Image Analysis and Pattern Recognition

02.06.2023

Final Project

Puzzle solving

Group 45

Erik Kvikne, Elias William & Sami Laubo

Introduction

Content:

- Image segmentation
- Feature extraction
- K-means clustering
- Solving the puzzle
- Conclusion

Requirements

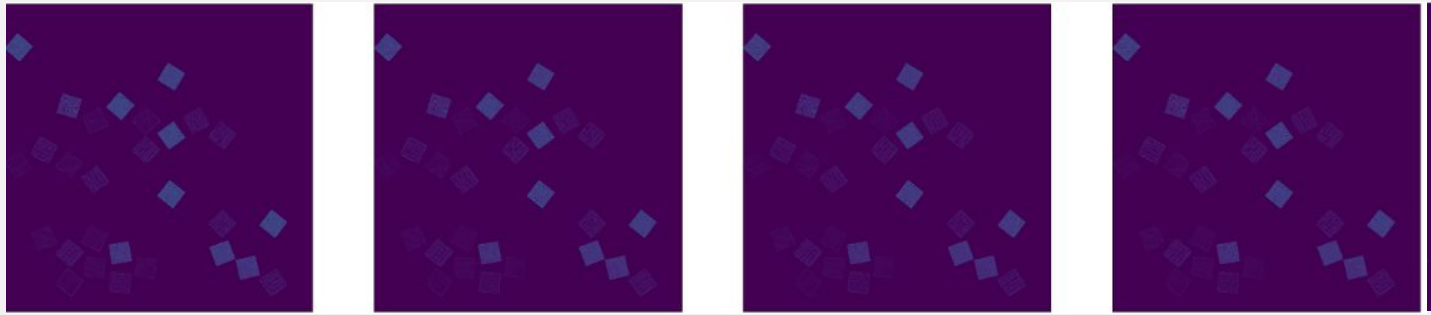
PIL
numpy
typing
matplotlib
skimage
sklearn
open-cv
scipy
skimage

Image segmentation

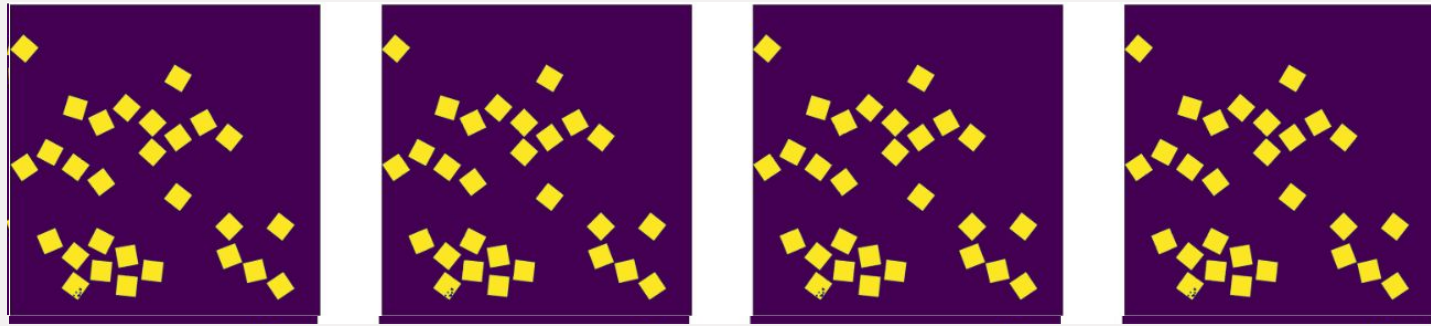
- Laplacian
- Histogram as backup
- 1 second per image

Laplacian method

Laplacian



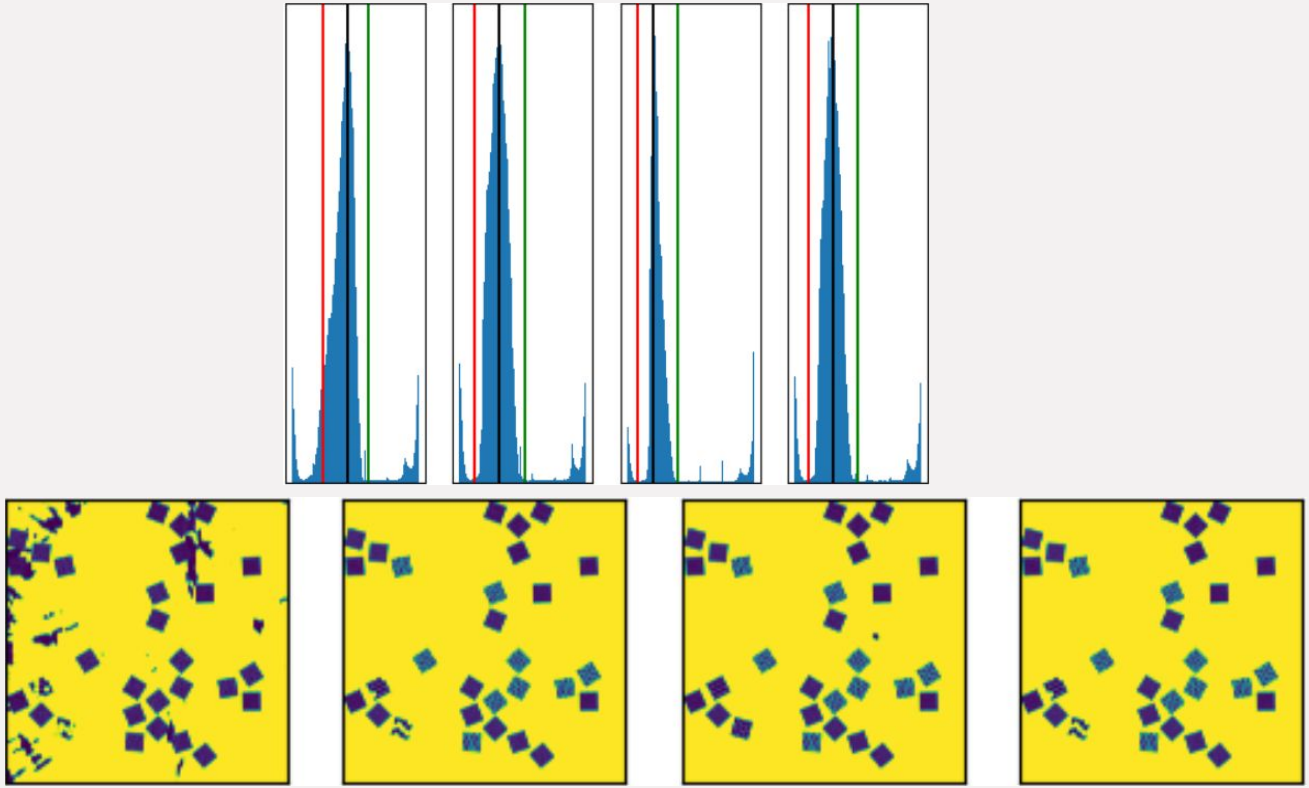
Threshold
&
Morphological
Closing



Histogram method

Histogram

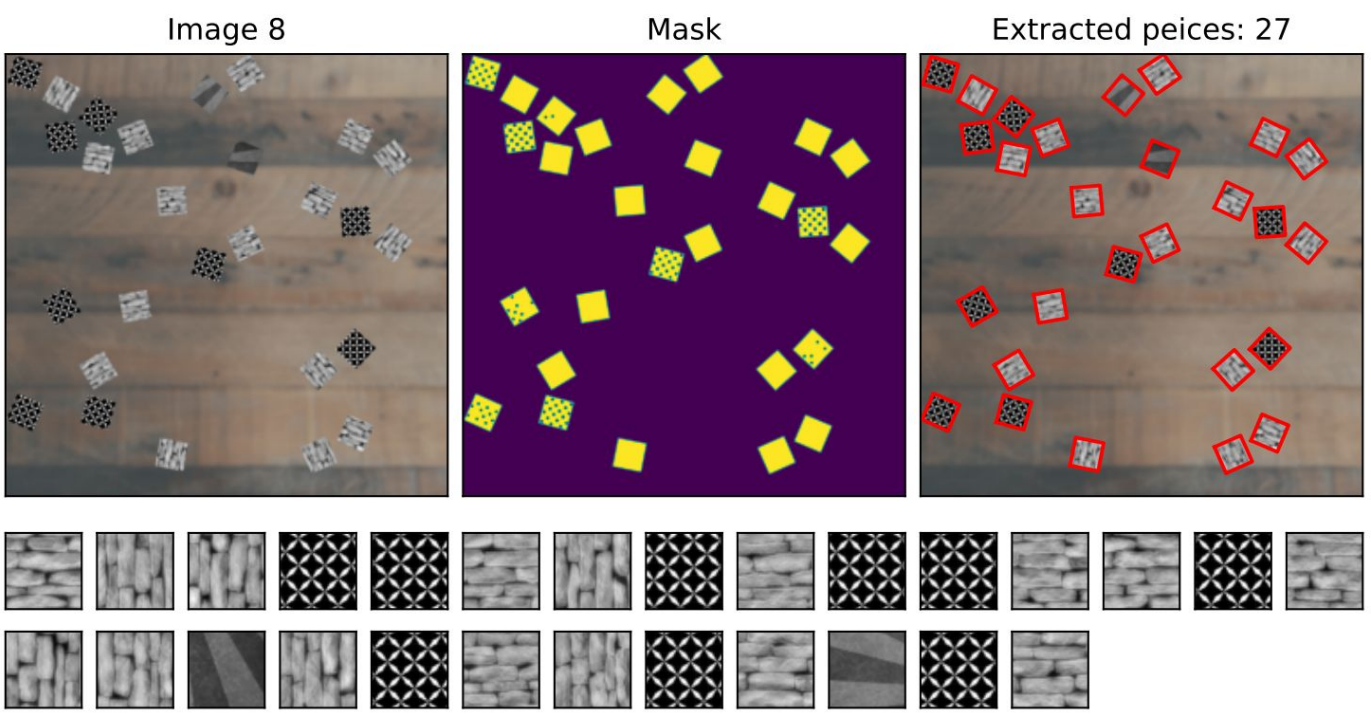
High & Low
Thresholds
around main peak



Find boxes and extract

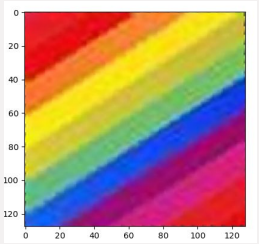
1. Find contours
2. cv2.minAreaRect -> Boxes
3. Replace bad boxes with histogram method
4. Extract image inside boxes

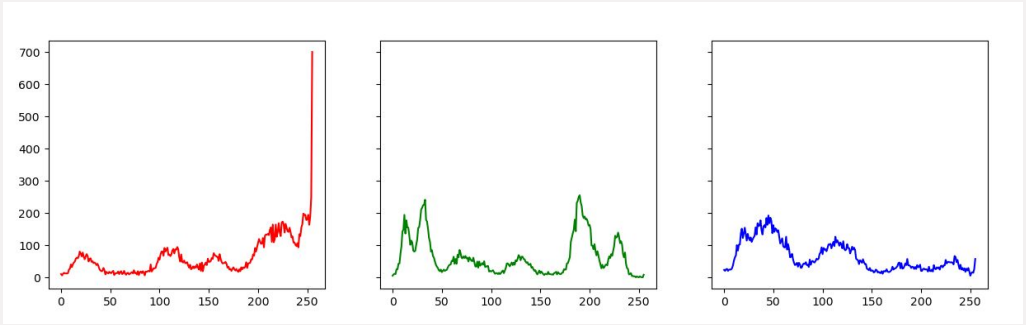
Final outputs



Feature extraction

- Idea: Gather many features and use Principal Component Analysis (PCA)
- Color histogram :



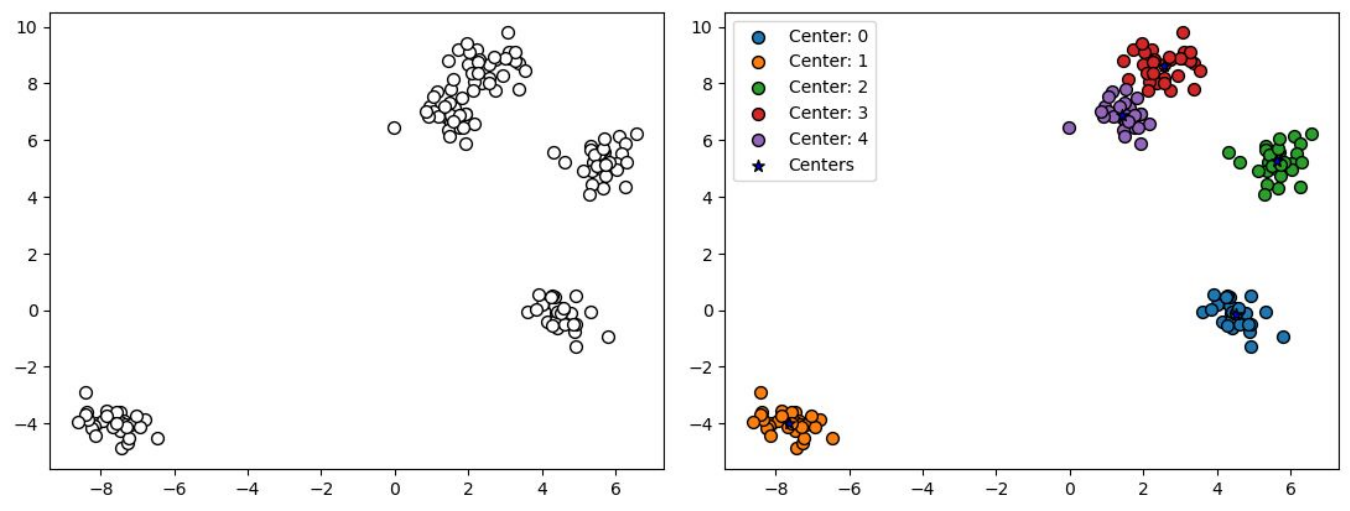


- Gabor filters:

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

K-means clustering

- ISODATA inspired
- Split clusters with large variance
- Merge clusters with nearby centers



Example run

- 150 samples
- K = 10
- max_iter = 100
- other hyperparameters are important
- convergence in 5 iterations

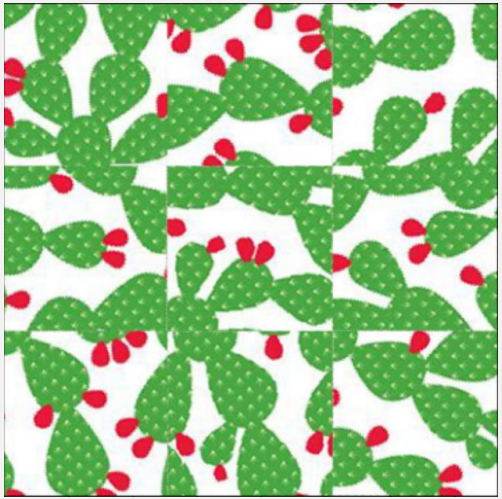
K-means clustering

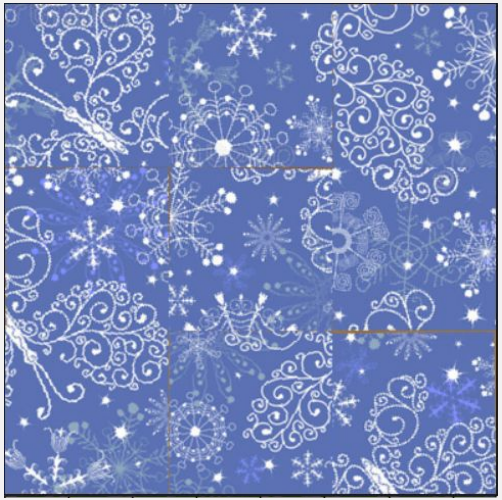
The algorithm

- Initialize the centers randomly
- Until convergence
  - Assign all samples to the closest center
  - Recompute centers
  - Merge nearby centers
  - Split clusters with high variance
  - Check if old centers are the same

Also, we retry the algorithm on long outliers lists

Solving the puzzle





Conclusion

Room for improvement:

- Better features, neural network?
- More experiments with K-means hyperparameters
- Detect non plausible puzzle sizes
- Better jigsaw edge features