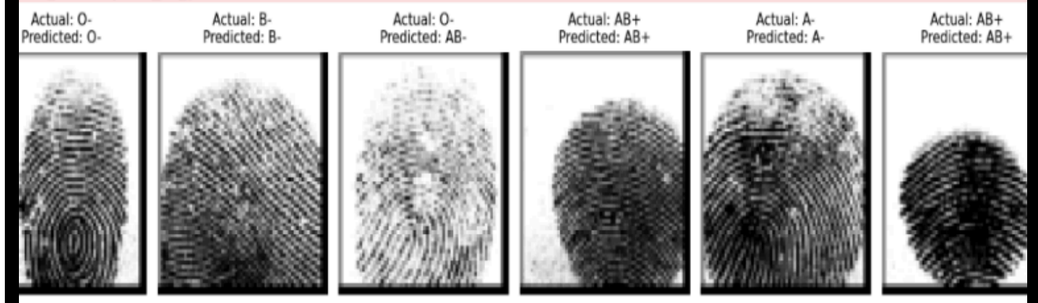


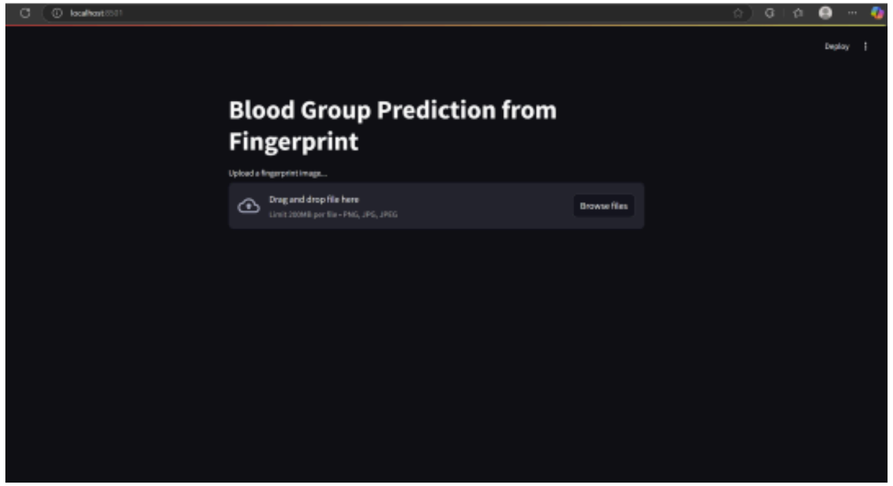
```python
In [12]: device = get_default_device()
         train_dl = DeviceDataLoader(train_dl, device)
         val_dl = DeviceDataLoader(val_dl, device)

         model = FingerprintToBloodGroup()
         model = to_device(model, device)

         num_epochs = 20
         lr = 0.001

         history = fit(num_epochs, lr, model, train_dl, val_dl)
```

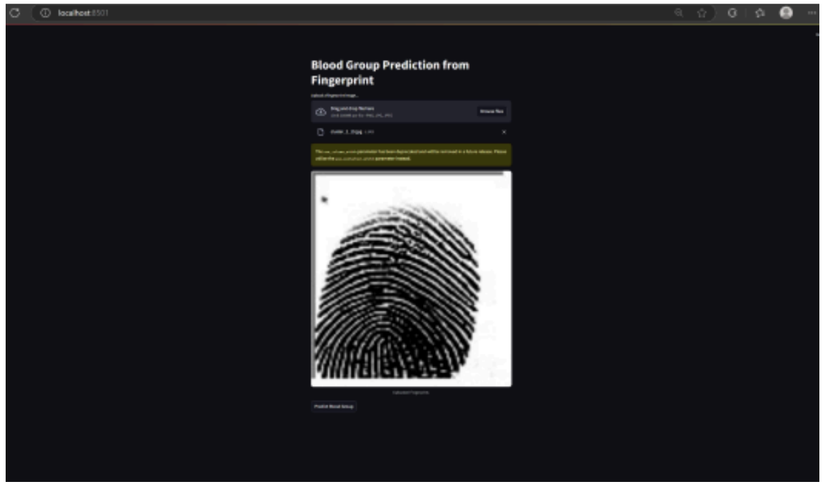
```
Epoch [1], val_loss: 2.0577, val_acc: 0.1759
Epoch [2], val_loss: 1.5725, val_acc: 0.4250
Epoch [3], val_loss: 1.0994, val_acc: 0.5776
Epoch [4], val_loss: 1.0691, val_acc: 0.6022
Epoch [5], val_loss: 0.9045, val_acc: 0.6692
Epoch [6], val_loss: 0.6123, val_acc: 0.7573
Epoch [7], val_loss: 0.6554, val_acc: 0.7605
Epoch [8], val_loss: 0.6500, val_acc: 0.7587
Epoch [9], val_loss: 0.5828, val_acc: 0.7669
Epoch [10], val_loss: 0.5081, val_acc: 0.7995
Epoch [11], val_loss: 0.4675, val_acc: 0.8165
Epoch [12], val_loss: 0.5753, val_acc: 0.7896
Epoch [13], val_loss: 0.4061, val_acc: 0.8436
Epoch [14], val_loss: 0.4676, val_acc: 0.8206
Epoch [15], val_loss: 0.4533, val_acc: 0.8520
Epoch [16], val_loss: 0.4453, val_acc: 0.8285
Epoch [17], val_loss: 0.3688, val_acc: 0.8670
Epoch [18], val_loss: 0.3925, val_acc: 0.8660
Epoch [19], val_loss: 0.4060, val_acc: 0.8653
Epoch [20], val_loss: 0.4827, val_acc: 0.8599
```

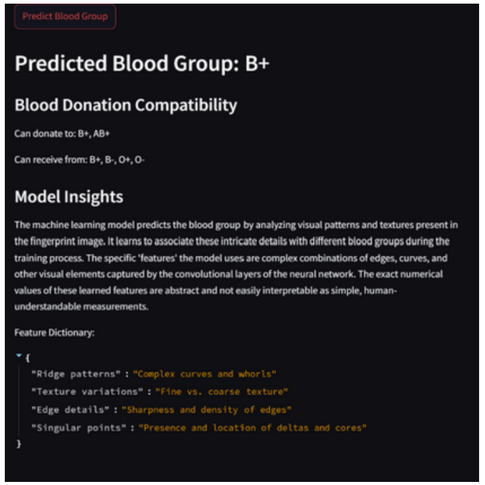
```python
ualize_predictions(model, val_dl)
```

```
ers\rsshi\anaconda3\Lib\site-packages\torch\utils\data\dataloader.py:665: UserWarning: 'pin_memory' argument is set
but no accelerator is found, then device pinned memory won't be used.
nings.warn(warn_msg)
```









Conclusion

- Developed a CNN-based system to predict blood groups from fingerprint images
- Integrated into a Streamlit web app for instant, user-friendly access
- Achieved strong results:
- Accuracy, Precision, Recall, F1-score
- Shows promise for non-invasive, rapid diagnostics

Future Work

- Expand dataset for better accuracy & diversity
- Improve model interpretability (visual explanations)
- Add blockchain integration for data security
- Can later even be used to calculate sugar levels for diabetics
- Leverage smartphone fingerprint tech for real-time predictions
- Deploy in rural clinics & emergency response

References

[1]Jain, A. K., Ross, A., & Nandakumar, K. (2011). Introduction to Biometrics. Springer Science & Business Media.

[2]Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). "ImageNet classification with deep convolutional neural networks". Advances in Neural Information Processing Systems, 25.

[3]Pal, M., & Ghosh, D. (2016). "Dermatoglyphics and ABO blood group: A Review". Asian Journal of Biomedical and Pharmaceutical Sciences, 6(57), 14-18.

[4]Chollet, F. (2017). Deep Learning with Python. Manning Publications.

[5]Simonyan, K., & Zisserman, A. (2014). "Very deep convolutional networks for large-scale image recognition". arXiv preprint arXiv:1409.1556.

[6]OpenCV Developers. (2023). OpenCV-Python Tutorials Documentation. https://docs.opencv.org/

References (cont.)

[7]Streamlit Inc. (2023). Streamlit: The fastest way to build data apps. https://streamlit.io/

[8]He, K., Zhang, X., Ren, S., & Sun, J. (2016). "Deep residual learning for image recognition". IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[9]Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press. 54

[10]Soni, D., & Agarwal, S. (2021). “Blockchain in Healthcare: A Comprehensive Review and Directions for Future Research”. IEEE Access, 9, 138393-138414.

[11]Patel, V. M., et al. (2015). "Visual biometrics: taxonomy, challenges, and future trends." Computer Vision and Image Understanding, 138, 1-20.

[12]Esteva, A., et al. (2017). "Dermatologist-level classification of skin cancer with deep neural networks." Nature, 542(7639), 115–118.

THANK YOU