Color Image Segmentation: Two Different approaches

Anup Sawant
Department of Electrical Engineering,
College of Engineering and Science,
Clemson University,Clemson,
SC 29631, USA

Vinay Gidla
Department of Electrical Engineering,
College of Engineering and Science,
Clemson University,Clemson,
SC 29631, USA

ABSTRACT

The purpose of this project is twofold: To ascertain the superiority of *La'b'* color space over the RGB for segmentation in tune with natural perception, and to explore a segmentation technique: 1D SOFM, as an alternative to the popular k-means clustering technique. The performance and efficiency of these two color techniques are compared.

INTRODUCTION

Segmentation is the partition of a digital image into similar regions to simplify the image representation into something that is more meaningful and easier to analyze. Pixels in the region are similar to each other with respect to some characteristic property like color, intensity or texture. In this project, we concentrate only on the color.

Color based segmentation is significantly affected by the choice of color space. The general RGB color space gives a high degree of detail, but it is not in tune with the normal human perception. A different color space, the *La'b'* color space is a better representation of the color content of an image. Hence, this color space appears to be an ideal candidate for color based segmentation. The performance of these 2 color spaces is measured using the k-Means, a conventional clustering technique.

The second goal of this project is to implement segmentation using Self Organizing Feature Map (SOFM), an artificial neural network based clustering technique. The results are analyzed and also compared with the previous results obtained using k-Means.

METHODOLOGY

Color spaces

The *La'b'* color space is a 3 parameter color space with dimension ***L*** for lightness, a and ***b*** for the color-opponent dimensions. Unlike the RGB color model, *La'b'* color is designed to approximate human vision. It strives for perceptual uniformity, and its *L* component closely matches human perception of lightness. It can thus be used to make accurate color balance corrections by modifying output curves in the and *b* components, or to adjust the lightness contrast using the *L* component. In RGB space, which models the output of physical devices rather than human visual perception, these transformations can only be done with the help of appropriate blend modes in the editing application. The conversion between these 2 spaces is a 2-step process involving conversion to an intermediate XYZ color space. These conversions are available in the literature. An added advantage of *La'b'* color space is that distance metric for clustering techniques continues to be Euclidean.

In this project, we represented each pixel of the input image as a 3 element vector in the RGB color space. To compare the color spaces, we generated another data set corresponding for the *La'b'* color space. We transform the RGB vector to the XYZ color space followed by transformation to La'b' color space. Similarly, after the segmentation was complete, reverse transformation were applied to identify the corresponding pixels in the input image. We implemented the code for all the transformation functions.

k Means

k means is an unsupervised clustering algorithm which is frequently used in image processing. It follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. The next step is to take each data point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early grouping is done. At this point we need to re-calculate k new centroids, as centres of activity of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a

result of this loop, we may notice that the k centroids change their location.
In other words, the centroids do not move any more. The objective of this algorithm is to minimize the squared error function V,

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2$$

Where k is the number of clusters
μ_i is the mean of the all the points in a cluster.

At the end of the segmentation, the input image will have k distinct colors, identifying the color clusters.

Since our goal was to compare the performance of the 2 color spaces, we choose the same value of k for both the RGB and La'b' color spaces. The value of k was chosen in an adaptive manner i.e. we chose k depending on the real world image. The initial k centroids were chosen randomly. These centroids were placed such that they are as far as possible from each other. This step was noted to be important for better segmentation, as otherwise the means were stuck in local minima.

SOFM

A **self-organizing feature map (SOFM)** is a type of artificial neural network that is trained using unsupervised learning to produce a low-dimensional (here 1Dim), discrete representation of the input space of the training samples, called a **map**. The map seeks to preserve the topological properties of the input space.

This makes SOFM useful for visualizing low-dimensional views of the high-dimensional data such as the image content, akin to multidimensional scaling. The model was first described as an artificial neural network by the Finnish professor Teuvo Kohonen, and is sometimes called a **Kohonen map**.

A self-organizing map consists of components called nodes or neurons. Associated with each node is a weight vector of the same dimension as the input data vectors and a position in the map space. The procedure for placing a vector from input data space onto the map is to find the node with the closest weight vector to the vector taken from data space and to assign the map coordinates of this node to our vector.

It has been shown that while self-organizing maps with a small number of nodes behave in a way that is similar to K-means, larger self-organizing maps rearrange data in a way that is fundamentally topological in character.
ALGORITHM:
1. Randomize the map's nodes' weight vectors
2. Grab an input vector
3. Traverse each node in the map
   - Use Euclidean distance formula to find similarity between the input vector and the map's node's weight vector.
   - Track the node that produces the smallest distance (this node is the best matching unit, BMU)

4. Update the nodes in the neighborhood of BMU by pulling them closer to the input vector

Wv(t + 1) = **Wv**(t) + Θ(t)α(t)(**D**(t) - **Wv**(t))
5. Increment t and repeat from 2. While $t < \lambda$

Above, α(t) is a monotonically decreasing learning coefficient and **D**(t) is the input vector. The neighborhood function Θ (v, t) depends on the lattice distance between the BMU and neuron *v*. In the simplest form it is one for all neurons close enough to BMU and zero for others, but a Gaussian function is a common choice, too. Regardless of the functional form, the neighborhood function shrinks with time.[2] At the beginning when the neighborhood is broad, the self-organizing takes place on the global scale. When the neighborhood has shrunk to just a couple of neurons the weights are converging to local estimates.

In this project, the number of units chosen in the 1D SOFM was adaptive i.e. varied based on the input image. The values of these units were initialized randomly from the input image La'b' color space. Also, the samples for training the map were selected from the input image at random. The map was trained for a fixed number of iterations = 500 * N i.e. 500 times the number of units in the map. This figure was proposed by the Kohonen.

Factors Affecting Segmentation

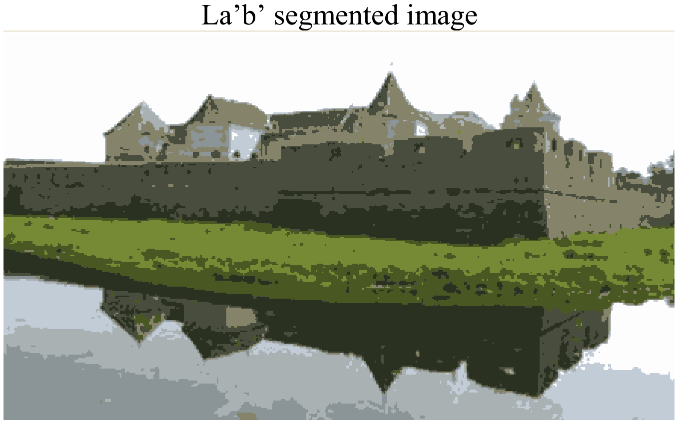
Several related factors affect the k means and SOFM segmentation. The factors that have great impact on the final segmented image are:
1. Number of cluster centers: k and number of nodes.
2. The location of the cluster centers: initial values of the k centroids and of the nodes in the map.
3. Noise or blurring in the image

RESULTS: RGB vs La'b'

Input image: Castle.jpg (K = 10)

RGB segmented image

La'b' segmented image

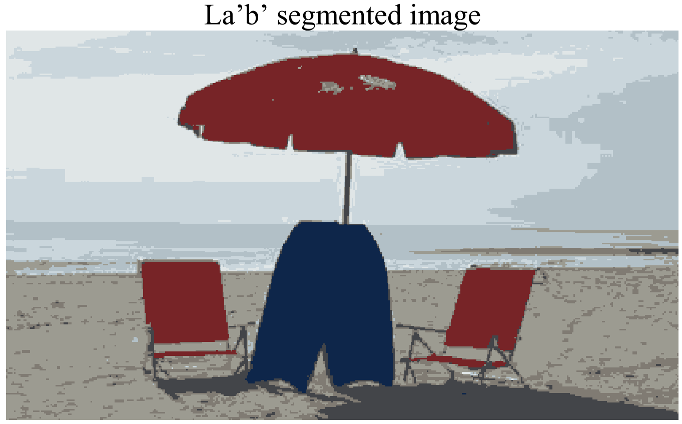


For the castle image above, the improved performance of Lab color segmentation over RGB segmentation can be perceived. Few points to note are the uniformity of the grass, the castle sidewalls and the reflection of the castle in the water.

Input image: Beach.jpg (K = 8)

RGB segmented image

La'b' segmented image



Here too, the improvement can be noted in the umbrella top and its shadow.

These examples in conjunction with further results from database clearly favor the use of La'b' color space for color based image segmentation.

RESULTS: k-Means vs SOFM

k-Means

(Execution time = 4.594 s)

SOFM

(Execution time = 0.016 s)

k-Means

(Execution time = 3.656 s)

SOFM

(Execution time = 0.016 s)

Here the input image was first transformed into the La'b' color space before being segmented by the 2 techniques.
Comparing the above results, we can see that SOFM approach gives results comparable to the k-Means. The important point to note is that it does in a fraction of the time required by the k-Means approach. The difference arises due to distinct convergence algorithms. K-Means convergence is closely related to the image, whereas SOFM converges after a fixed number (user controlled) of iterations.

The comparable performance of SOFM for this figure in a minute execution time is again apparent.

These results along with other tests on different images suggest two things. First, SOFM with appropriate number of neural units is a good segmentation technique for image segmentation. Second, it is far efficient procedure compared to the conventional k-Means technique.

CONCLUSION

Image segmentation in *La'b'* color space scores over conventional RGB color space. *La'b'* results are based more on visual perception and hence not obscured by high degree of detail. In the project, this has been validated by testing a variety of different images.
However the output depends largely on the initial cluster centroids and the number of clusters selected. In order to get the desired results, the choice of initial clusters should be made carefully.

The SOFM technique of image segmentation turns out to be much more efficient segmentation technique compared to the k-Means. While it constantly gave a comparable result to k-Means, the execution time was merely a fraction of the other technique. One of the main disadvantages with the k means algorithm is that there is no guarantee that the algorithm would converge. However this problem does not exist with SOFM as it executes only for a fixed time decided by the neural architecture, independent of the image.

FUTURE WORK

Further neural net algorithms such as Neural Gas, Growing Neural Gas can be tested for improvement over SOFM performance.
Also, the color attributes can be combined with the texture features for better segmentation.

REFERENCES

1. Segmentation and Histogram Generation using the HSV Color Space For Image Retrieval - Shamik Sural, Gang Qian and Sakti Pramanik
2. www.wikipedia.org : For definitions & formulae
3. Color Image Compression using Self Organizing Feature Map - B. Diri and S.Albayrak (Turkey)
4. **Image segmentation Using Color and Texture Features –** *Mustafa Özden and Ediz Polat*