Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

COMPUTER VISION PROJECT

AUTOMATIC RUBIK’S CUBE DETECTION AND RECOGNITION

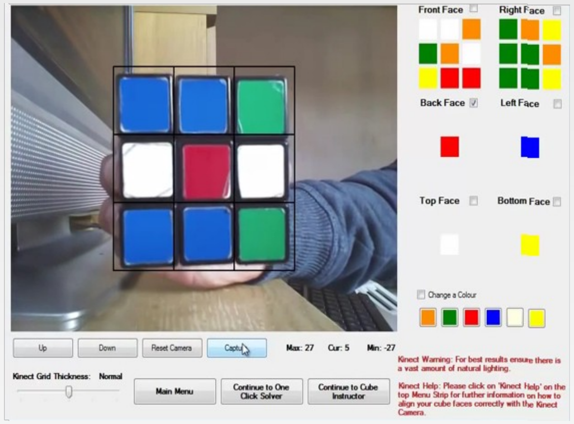
BY
Jeyaprakash Rajagopal
15-Jan-2015

ADVISORS
Prof. Dr.-Ing. Rainer Herpers

Introduction

- To develop a state detector for Rubik’s cube without manual configuration.
- Objective: to detect and recognize 3 faces of the cube over single face at a time.
- Modules
  - Rubik' cube detection
  - Surface rotation
  - Color cell recognition

Problem statement

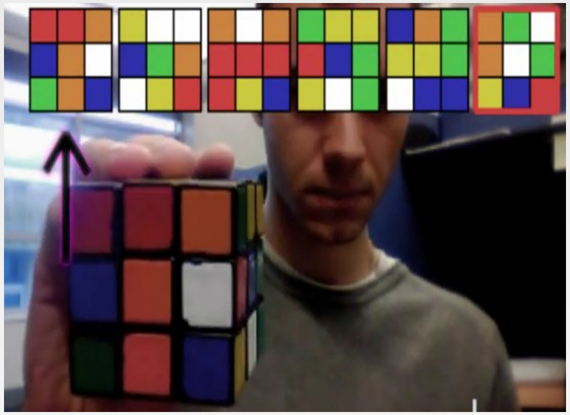


Max: 27 Cur: 5 Min: -27

Kinect Grid Thickness: Normal

Kinect Warning: For best results ensure there is a vast amount of natural lighting.

Kinect Help: Please click on 'Kinect Help' on the top Menu Strip for further information on how to align your cube faces correctly with the Kinect Camera.



A lot of solvers exist

* Existing approaches use feature extraction methods such as

- template matching
- robust fitting methods.

* Pose prediction based on Kalman filter.

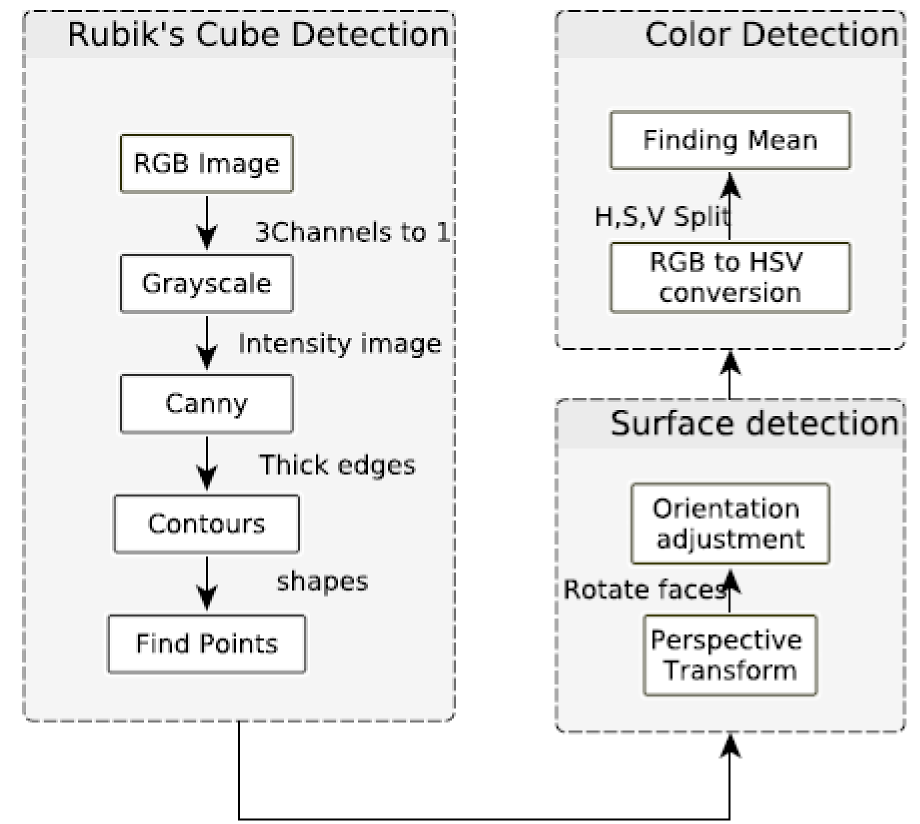
* Not many approaches detecting 3 faces of the cube.

Challenges

To automate Rubik’s cube detection and recognition towards

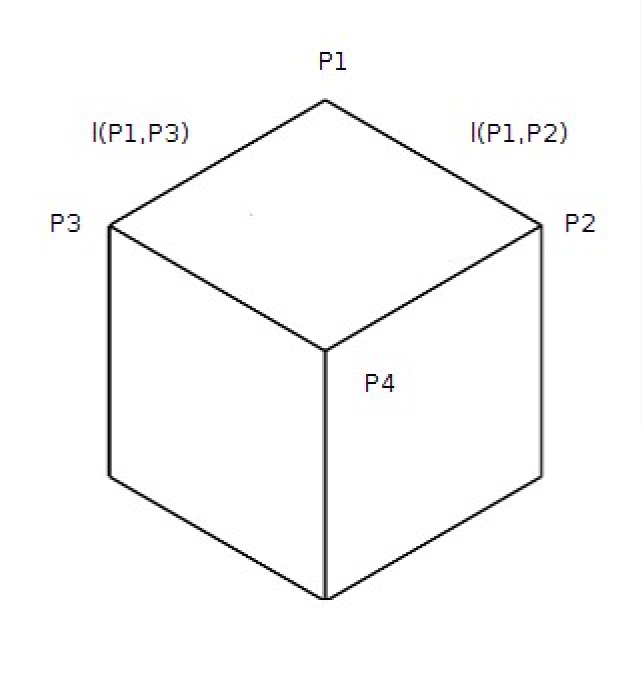
- Varying illumination conditions
- Non-standard sticker color schemes
- Mixture of different cube colors.
- Finding the hidden point of the cube when contours applied.

Rubik's Cube Detection



Rubik's Cube Detection Continued...

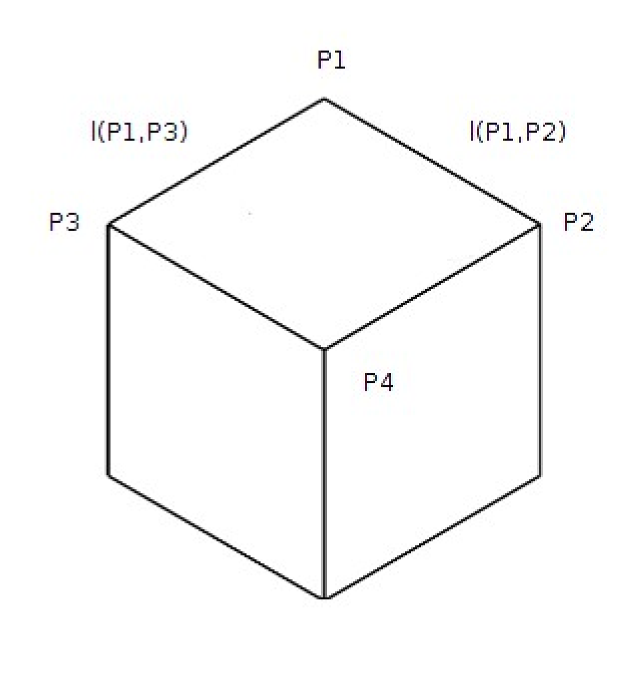
- Knowing P1, P2, P3, Find P4?
- Step 1: Finding Slopes between 3 points
  - M1 = (P1.y-P2.y)/(P1.x-P2.x)
  - M2 = (P1.y-P3.y)/(P1.x-P3.x)



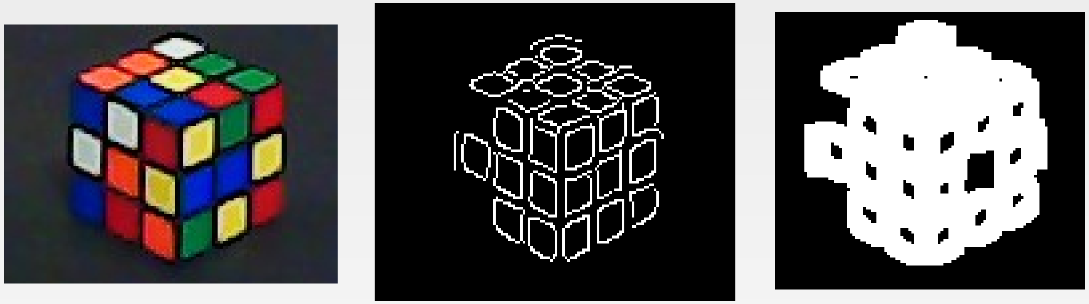
Rubik's Cube Detection Continued...

- Step 2:Finding the (x,y) of P4
  - P4.x=((P2.y-M2*P2.x)-(P3.y-M1*P3.x))/(M1-M2)
  - Knowing P4.x, P4.y can be calculated
  - P4.y=P2.y-(M2*P2.x)+(M2.P4.x)

where P1, P2, P3, P4 are the points Point1, Point2, Point3, Point4 respectively. M1, M2 represents slopes



Rubik's Cube Detection Continued...



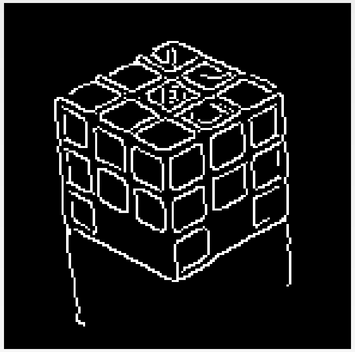
The problem on either sides of the cube in detecting visible sides.

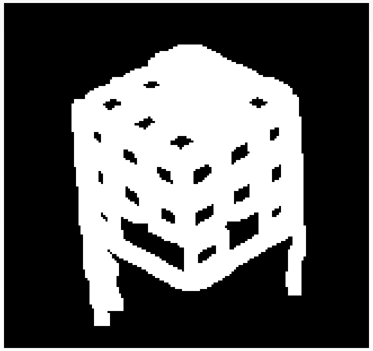
- Bad lighting conditions
  - Black background suffers in finding the grids of blue.
  - Difficultly in extracting sticker colors.

Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

Rubik's Cube Detection Continued...

- White background during edge detection
  - Approximations lead to false detection
    - Due to sensor errors.
  - Possible Solution
    - Multi level edge detection can be used.





Surface Detection

- Separate three sides of the cube using end points.
- Left, right and top faces are separated.
- Rotate surface using perspective transformation.
  - getPerspectiveTransform() is used from OpenCV library.
- Each face is processed to take care of the missing pixels.
- Disadvantage:
  - Loss of pixel information

Color Detection

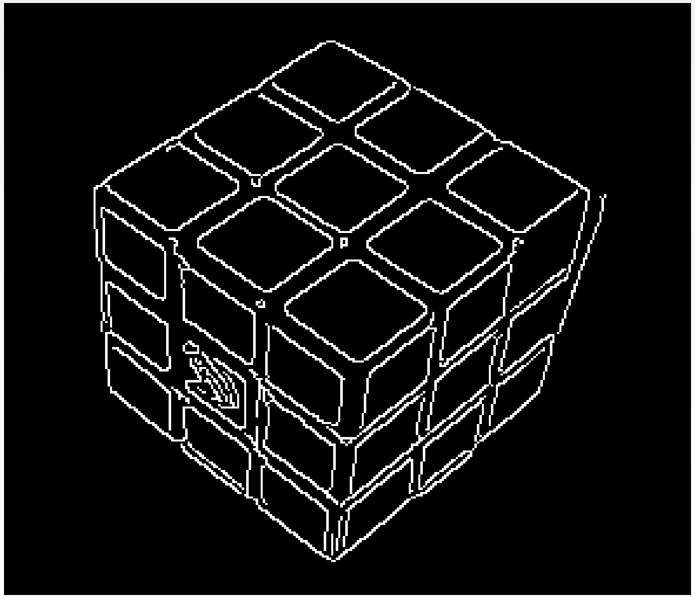
- HSV separates color components with different intensities.
- Acquires robustness in varying illumination conditions.
- To extract colors, 3x3 matrix created in reference frame.
- Averaging region of pixels
  - Mean = sum(components) / no. of pixels
  - Components are Hue and Saturation

Color Detection Continued...

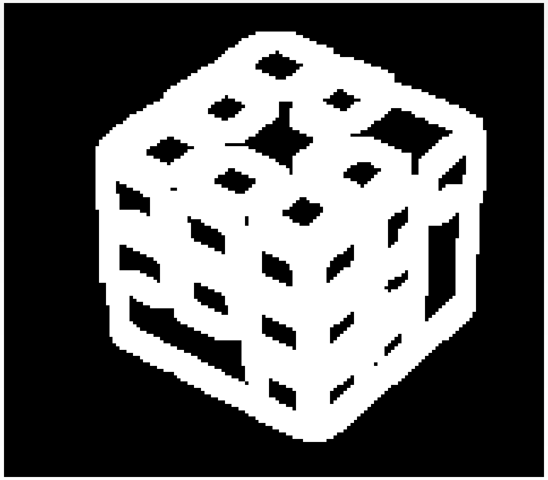
- Mean is compared against the color ranges.
- For example, range of Hue differs from
  - Red - 160 to 179
  - Orange - 0 to 17
  - Green - 50 to 75
  - Yellow - 20 to 30
  - White - 0 to 180
  - Blue - 110 to 120

Results of Detection

- Step 1: After Applying Canny Edge Detector

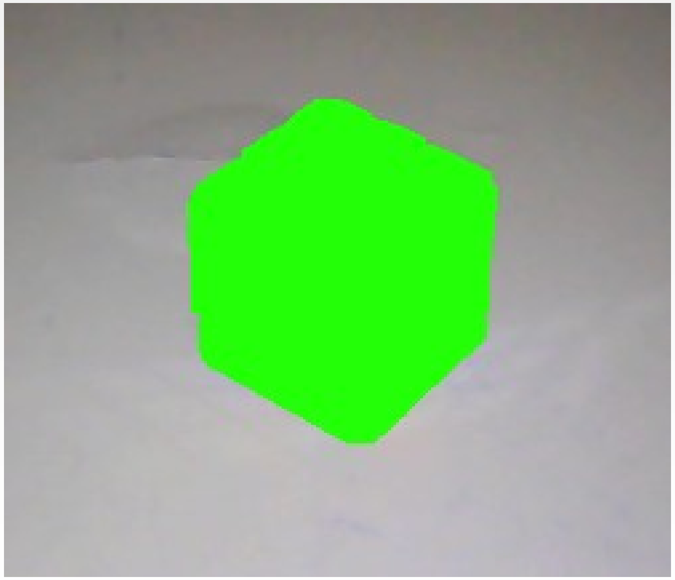


- Step 2: Dilating the result image of Canny.

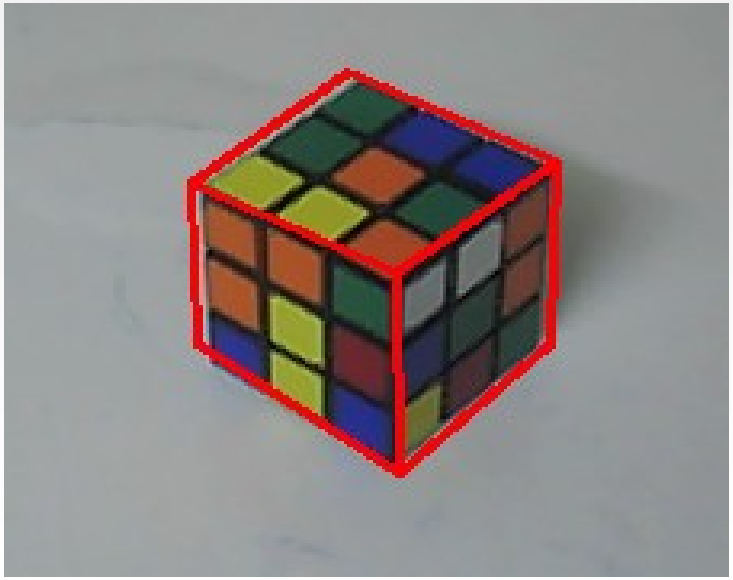


Results of Detection Continued...

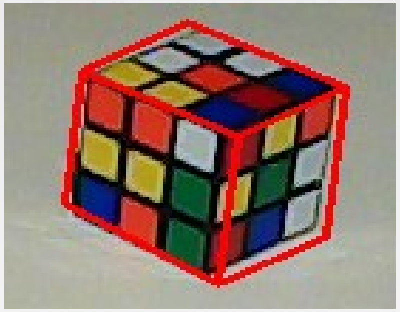
- Step 3: Applying contours

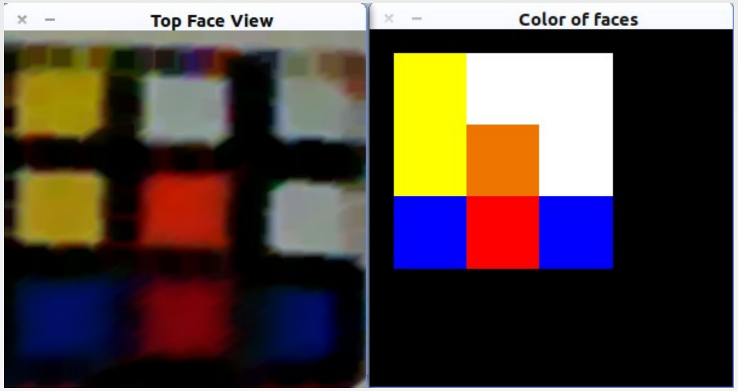


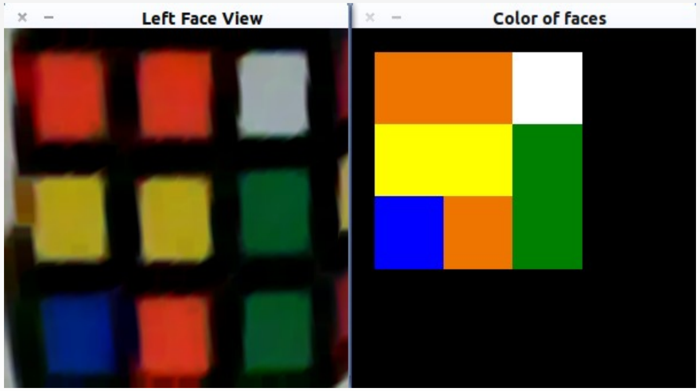
- Step 4: Detecting the Cube’s 3 surfaces

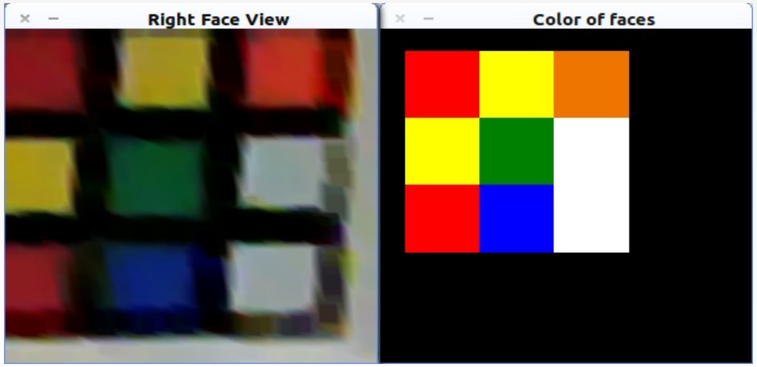


Color Recognition and Surface Transform.



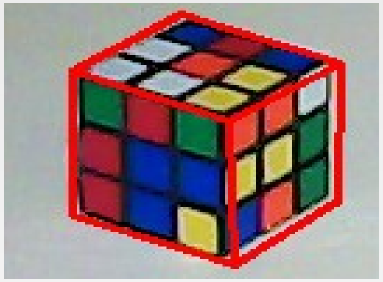


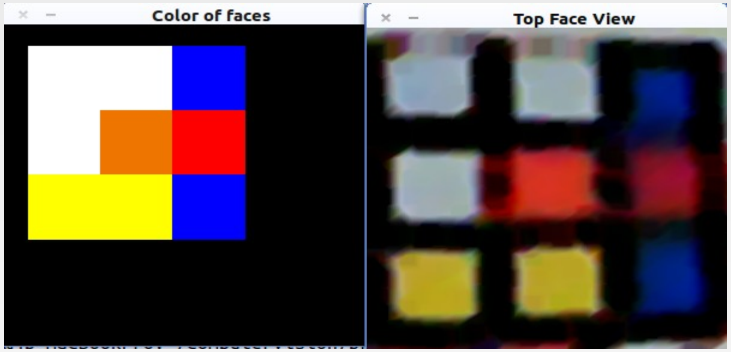


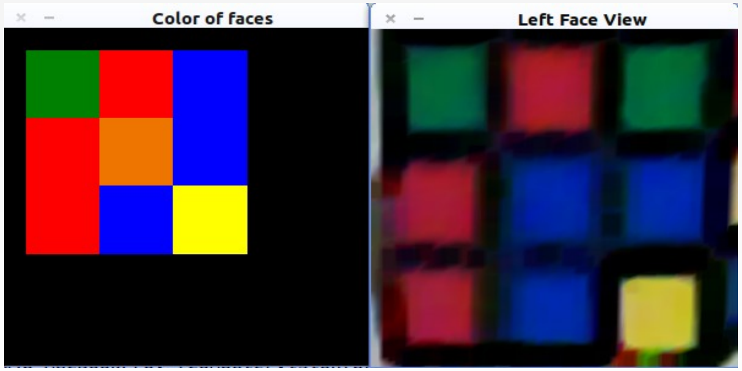


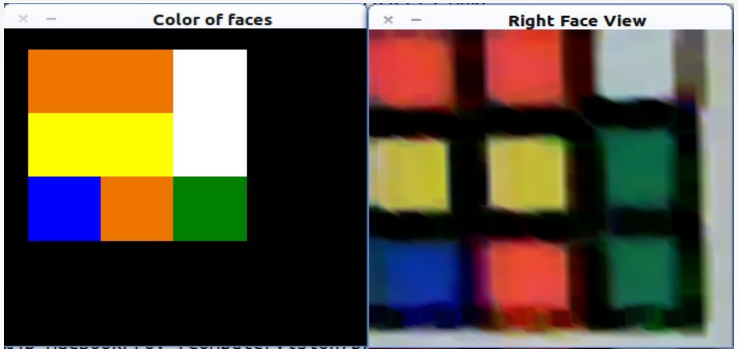
Only one output window for all faces. Top, left, right → by t,l,r keys respectively

Color Recognition issues

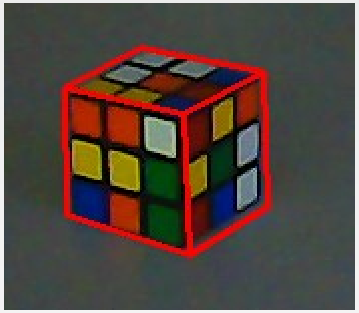


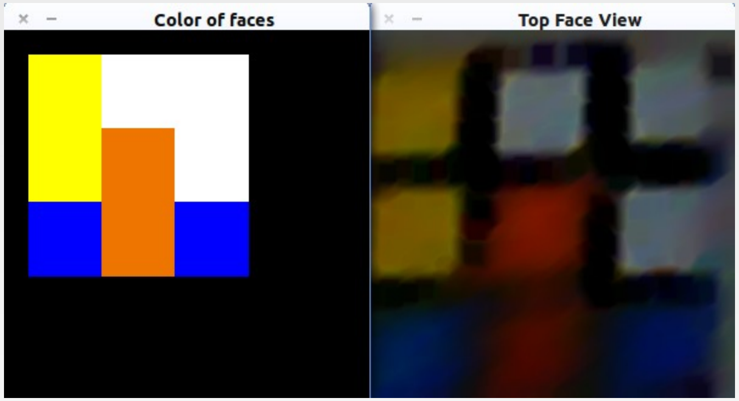


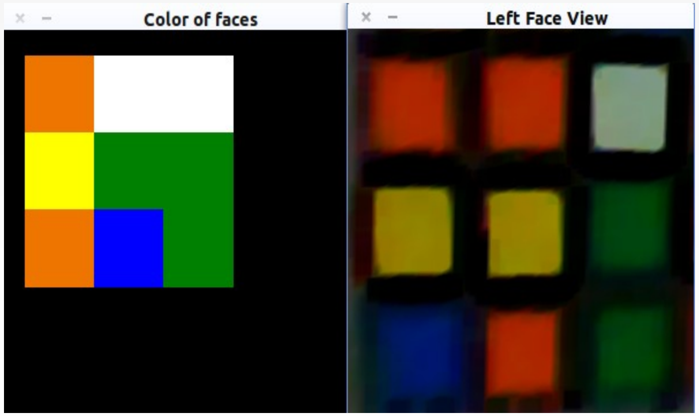


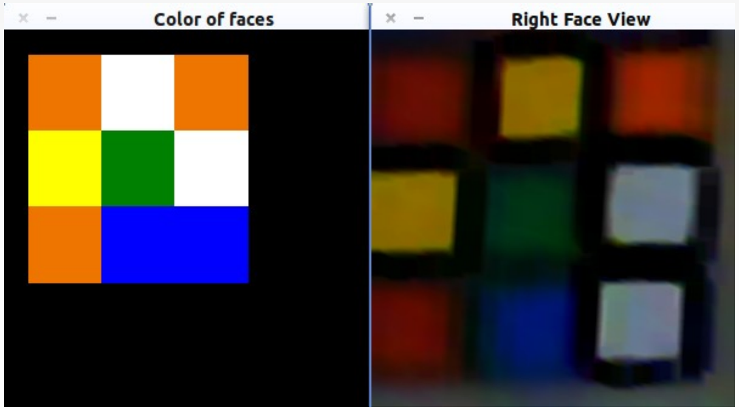


Color Recognition issues with flashlight

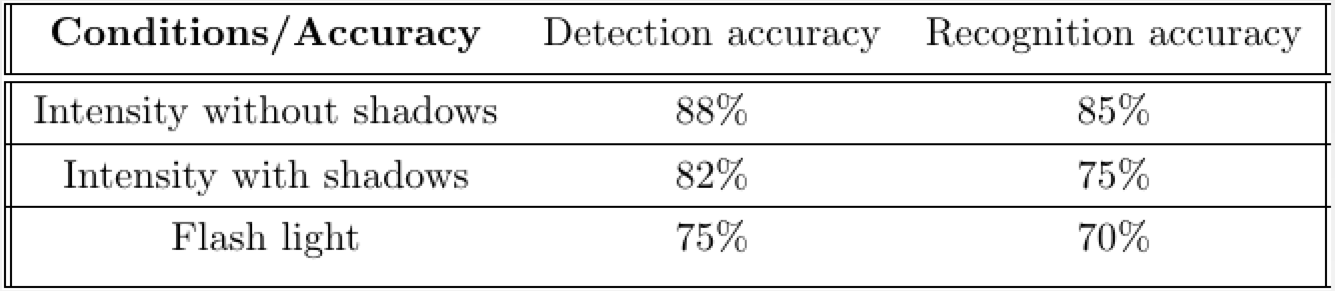








Color Recognition Final Result



Conditions/Accuracy	Detection accuracy	Recognition accuracy
Intensity without shadows	88%	85%
Intensity with shadows	82%	75%
Flash light	75%	70%

- Detection accuracy is based on number of overlapping edges
- Recognition accuracy is based on
  Accuracy = No of colors correctly identified / total cells.
- Totally 15 set of experiments conducted for evaluation.
- Recognition accuracy degrades with a change in illumination color.

Discussion

- How to analyze a problem in computer vision. Pixel-wise operations during color recognition.
- Loss of pixel information during the perspective transformation.
- Difficulty faced in finding missing point. Working with mathematical models enabled it possible.

Future Work

- Approximation improvement.
  - Multi level edge detection capabilities
- Improved color detection with machine learning techniques.
  - Due to varying lighting conditions.
  - Similarity in color values e.g., orange and red.
  - Histogram based and particle filtering based approaches.

References

[1] Andrej Karpathy, "Extracting sticker colors on Rubiks cube, Semester project",\newline github.com/kopernicky/rubik/tree/master/materials

[2] Kasprzak W, lodzimierz and Szynkiewicz, Wojciech and Czajka, Lukasz, "Rubiks Cube Reconstruction from Single View for Service Robots", Machine Graphics and Vision International Journal, Feb 2006.

[3] O Faugeraus, "Three dimensional computer vision, a geometric viewpoint", The MIT press, Nov 1993.

[4] L. Venturino et al. "Improving 3D scene reconstruction through the application of geometric constraints". Workshop on Systems, Signals and Image Processing , Poznan University of Technology Press, Sept 2004

[5] Richard Szeliski, "Computer Vision: Algorithms and Applications", Springer Sept 2010.

[6] Peng Chang, Krumm J, "Object recognition with color concurrence histograms", Computer Vision and Pattern Recognition, IEEE Computer Society Conference on June 1999.

[7] R Maini, H Agarwal, "Study and comparison of various image edge detection techniques", International Journal of Image processing, 2009.

[8] Katja Nummiaro, Esther Koller-Meier, Luc Van Goo, "An adaptive color-based particle filter", Image and Vision Computing, Jan 2003.

QUESTIONS???