Rubik‘s Cube Solver

MUSTAFA MAHMOUD 01131919

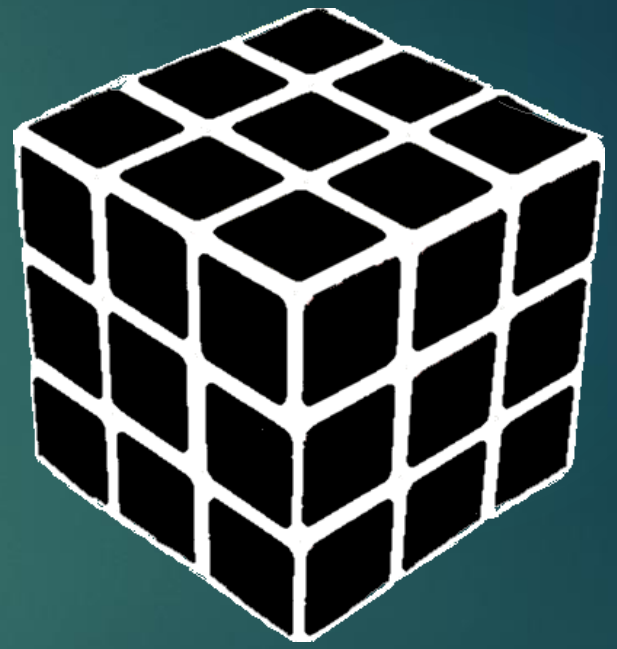
MARTI ZACH 01430672

Overview

- Goal
  - Reconstruct constellation of Rubik‘s Cube
  - Appropriate representation
  - Feed into solver (or at least verify correctness)
- Input
  - Ordinary webcam video stream or input video
- Tools
  - Python
  - Image Processing and Computer Vision libraries (OpenCV)

Object Models

- Very explicit and well defined problem
- 1 cube, 6 faces, 9 facelets, 4 corners



- Top Down

- Bottom Up

Finding features

- General envisioned pipeline
  - Noise Filtering, Grayscale Conversion, Edge Detector
  - Discriminate contours using different characteristics
  - Group facelets to faces, interpolate missing values
- Alternatives
  - Region or point based segmentation
  - Implicit modeling
  - Top down approach
  - Hough transform, …

Edge based approach

Problems

- Finding Contours
  - Reflections, occlusions
  - Discriminating between wanted and unwanted contours
- Grouping Facelets
  - Stable mechanisms very hard to find (centroids, relative or absolute angles, line groupings)
- Dealing with uncertainty
  - New result contradicting the other ones
- Filter Parameters
  - Very much dependent on environment

In Depth: Finding,discriminating and grouping contours

- Finding
  - All of them can never reliably be found
  - Hugely dependent of filter parameters (especially of edge detector)
- Discriminating
  - Characteristics of all facelets
    - Same size, quadratic, uniform intensity values inside, not too big, not too small
- Grouping
  - Characteristics of facelets on same face
    - Similar angles, pairwise similar alignment of lines
  - Projective geometry makes this a hard task