Face Recognition using PCA + ANN – Interview Cheat Sheet

Workflow:
1. Dataset: 450 images (100×100 pixels, 9 persons)
2. Preprocessing: Resize → Grayscale → Flatten
3. PCA:
   • Compute mean face
   • Extract eigenfaces (principal components)
   • Project images into k-dim feature space (k=50)
4. ANN:
   • Input: PCA features (50)
   • Hidden Layers: Dense + ReLU
   • Output: Softmax (9 classes)
   • Loss: Categorical Cross-Entropy
   • Optimizer: Adam
5. Results:
   • Accuracy: ~50%
   • Saved model (face_ann_model.h5)
   • Visualizations: mean face, eigenfaces
6. Prediction: New image → PCA → ANN → Predicted Person

Key Concepts:
• PCA: Reduces dimensionality, keeps max variance → eigenfaces
• Eigenfaces: Principal components representing max variance in faces
• ANN vs CNN: ANN after PCA, CNN learns features directly
• Why PCA? Avoids training on 10,000 pixels → reduced to 50 features
• Why ANN? Learns non-linear boundaries on PCA features

Q: Why PCA?	**A: To reduce dimensionality, remove noise, and avoid overfitting.**
Q: Why ANN (not CNN)?	A: Simplicity – PCA extracts features. CNN would perform better.
Q: Accuracy is only ~50%. Why?	A: Small dataset, ANN overfitting. CNN or transfer learning would help.
Q: Challenges?	A: Dataset structuring, imports, preventing overfitting, ensuring PCA→ANN compat
Q: Improvements?	A: Use CNN (e.g., VGGFace), data augmentation, larger dataset, real-time deploym

Workflow Diagram:
Image (100x100) → Flatten → PCA (50 comps)
↓
PCA Features
↓
ANN Classifier
↓
Predicted Person (Softmax)