01/28/2016

ESE650 Project 1: Color Segmentation

Nischal K N
nischal@seas.upenn.edu

Abstract
Detection of unique objects such as cones, barrels, signs, etc is a crucial part of robotic system be it autonomous cars or humanoids. In this project an approach to detect barrels based on their color and physical dimensions has been discussed. Gaussian mixture models(GMM) are used to model the color of the barrel including variations at different lighting conditions and is used to perform color segmentation. A series of morphological operations and filtering based on dimensions of the segments are performed to determine the barrel and overcome occlusion. The model is created from a training set and evaluated on a separate test set. The performance of the test set is shown in this report.

Contents

1. Introduction

In this project red colored barrels are detected from static high resolution images by color segmentation using Gaussian mixture model(GMM). First the model of the red colored pixels is generated using a multi-variance GMM on a custom color space designed to highlight the red regions of the barrel. To cluster the pixels into gaussians, EM optimization is used. Additionally the parameters such as height,width and distance from camera of the barrel on the training set is recorded and modeled.

To detect the barrels in a given image, the probability of each pixel belonging to a Gaussian is determined. A likelihood map of all the pixels is generated. From this a binary image is created by thresholding the probabilities. A series of morphological operations are performed to remove noise and small stray clusters. Finally from the model of heights and widths of barrel generated during training, the barrel is detected and its distance is estimated.

2. Methods

The process was divided into 2 parts, viz., training which was performed offline to generate the model and then this model was used on testing images to determine the centers and distances of the barrel from the camera. The block diagram of the entire process is shown in Fig. 1.

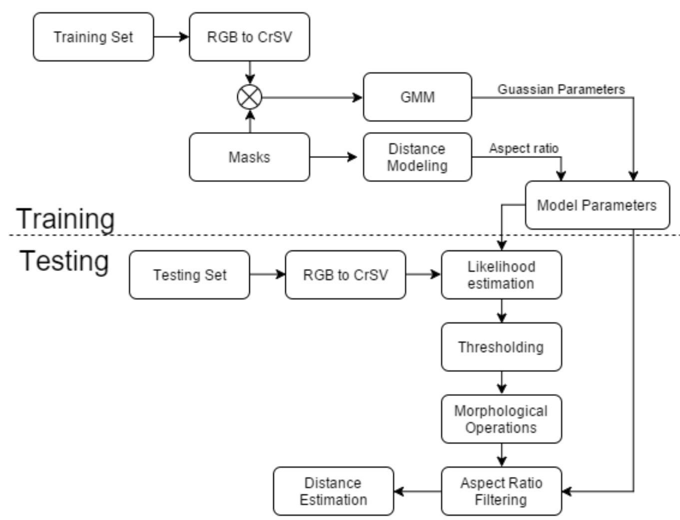


Figure 1. Block diagram

The training process involved generation of masks to separate the barrel pixels form rest of the image for modeling. This was done using roipoly function in matlab.

2.1 Color Space

Choosing a particular color space was extremely important because of wide variation in the illumination of the barrels from outdoors to indoors to dark rooms. Choosing a right color space makes the system robust to illumination changes.

First RGB color space was evaluated. The scatter plot of the red pixels spread over half the entire space making it less unique(Fig. 2). HSV color space however performed better for illumination variations, but also had quite a few outliers. Similarly YCbCr was not robust enough. This can be seen from the scatter plots in Fig. 3 and Fig. 4.

Finally a custom color space using S and V components of HSV and Cr component of YCbCr helped to capture the

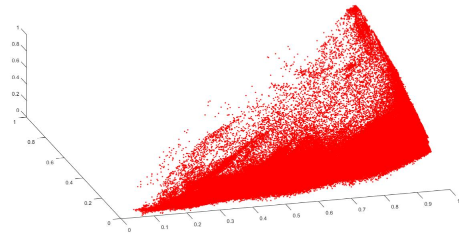


Figure 2. Scatter plot for RGB color space

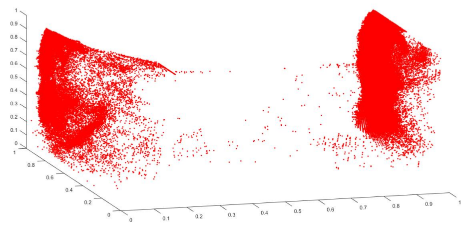


Figure 3. Scatter plot for HSVB color space

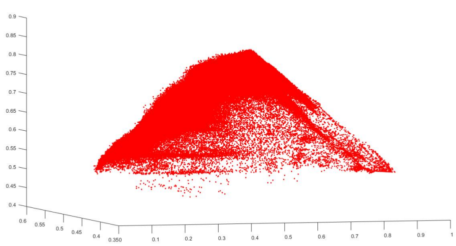


Figure 4. Scatter plot for YCbCr color space

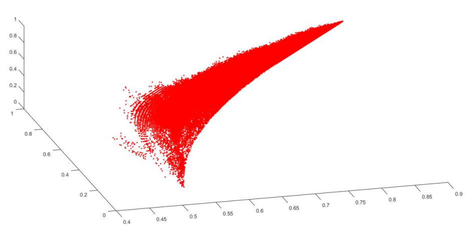


Figure 5. Scatter plot of CrSV color space

variations perfectly. It can be seen from the scatter plot(Fig. 5) that all the pixels are bunched much closely together and also there are no outliers. This allows easy classification of red pixels from others.

2.2 Gaussian Mixture Model

Initially a single Gaussian was used to model the color space. As the distribution was not truly ellipsoidal, the entire information was not captured well. Hence detection of few shades of red was unsuccessful as shown in Fig. 8 which is the segmented image of Fig. 7.

The parametric model using a single Gaussian for a pixel X was modeled as in equation.

$$P(X \mid Color) = \frac{1}{\sqrt{(2\pi)^3 det(A)}} \exp\left(\frac{(X-\mu)^\top A^{-1}(X-\mu)}{-2}\right)$$

where, A is the co-variance matrix and μ is the mean of the Gaussian distribution.

The performance of the system was drastically improved by using a mixture of such Gaussians with variable variance. Expectation Maximization algorithm was used to optimize the distribution parameters. The mean, variance and weights of each cluster was generated randomly and then the EM algorithm determined the optimal parameters. Cross validation was performed to determine the number of clusters by comparing the masks generated with the hand labeled masks. Fig. 9 shows the region of the color space modeled by GMM. It can be seen that the area of the scatter plot modeled is much greater than a single Gaussian. As a result the likelihood model captures the barrel pixels much better than single Gaussian.

2.3 Morphological operations

The training set consists of images with barrels that are occluded by poles and railings. One way of handling this was by using morphological operations such as erode and dilate. The following sequence of operations was performed and the corresponding outputs are shown in Fig. 6.

1. Erode operation with a disk structural element of size 1 which removes all the stray pixels.

2. Dilate operation with a disk structural element of size 10 which bridges the gaps between disconnected components of the barrel.

3. Second Erode operation with a disk structural element of size 7 to prevent the barrel blowing up in size which can affect the distance estimate.

2.4 Aspect Ratio filtering

After the series of morphological operations, blobs are detected on the resulting image using regionprops. The series of blobs detected are filtered through aspect ratio check which was modeled during training. During training, the ratio of

(a) Segmented Image **(b)** erode 1 **(c)** dilate **(d)** erode 2

Figure 6. Morphological operations to join blobs

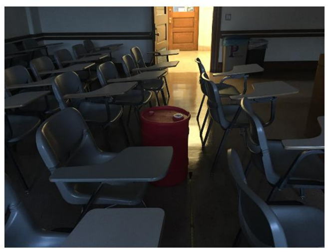


Figure 7. An image at which a single Gaussian fails

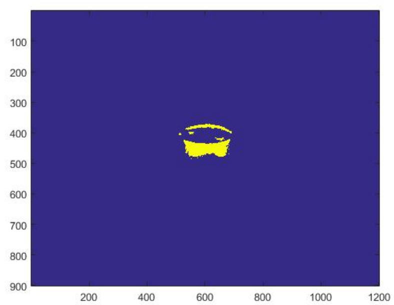


Figure 8. Segmented image at which a single Gaussian fails

height to width for all the hand labeled masks was calculated. The mean aspect ratio and the standard deviation was used as the parameters to filter the blobs. The threshold was set to 3 times the standard deviation.

$$meanAR - 3 * sdAR < AspectRatio < meanAR + 3 * sdAR$$

where meanAR is the mean aspect ration and sdAR is the standard deviation of aspect ratio obtained from training. The major and minor axis length of each blob was determined form regionprops and its aspect ratio was calculated. The blob was retained if the aspect ratio matched the above criteria, else was eliminated. It can be seen from Fig. 10, blobs 1 and 2 do not match the aspect ratio and are eliminated. But however this also led to failure in detection of occluded barrels in the test set shown in Table 2, Image 003.png.

2.5 Center and Distance Estimation

Once the barrel was detected based on the aspect ratio the center of mass of the blob was determined using regionprops.

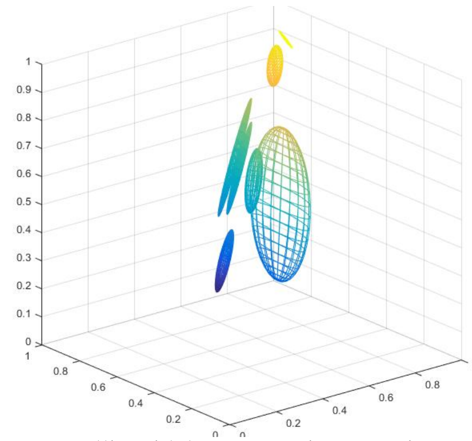


Figure 9. Ellipsoidals representing guassians of CrSV

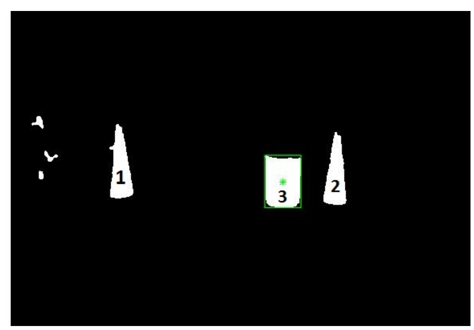


Figure 10. Rejection of blobs based on aspect ratio

The distance of the barrel was estimated using the width of the barrel which was again obtained from the minor axis of regionprops. From the training set, it was seen that the width of the barrel was proportional to inverse of distance. Thus a linear regression model was fit into this data to obtain the coefficients(Fig. 11).

$$width \propto \frac{1}{distance}$$

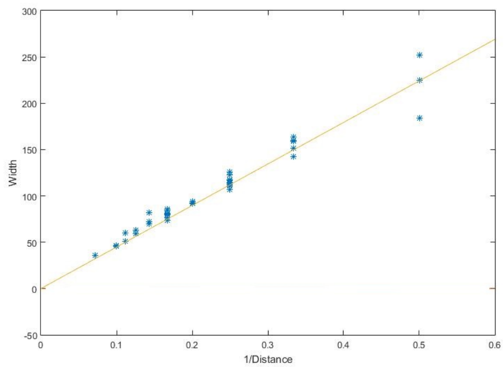


Figure 11. Variation of distance wrt width

3. Results

The above algorithm was run on 10 test images. The centers and the distances are tabulated in Table 2. The segmented images are shown in the same table with the center marked with a red *. The green star on the original image also shows the center. The parameters of the model used in evaluation are listed in Table 1. From the results, it can be seen that improper erosion and dilation affected a few results.

Table 1. Parameters of the model

Parameter	Value
Clusters	7
Mean Aspect Ratio	1.5371
SD of Aspect Ratio	0.102
Aspect Ration thershold	3
Distance Model	$0.0022 \times w - 0.0096$

4. Discussion and Conclusion

The incorrect result of 002.png can be traced to incorrect erosion and dilation process. 002.png failed due to minimal dilation preventing the two parts of the barrel to merge. But however this prevented merging of 2 the barrels in image 003.png. Hence one of the barrel was rightly detected while the other barrel even though segmented properly was rejected due to incorrect aspect ratio. More than half the barrel was hidden which drastically changed the aspect ratio.

Image 008.jpg was detected successfully as 3/4th of the barrel was visible in the bottom portion, but however the distance estimate was way off due to incorrect width measurement. This can be compensated by using both the height and the width of the barrel along with the detected area to determine the distance over just using width. Image 007.png and 005.png were captured pretty well because of the new color space that normalized most of the lighting effect.

The drawback of the current process is that the blobs are filtered based on only aspect ratio. This fails when the object is occluded as seen in some of the test sets.

Table 2. Results on test images

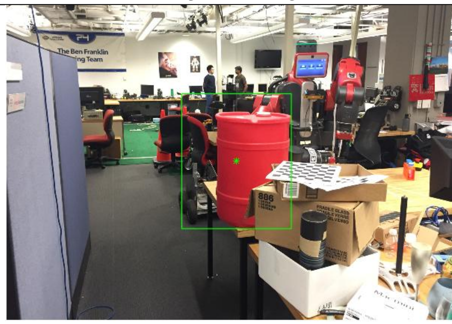
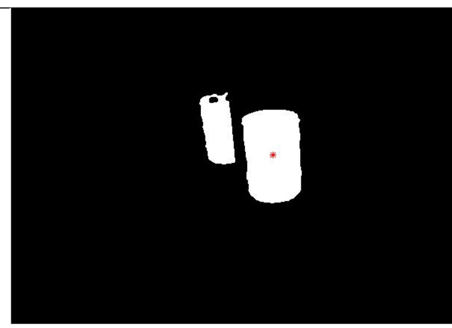
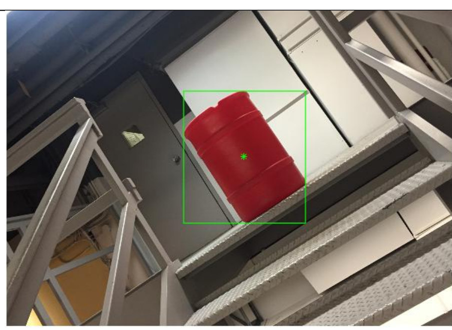
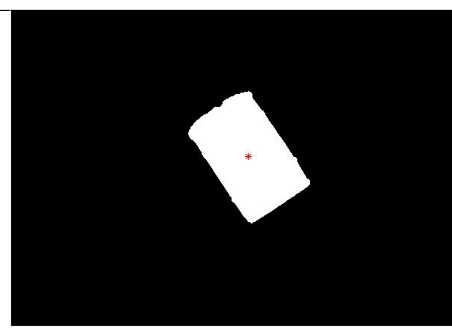
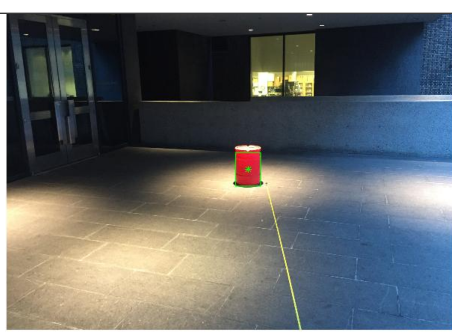
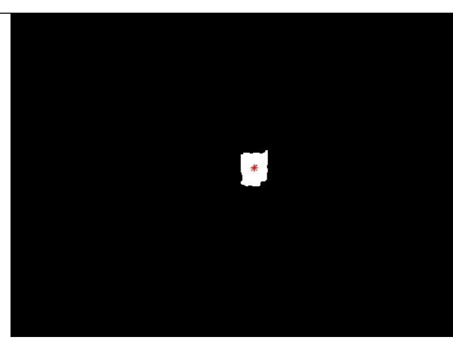
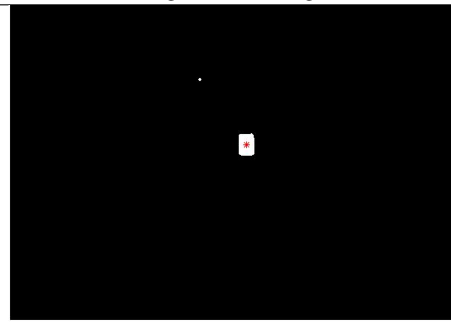
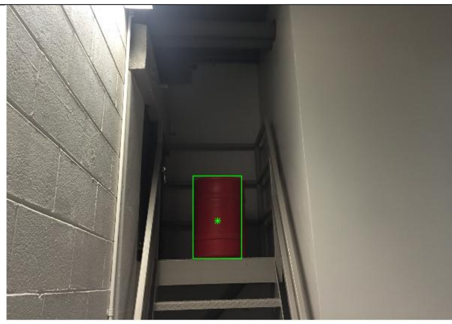
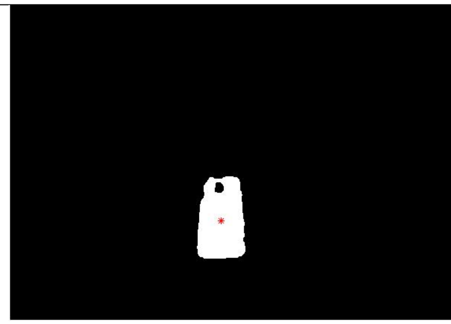
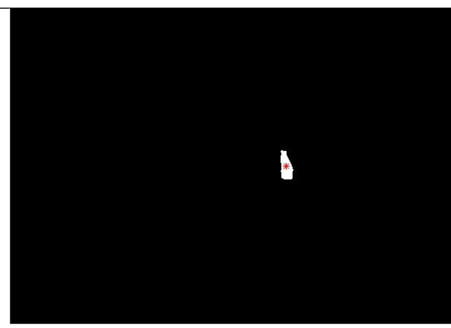
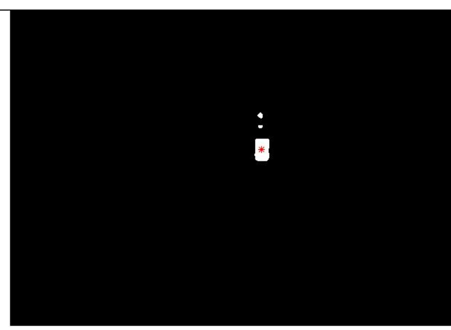
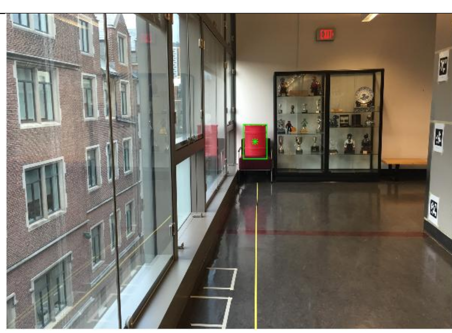
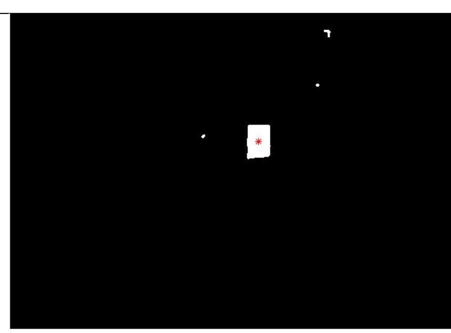
Image No	Original Image	Segmented image	Center	Distance
001.png			x = 616, y = 476	2.0969
002.png			x = 633.13, y = 513.52	3.3666
003.png			x = 700.65, y = 448.93	2.7914
004.png			x = 635.52, y = 446.75	1.9983
005.png			x = 647, y = 472	8.1899

Image No	Original Image	Segmented image	Center	Distance
006.png			x = 632.95, y = 430.65	10.5322
007.png			x = 566.02, y = 634.66	3.4350
008.png			x = 738.76, y = 480.19	16.5787
009.png			x = 674.24, y = 429.40	11.5428
010.png			x = 665.41, y = 398.55	6.8694