Paul Devianne & Rizhong Lin

Image Analysis and Pattern Recognition

SQUARE TILING PUZZLE

Group 44

June 2, 2023

Introduction

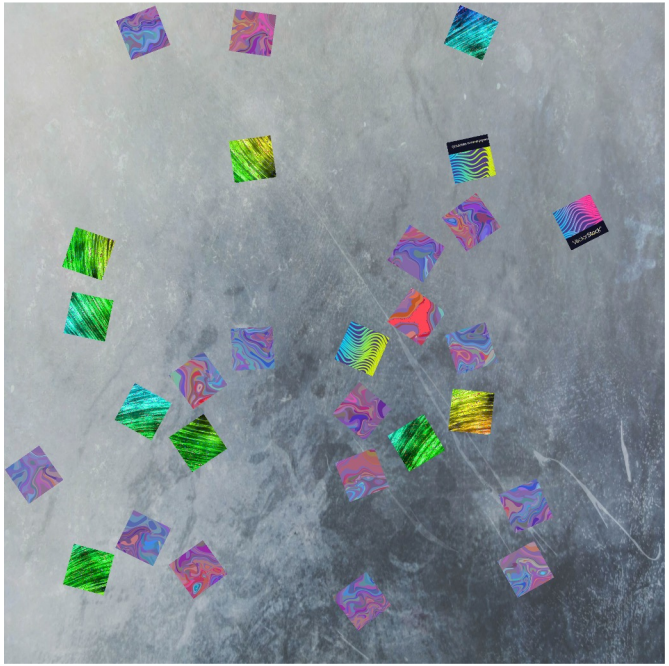
Paul Devianne & Rizhong Lin

Input: An 2000x2000 image with randomly arranged puzzle pieces

Output: Piece masks, feature maps and clustered pieces

Objective:

- Segment puzzle pieces
- Extract & filter different features
- Cluster pieces correctly



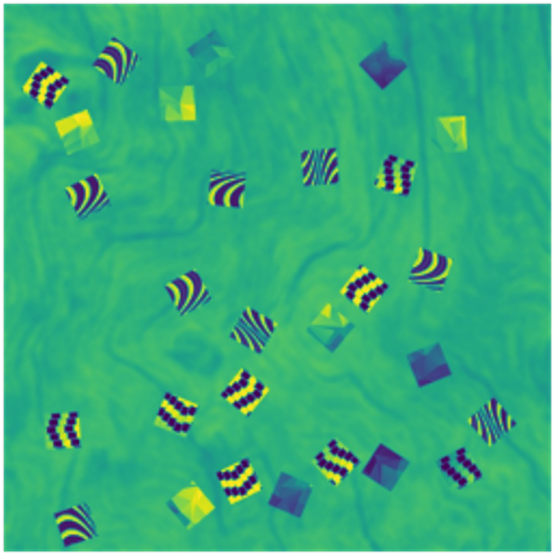
Part 1 - Segmentation

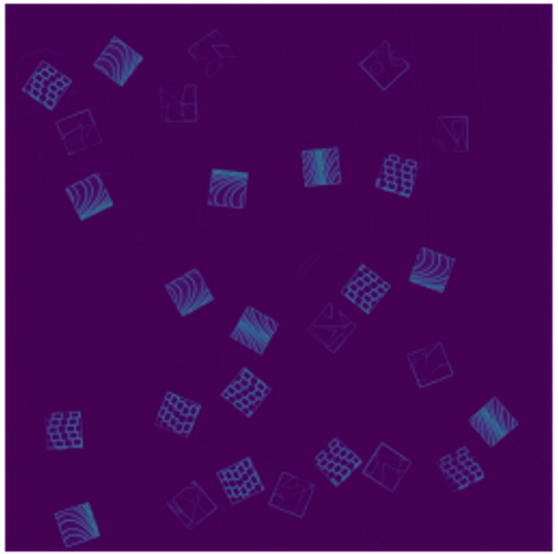
- Applying a Sobel filter
- Identifying contours
- Refining contours
- Creating convex hulls
- Piece extraction

Paul Devianne & Rizhong Lin

Sobel Filter

$$h_1 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \qquad h_2 = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix} \qquad h_3 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

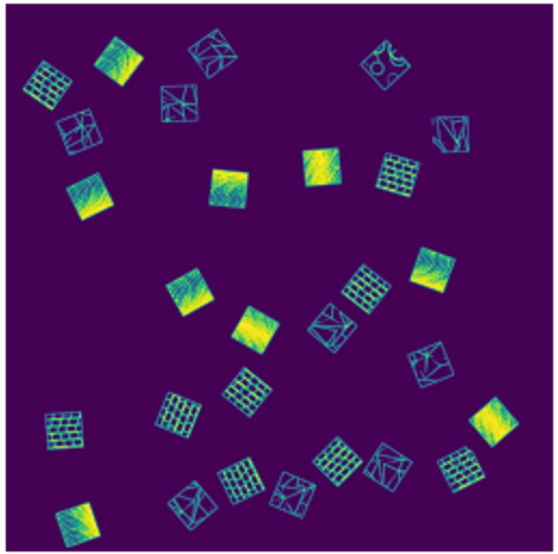


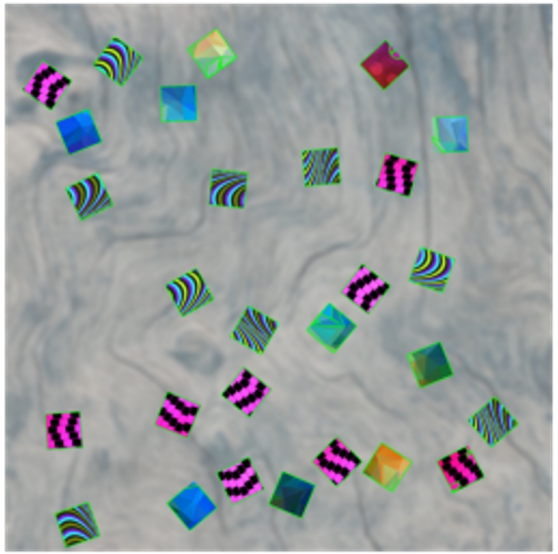


Paul Devianne & Rizhong Lin

Contour Identification

- Setting a threshold to the Sobel-processed image
- cv2.findContours()

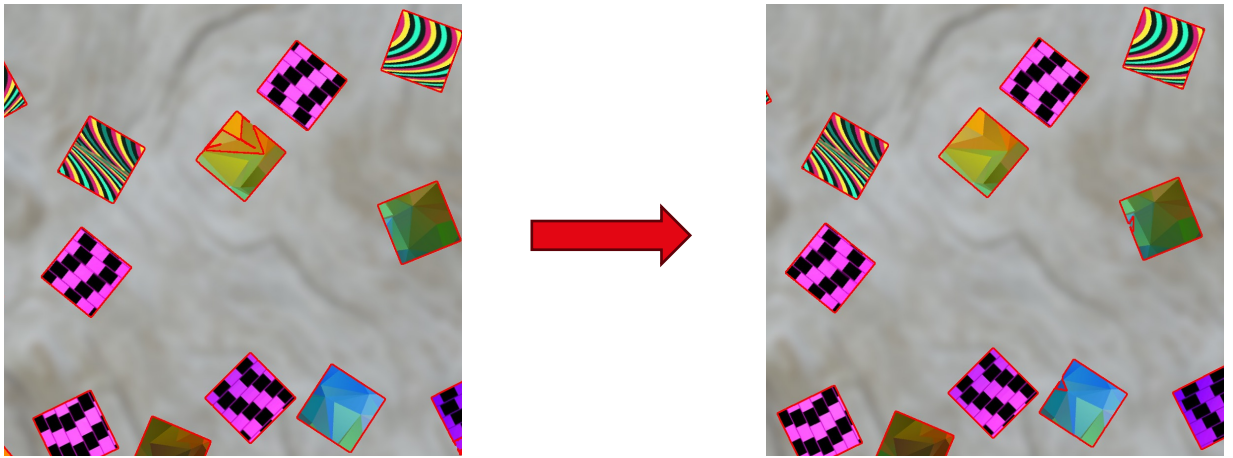




Paul Devianne & Rizhong Lin

Contour Refinement

- Filter:
  - Multiple overlaps, too small/large contours, non-square ones
- Check:
  - “Twin contours” – pairs of contours that have centroids close together

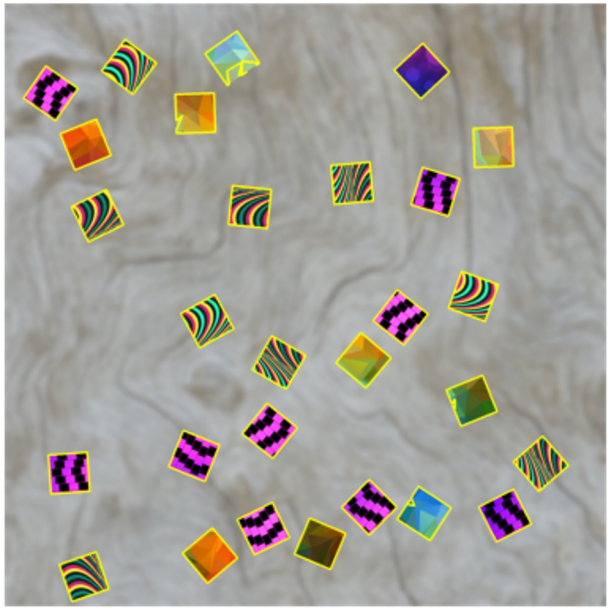


Paul Devianne & Rizhong Lin

Paul Devianne & Rizhong Lin

Convex Hull Contours

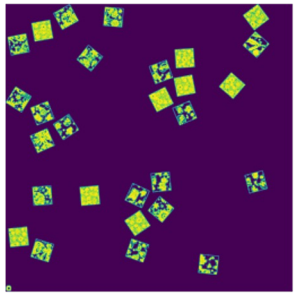
- Convex hull: the smallest convex polygon that contains all the points of the contour
- cv2.convexHull()

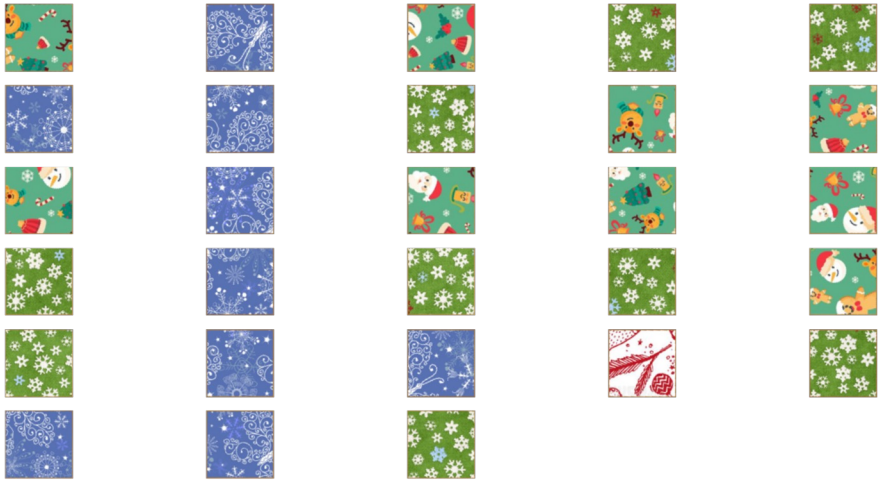


Piece Extraction

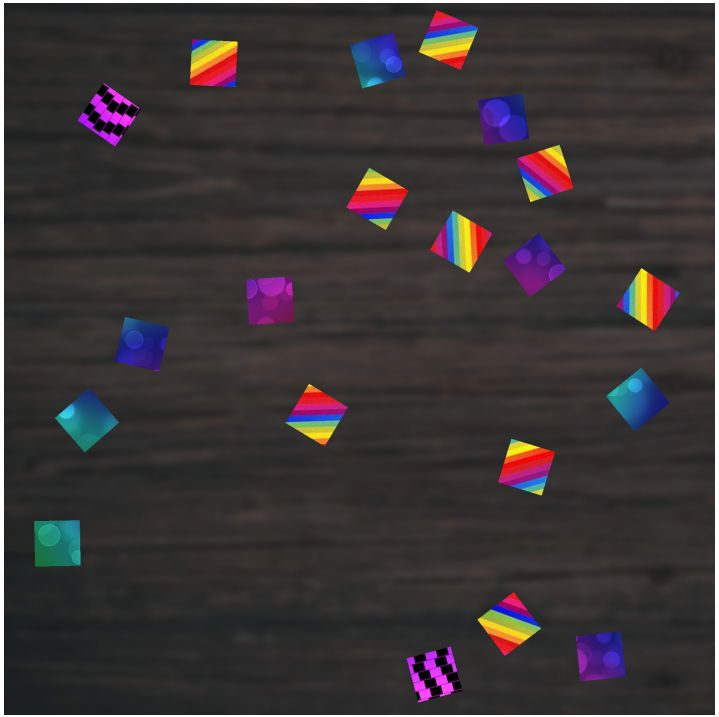
Paul Devianne & Rizhong Lin

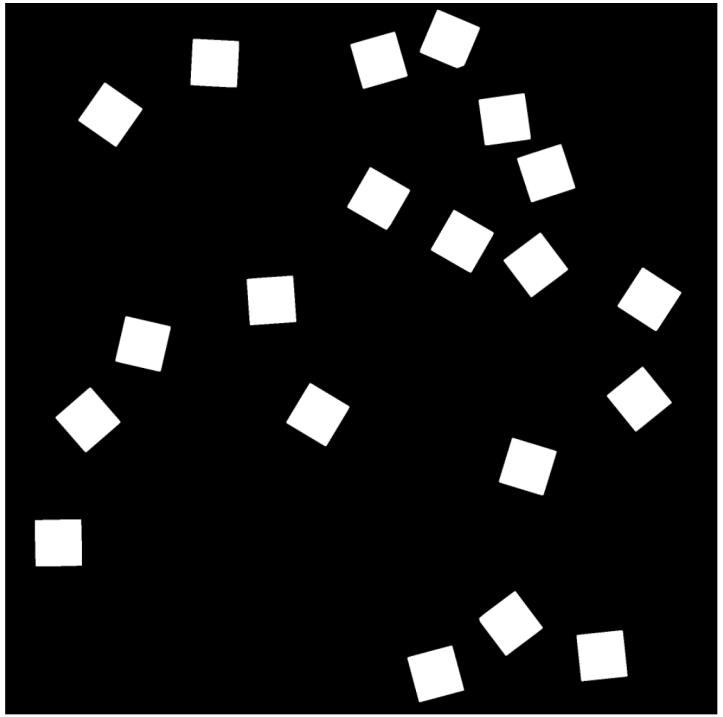
- Calculate the minimum enclosing rectangle of each piece
- Calculate its rotation angle and rotate the whole image
- Calculate the bounding box, and crop!





Piece Mask Result





Paul Devianne & Rizhong Lin

Part 2: Feature Extraction

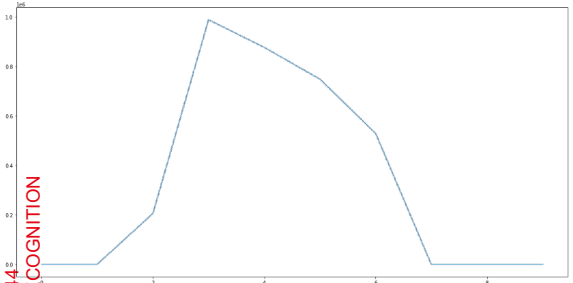
- Gabor filters
- Color histograms

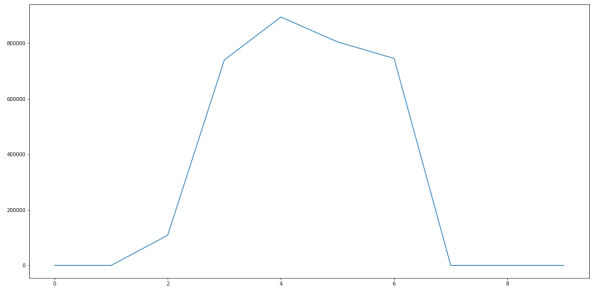
Paul Devianne & Rizhong Lin

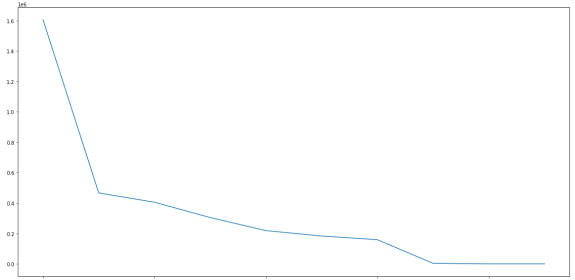
Gabor Filters

Paul Devianne & Rizhong Lin

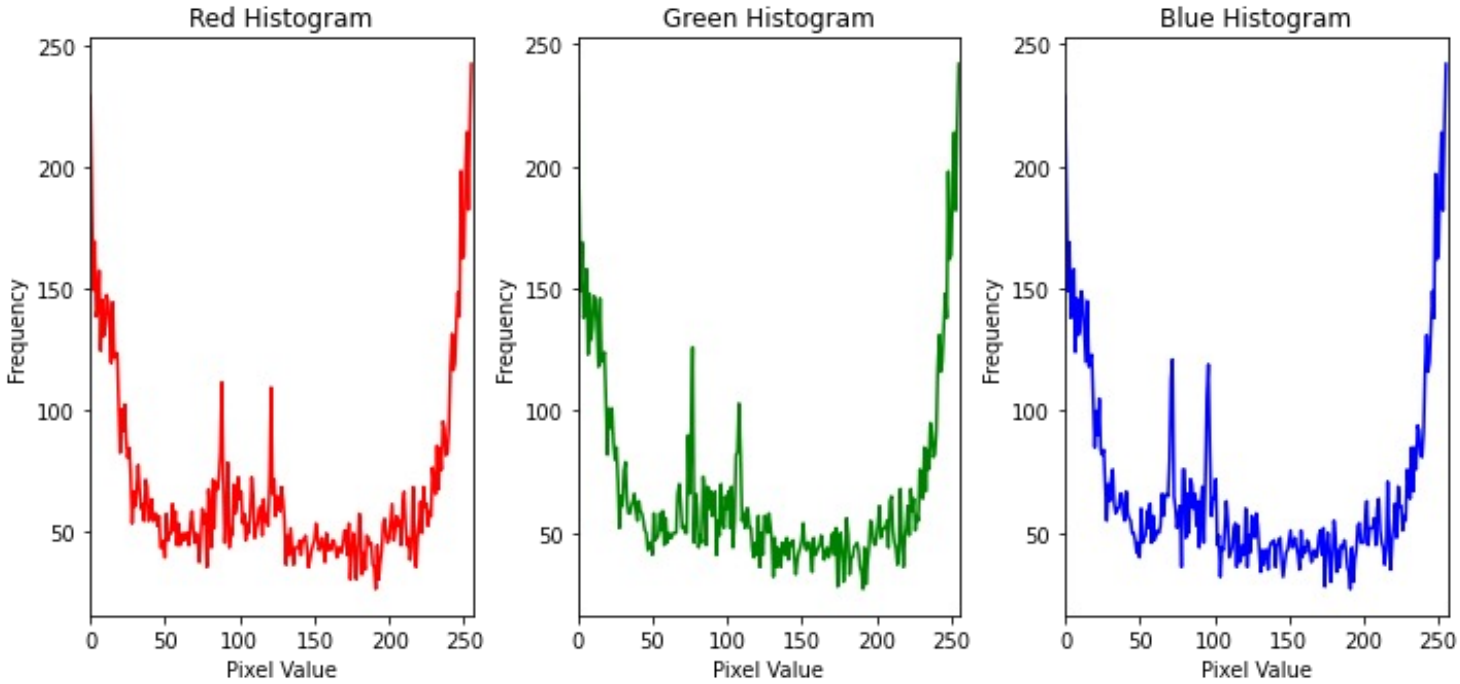
$$gb(x,y) = \exp\left(-\frac{1}{2}\left(\frac{x_\theta^2}{\sigma^2} + \frac{y_\theta^2}{(\Gamma\sigma)^2}\right)\right)\cos\left(\frac{2\pi}{\lambda}x_\theta + \psi\right)$$







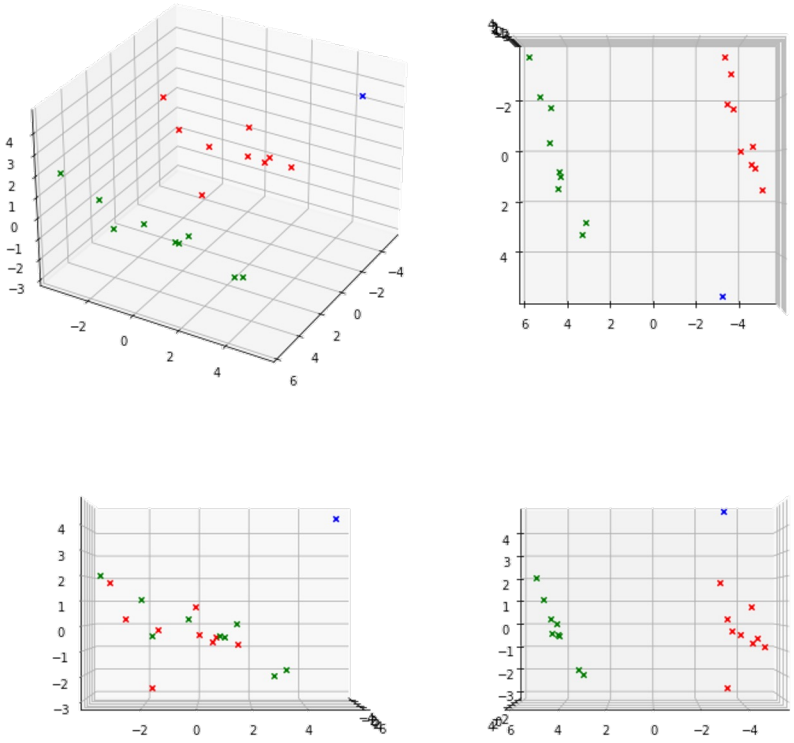
Color Histogram



Paul Devianne & Rizhong Lin

Paul Devianne & Rizhong Lin

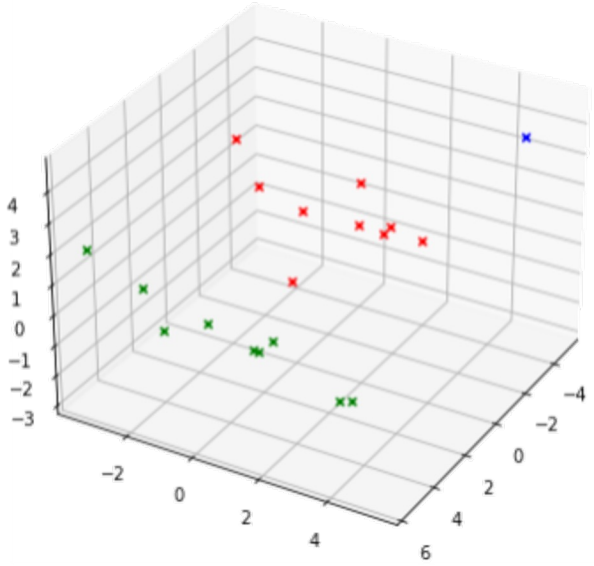
Feature Extraction Result: Principal Component Analysis (PCA)

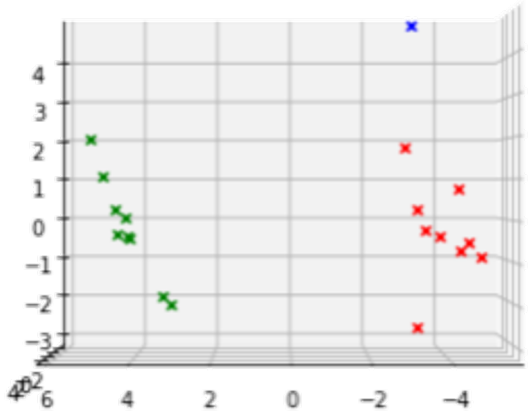


Part 3: Clustering

- KMeans Clustering
- Inertia and Silhouette Scores
- Determining the Optimal Number of Clusters
- Removal of Outliers & Combination of Similar Clusters

KMeans Clustering





Paul Devianne & Rizhong Lin

Inertia and Silhouette Scores

Paul Devianne & Rizhong Lin

- Inertia
  - measures the sum of squared distances of samples to their closest cluster center. We want to minimize this.

$$\sum_{i=1}^{n} \min_{j=1}^{k} |x_i - c_j|^2$$

- Silhouette Score
  - measures how close each sample in one cluster is to the samples in the neighboring clusters. We want to maximize this.

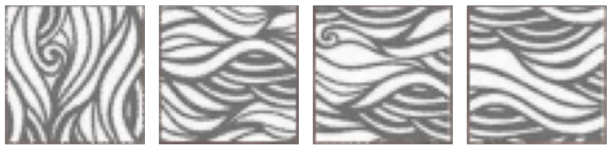
$$s_i = \frac{b_i - a_i}{max(b_i, a_i)} \qquad b_i = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i,j) \qquad a_i = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i,j)$$

Determining the Optimal Number of Clusters

- Normalizing and combining multiple metrics: Silhouette, Inertia
- The optimal number of clusters is then chosen to be the number that maximizes this combined score.

Removal of Outliers & Combination of Similar Clusters

Cluster 1



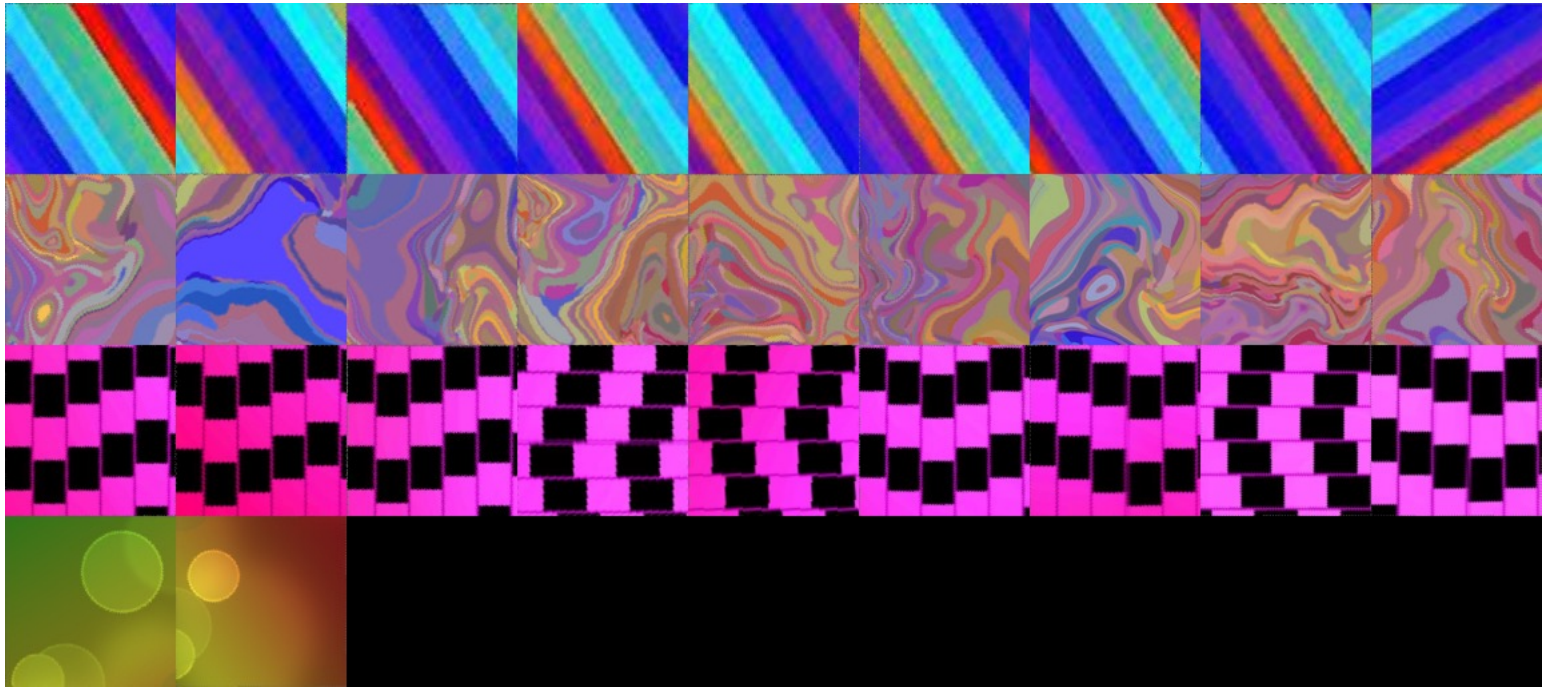
Cluster 3

Paul Devianne & Rizhong Lin

Paul Devianne & Rizhong Lin

Our Results

- 70 ~ 80% accurate



Possible Improvement Directions

- Applying more preprocessing steps before the feature extraction
- Feature selection?
- Applying more filters to extract more efficient and informative features

Paul Devianne & Rizhong Lin