CS7.404: Digital Image Processing
Project Proposal
KNN-Based Jigsaw Puzzle Solver

Farzana S (2024121006), Shaikh Haris Jamal (2023112007)

1 Overview

This project proposes an automated jigsaw puzzle solver using K-Nearest Neighbors (KNN) classification. Square image patches will be reassembled by predicting pairwise compatibility from boundary and texture features, followed by a structured assembly algorithm. The method emphasizes interpretability while remaining computationally feasible for puzzles up to 16×16 pieces.

2 Problem Statement

Reconstructing images from shuffled patches requires:

- Reliable compatibility assessment
- Spatial reasoning for placement
- Handling ambiguous textures
- Enforcing grid consistency

3 Methodology

3.1 Feature Extraction

Boundary features (RGB, gradients, LBPs), patch-level statistics (mean/dominant colors), and optional position embeddings.

3.2 KNN Compatibility

Database of true adjacent pairs used to train KNN ($k = 5$–10). Separate classifiers for horizontal/vertical adjacencies. Compare Euclidean, Manhattan, and cosine distances.

3.3 Puzzle Assembly

Pairwise compatibilities are integrated via:

1. **Corners/Edges:** Identify boundary pieces.
2. **Frame:** Assemble edges with grid constraints.
3. **Greedy Interior:** Place pieces with max compatibility.
4. **Backtracking:** Swap/reassign patches if conflicts arise.
5. **Refinement:** Use graph representation with local optimization.

4 Implementation

- **Libraries:** Python, OpenCV, scikit-learn, NumPy, matplotlib, PIL.
- **Dataset:** 50–100 natural images; puzzles of 8×8 to 16×16.

5 Evaluation

The system will be evaluated using multiple accuracy measures:

- **Direct Accuracy:** Percentage of pieces placed in the correct absolute position.
- **Neighbor Accuracy:** Proportion of correctly matched adjacent pairs.
- **Image Similarity:** Structural Similarity Index (SSIM) between the reconstructed and original image.

6 Timeline

Weeks	Tasks
1–2	Literature review, dataset setup
3–4	Feature extraction implementation
5–6	KNN model development
7–8	Assembly algorithm (greedy + backtracking)
9	Evaluation and analysis
10	Documentation and presentation

7 Challenges & Solutions

- **High-dimensional features:** Apply PCA
- **Slow KNN queries:** Use approximate nearest neighbors
- **Ambiguities:** Enforce grid constraints and refinement

8 Outcomes

A functional solver with $> 70\%$ accuracy on 64-piece puzzles, scalability to 256 pieces, runtime under 5 minutes for 100-piece puzzles, comparison with baselines such as random matching or colour histogram matching, and a well-documented report with comparative analysis.

9 Conclusion

By combining KNN-based compatibility with structured assembly, this project offers a practical and interpretable approach to puzzle solving. The 10-week schedule balances technical depth with achievable milestones.