Color Image Segmentation Using K-means Clustering, EM Algorithm and Mean-shift Method

Ke Tang	**16823384**
Di Sun	**69632624**
Yifei Tang	**15894301**

Apr-27-2010

Abstract

Image segmentation is very essential and critical to image processing and pattern recognition. This project addresses the problem of segmenting a color image into different regions. We have analyzed and implemented three unsupervised learning algorithms namely K-Means clustering, EM algorithm and Mean-Shift method in color image segmentation. The three algorithms all produce relatively good results and can partition color images into regions. We have also compared these segmentation methods in different color spaces such as RGB, HSI and CIE color spaces. K-Means clustering and EM algorithm are sensitive to initial value and require the number of regions. However, Mean Shift method can segment images into variable number of regions and has better performance than K-Means and EM algorithm due to insensitivity to initial values without significant increase of complexity. In HSI color space, the results are not as good as in RGB and CIE because Euclidean distance is not sensitive to hue and saturation. The performance of image segmentation in CIE Luv color space is the best because of perceptually uniform property of CIE Luv.

1. **Introduction:**

Images are considered as one of the most important medium of conveying information. Understanding images and extracting the information from them is an important aspect of Machine learning. One of the first steps in direction of understanding images is to segment them and find out different objects in them. In [4], the author introduces that K-means clustering technique is computationally efficient and can be applied to multidimensional data, especially color image. In [5], the author proposes that EM algorithm can give a stable solution for color image segmentation and improve the initial estimates. In [2], the author introduces the mean shift as a robust method to efficiently detect the modes of the density. Due to simplicity and relatively good performance of the three algorithms, we have applied and even improved them in color image segmentation and compared their performance in different color spaces.

2. **Data set:**

We have implemented the three methods on color images, which are 3-dimensional data. Through experiments, color images as a dataset do give us very straightforward information about how these methods work and are really easy for us to compare their performance. Applying these algorithms to color images is shown to be very useful and practical for machine learning. So this dataset gave us reliable results and was a good source to accomplish our goal indeed.

3. **Methods:**

(1) K-means Clustering Algorithm:

K-Means algorithm is an unsupervised clustering algorithm that classifies the input data points into multiple classes based on their inherent distance from each other. The algorithm assumes that the data features form a vector space and tries to find natural clustering in them. The centroids μ_i of the clusters are obtained by minimizing $V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2$ iteratively, where there are k clusters $S_i, i = 1, 2, \ldots, k$ and μ_i is the centroid or mean point of all the points $x_j \in S_i$.

Without other information, we need to guess the “proper” number of cluster. We denote the desired number of clusters k and let k samples randomly chosen from the data set serve as initial cluster centers. The algorithm is then:

(1) Begin initial values N, k, $\mu_1, \mu_2, ..., \mu_k$

(2) Classify every sample according to the nearest mean μ_i and gather the k clusters

(3) Compute new means for every new cluster

(4) Repeat (2) and (3) until the k means convergence

In color image segmentation, the algorithm will take color images as input, iteratively cluster the points based on Euclidean distance from the centroids and finally segment the color image into several different regions. However, the results of K-means clustering which uses a random initialization may differ significantly according to the selection of initial parameters.

The experiment results are shown in Fig 1 and Fig 2. From the results, we can see that K-means clustering method can segment color images into different regions. Moreover, the results in Fig 2 varying with different initial values, such as leaves in the corner, show that K-Means is very sensitive to initial values and require the number of regions.

(2) EM algorithm:

Expectation Maximization (EM) is one of the most common algorithms used for density estimation of data points in an unsupervised setting. The algorithm relies on finding the maximum likelihood estimates of parameters when the data model depends on certain latent variables. In EM, alternating steps of Expectation (E) and Maximization (M) are performed iteratively till the results converge. The E step computes an expectation of the likelihood by including the latent variables as if they were observed, and maximization (M) step, which computes the maximum likelihood estimates of the parameters by maximizing the expected likelihood found on the last E step. Although EM iteration does not decrease the observed data likelihood function, there is no guarantee that the sequence converges to a maximum likelihood estimator.

Given a training dataset $\{x_1, x_2, \cdots x_m\}$ and model $p(x,z)$ where z is the latent variable, we have:

$$l(\theta) = \sum_{i=1}^{m} \log p(x;\theta) = \sum_{i=1}^{m} \log \sum_{z} p(x,z;\theta)$$

Since the latent variable z is unknown, we use approximations in the form of E & M steps mentioned above and formulated below.

E Step, for each i:

$$Q_i(z^{(i)}) := p(z^{(i)} \mid x^{(i)};\theta)$$

M Step, for all z:

$$\theta := \arg\max_{\theta} \sum_{i} \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}$$

where Q_i is the posterior distribution of $z^{(i)}$ ’s given the $x^{(i)}$ ’s.

In color image segmentation, the observed color image is considered as a mixture of multi-variant Gaussian densities and the number of densities is assumed to be known. The segmentation of the color image can be proposed on the basis of the maximum likelihood (ML) estimation for the clustering problem. The iterate EM algorithm is introduced to estimate the parameters of the likelihood functions. Given the estimation result, the ML based pixel-cluster memberships provide a reasonable segmentation of the image.

In color spaces, every pixel is expressed as a vector which contains three elements, for example, in RGB model, $\boldsymbol{y}_i = (y_i^R, y_i^G, y_i^B)$. Here, $\boldsymbol{y}_i$ is supposed to be observations drawn from a mixture of multi-variant Gaussian density functions whose parameters (mean vectors μ and covariance matrices Σ) are unknown. Thus, the number of densities is manually set to be M and the form of the multivariate Gaussian mixture is $p(\boldsymbol{y}_i) = \sum_{j=1}^{M} \alpha_j p_j(\boldsymbol{y}_i; \mu_j, \Sigma_j)$. α_j is prior probability of the j-th Gaussian density in the mixture that satisfies $\sum_{j=1}^{M} \alpha_j = 1$.

First, we initialize those parameters as: $\alpha_j^{(0)} = \frac{1}{M}$; $\mu_j^{(0)} = \frac{255 \cdot j}{M}$; $\Sigma_j^{(0)} = [\boldsymbol{I}]_{3\times 3}$.

Then, assuming the p-th estimated value has been obtained, the updating rule for the p+1-th estimates is given as

$$T_{ij}^{(p)} = \frac{\alpha_j^{(p)} p(\boldsymbol{y}_i; \mu_j^{(p)}, \Sigma_j^{(p)})}{\sum_{j=1}^{M} \alpha_j^{(p)} p(\boldsymbol{y}_i; \mu_j^{(p)}, \Sigma_j^{(p)})}; \quad \alpha_j^{(p+1)} = \frac{1}{N} \sum_{i=1}^{N} T_{ij}^{(p)}; \quad \mu_j^{(p+1)} = \frac{\sum_{i=1}^{N} T_{ij}^{(p)} \boldsymbol{y}_i}{\sum_{i=1}^{N} T_{ij}^{(p)}};$$

$$\Sigma_j^{(p+1)} = \frac{\sum_{i=1}^{N} T_{ij}^{(p)} (\boldsymbol{y}_i - \mu_j^{(p)})^T (\boldsymbol{y}_i - \mu_j^{(p)})}{\sum_{i=1}^{N} T_{ij}^{(p)}}$$

where N is the number of i.i.d. observations.

Once the EM algorithm converges when the difference between old and new estimates are less than certain threshold (i.e 0.00001), the final estimates are obtained. Finally, the image segmentation is carried out by conventional ML method according to $L_j = \max_i \frac{\exp\{-\frac{1}{2}(\boldsymbol{y}_j - \mu_i^{(final)})(\Sigma_i^{(final)})^{-1}(\boldsymbol{y}_j - \mu_i^{(final)})^T\}}{\left|\Sigma_i^{(final)}\right|^{-\frac{1}{2}}}$, where L_j is the label of the j-th element.

During the process of implementation EM algorithm, the problem arises that when the number of clusters increases, singular matrix always appears after certain times of iterations and the performance of the algorithm is getting worse. Consequently we restrict the maximum number of iterations to avoid this to happen. The experiment results are shown in Fig 3 and Fig 4. In Fig 3, the color images can be partitioned appropriately using EM algorithm. From Fig 4, we can see that with different initial values, the segmentation results differ much, especially the leaves. So, although EM algorithm can segment color images, it is computationally expensive and very sensitive to initial values while requiring the number of clusters.

(3) Mean-shift method:

Mean shift is a commonly-used unsupervised clustering technique for locating the maxima of a density function (density mode) given discrete samples. It does not require prior knowledge of the number of clusters, and does not constrain the shape of the clusters, which is very useful for detecting the modes of this density without any prior information, especially in image segmentation. Mean shift is an iterative method and belongs to the most popular density estimation method namely kernel density estimation (also known as Parzen window technique), which is a non-parametric way of estimating the probability density function of a random variable.

Given n data points $x_i, i = 1, \dots, n$ on a d-dimensional space R^d, the multivariate kernel density estimate obtained with kernel K(**x**) and window radius h (also called bandwidth) is

$$f(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K(\frac{x - x_i}{h})$$

For radially symmetric kernels, it suffices to define the profile of the kernel $k(x)$ satisfying

$$K(x) = c_{k,d} k(\| x \|^2)$$

where $c_{k,d}$ is a normalization constant which assures $k(x)$ integrates to 1. Quite often K is taken to be a multivariate normal kernel $K_N(x) = (2\pi)^{-d/2} \exp\left(-\frac{1}{2}\| x \|^2\right)$, in which the profile is defined as:

$$k_N(x) = \exp(-\frac{1}{2}x), x \geq 0$$

In order to get the modes of the density function, the gradient function $\nabla f(x)$ should be zero. So that:

$$\nabla f(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^{n} (x_i - x) g(\| \frac{x - x_i}{h} \|^2)$$

$$= \frac{2c_{k,d}}{nh^{d+2}} \left[\sum_{i=1}^{n} g(\| \frac{x - x_i}{h} \|^2) \right] \left[\frac{\sum_{i=1}^{n} x_i g(\| \frac{x - x_i}{h} \|^2)}{\sum_{i=1}^{n} g(\| \frac{x - x_i}{h} \|^2)} - x \right] = 0$$

where $g(x) = -k'(s)$. So the mean shift vector is obtained by

The mean shift vector always points toward the direction of the maximum increase in the density. Hence, the mean shift procedure can be implemented by successive computation of the mean shift vector $m_h(x^t)$ and translation of the window $x^{t+1} = x^t + m_h(x^t)$, which is guaranteed to converge to a point where the gradient of density function is zero.

Although mean shift algorithm using normal kernel requires large number of steps to converge, we still choose normal kernel since the quality of the results is always better than uniform kernel. For normal kernel, the mean shift vector is given by

$$m_h(y_j) = y_{j+1} - y_j = \frac{\sum_{i=1}^{n} x_i \exp(-\| \frac{y_j - x_i}{h} \|^2)}{\sum_{i=1}^{n} \exp(-\| \frac{y_j - x_i}{h} \|^2)} - y_j$$

So that the successive location y_j of the kernel can be obtained iteratively by

$$y_{j+1} = \frac{\sum_{i=1}^{n} x_i \exp(-\| \frac{y_j - x_i}{h} \|^2)}{\sum_{i=1}^{n} \exp(-\| \frac{y_j - x_i}{h} \|^2)}$$

In application of image segmentation, when given the number of the clusters, mean shift method is very similar to k-means and more generalized and robust by using kernels to estimate the densities. The procedure is as follows:

(1) Choose a search window size and k initial locations of the search window.
(2) Compute the mean location (centroid of the data) in each search window.
(3) Center the search window at the mean location computed in step (2).
(4) Repeat (2) and (3) until convergence and store the k mean values as mode peaks.
(5) For each pixel, find the nearest mode peak, substitute it with the specific mean value and assign label to it.

By this method, color images can be segmented into given number of clusters but the performance is very sensitive to initial values of mean location and the number of clusters should be given as K-Means and EM algorithms.

In order to overcome the drawback of fixed clusters, we propose a new method to segment images into variable number of regions and further reduce the computation complexity. The proposed algorithm is:

(1) Choose a search window size.
(2) Choose an initial location of the search window.
(3) Iterate the mean shift procedure until convergence.
(4) Group all pixels closer than a specific range of values and assign labels to them.
(5) Find the rest pixels which are not assigned a label and repeat (2) to (4) until all pixels are clustered.

Using this method, color images are segmented properly without given number of regions and the performance is also improved as the effect of initial values are reduced to some extent by reducing the impact of irrelevant pixels in every repetition.

However, images are two-dimensional lattice of p-dimensional vectors, where p=1 in the gray level case, p=3 for color images. As the space of the lattice is known as the spatial domain while the gray level or color information is represented in the range domain, so the authors in [2] combined both spatial and range values in the method by using two separate bandwidth h_s and h_r to control the spatial and range kernels respectively. The whole procedure contains two steps: mean shift discontinuity preserving filtering and mean shift segmentation. For discontinuity part, the algorithm is:

(1) For each image pixel x_i, initialize $y_{i,1} = x_i$.
(2) Iterate the mean shift procedure until convergence.
(3) The filtered pixel values are defined as $z_i = (x_i^s, y_{i,con}^r)$, where x_i^s is the location of filtered pixel and $y_{i,con}^r$ is the convergence value of the pixel.

For segmentation part, the algorithm is:

(1) Determine the clusters by grouping all z_i, which are closer than h_s in the spatial domain and h_r in the range domain.
(2) Assign class labels to clusters.
(3) Optional: Eliminate regions smaller than P pixels.

However, this algorithm is really computationally expensive and not practical in reality because time consumption is sometimes more important than performance. So in implementation, we relax the condition of using all data to calculate the new mean for every pixel. Instead, we combine spatial values into calculation and use the spatial range, which is an area of $h_s \times h_s$, to iteratively compute the mean.

The experiment results based on the three Mean-Shift algorithms described above are shown in Fig 5. From the results, we can see the performance of the method considering both spatial and range values is improved greatly with more continuous regions and the algorithm can automatically partition color images into variable number of regions without any prior knowledge of the number of clusters. Moreover, it is more robust to initial values and outliers.

The complexities of the three kinds of mean shift methods are also displayed in Table 1. Obviously, we can see that the method given the number of clusters has the worst performance with relatively less complexity while the method considering both spatial and range values has the best performance but the most complexity even when we relax the computation during implementation. Hence there is always tradeoff between time consumption and performance.

(4) Comparison between K-Means clustering, EM algorithm and Mean-Shift method

We have carried out experiments to compare these three algorithm mentioned above. The results are shown in Fig 6. Just as analyzed, Mean-Shift has the best performance with clearer boundary and more continuous regions. The performances of K-means clustering and EM algorithm are not as good since they are sensitive to initial value and require the number of clusters.

(5) Color Spaces**:**

Color is perceived by humans as a combination of tristimuli R (red), G (green), and B (blue). From R, G, B representation, we can derive other kinds of color representations (spaces) by using either linear or nonlinear transformations. Several color spaces have been used for color image segmentation, but according to [2], none of them is proved best for all kinds of color images in different applications.

RGB: RGB is the most commonly used model for the television system and pictures acquired by digital cameras. Directly based on the three primary color components, the RGB color space can be geometrically represented in a 3-dimensional cube (Fig. 6). The coordinates of each point inside the cube represent the values of red, green and blue components, respectively.

HSI/HSV: The HSI (hue, saturation, intensity) system is another commonly used color space in image processing, which is more intuitive to human vision. HSV (hue, saturation, value) is one variant of HSI system. The HSI system separates color information of an image from its intensity information. Color information is represented by hue and saturation values, while intensity, which describes the brightness of an image, is determined by the amount of the light. The HSI/HSV coordinates can be directly transformed from the RGB space.

CIE: CIE (Commission International de l'Eclairage) color system was developed to represent perceptual uniformity, and thus meets the psychophysical need for a human observer. It has three primaries denoted as X, Y, Z, which can be computed by a linear transformation from RGB tristimulus coordinates. Then two typical examples, CIE ($L^*a^*b^*$) and CIE($L^*u^*v^*$) can be obtained by nonlinear transformation of X, Y, Z.

In our project, we primarily use Euclidean distance for color comparison. The Euclidean distance metric is defined as:

$$D(\vec{v}_1, \vec{v}_2) = \| \vec{v}_1 - \vec{v}_2 \| = \sqrt{(v_{1,1} - v_{2,1})^2 + (v_{1,2} - v_{2,2})^2 + (v_{1,3} - v_{2,3})^2}$$

where $\vec{v}_1 = [v_{1,1} \quad v_{1,2} \quad v_{1,3}]^T$ is a color triplet. The Euclidean doesn't quantify color similarity in the RGB space as well as it does in the CIE Luv space. Hence, it is sensitive to variations in intensity, but not very sensitive to variations in hue and saturation.

We applied the three methods in the three color spaces and the results are shown in Fig 6. From the results, it is clearly shown that color image segmentation works best in CIE Luv color space and worst in HSI color space because Euclidean distance cannot recognize hue and saturation differences well and perceptually uniform property of CIE Luv is very suitable for Euclidean distance.

4. **Conclusion**:

In this project, we have implemented three unsupervised learning algorithms, which can basically produce good segmentation results. For both K-means clustering and EM algorithm, they require the number of regions to be segmented and are very sensitive to initial values. Mean-Shift method combining spatial and range values gives more robust performance because it does not require prior knowledge of the number of clusters and it will produce arbitrarily-shaped clusters that depend upon the topology of the data. However, both EM algorithm and Mean-Shift are computationally expensive, which is not so practical and needs to be improved by relaxing some restrictions. Through experiments of comparison between segmentation results in different color spaces, we find out that Euclidean distance we used for segmentation is not sensitive to hue and saturation and doesn't quantify color similarity in RGB color space as well as it does in CIE Luv color space. So the results in HSI color space are worst and the ones in CIE Luv color space are the best of all.

5. References:

[1] Cheng, H.D., Jiang, X.H., Sun, Y., Wang, JingLi. Color image segmentation: advances and prospects(2001), Pattern Recognition (34), No. 12, December 2001, pp. 2259-2281.
[2] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Machine Intell., 24:603–619, 2002.
[3] Richard O.Duda, Peter E. Hart, David G. Stork. Pattern Classification. p526-530.2009
[4] Ilea, Dana E. and Whelan, Paul F. (2006) Color image segmentation using a spatial k-means clustering algorithm. In: IMVIP 2006 - 10th International Machine Vision and Image Processing Conference, 30 August - 1 September 2006, Dublin, Ireland.
[5] T. Yamazaki. Introduction of EM Algorithm into Color Image Segmentation(1998). Proceedings of ICIRS'98, pp.368-371, August 1998.
[6] Wu, Y., Yang, X., and Chan, K. L. (2003). Unsupervised color image segmentation based on Gaussian mixture model. Information, Communications and Signal Processing, Vol 1, pp. 541–544.
[7] Slawo Wesolkowski and Ed Jernigan. Color edge detection in RGB using jointly Euclidean distance and vector angle. Vision Interface '99, Trois-Rivières, Canada, 19-21 May, 1999.
[8] J. Park, G. Lee, S. Park. Color image segmentation using adaptive mean shift and statistical model-based methods. Computers & Mathematics with Applications, Volume 57, Issue 6, Pages 970-980
[9] M. Luo, Y.F. Ma, H.J. Zhang, “A Spatial Constrained K-Means Approach to Image Segmentation”, Proc. the 2003 Joint Conference of the Fourth International Conference on Information, Communications and Signal Processing and the Fourth Pacific Rim Conference on Multimedia, Vol. 2, pp. 738-742, 2003.
[10] Y.Deng, and B.S.Manjunath, "Unsupervised segmentation of color-texture regions in images and video," IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI '01), August 2001.

6. Tables and Figures:

Table 1

Methods	Mean shift given the number of clusters	Mean Shift in [2]	Mean Shift relaxing the condition of [2]	Mean Shift considering tradeoff
Complexity	$O(c*N*E_i)$	$O(N*N*E_i)$	$O(h_s^2*N*E_i)$	$O(\sum_j N_j*E_i)$

c : The number of clusters

N : The total number of pixels

E_i : The expected times of iteration

h_s : Spatial window

N_i : The number of remaining pixels during determining ith cluster

(a) (b)

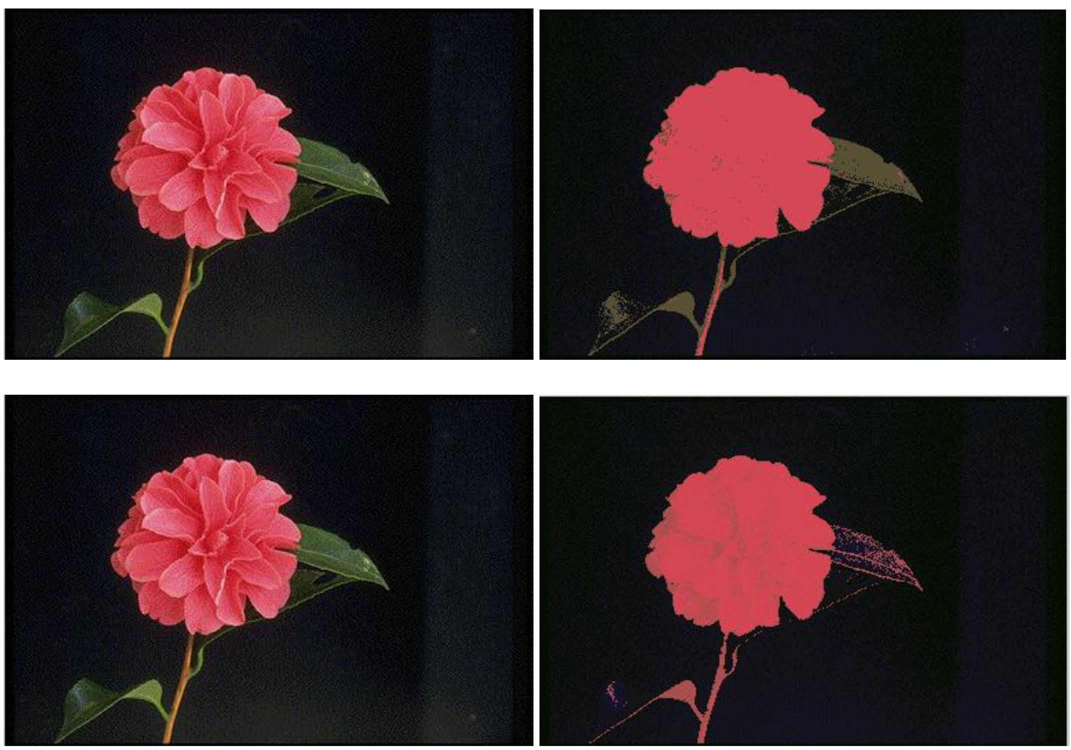
Fig 1: color image segmentation using K-Means Clustering given k=7
(a) Original color image (b) Segmented color image

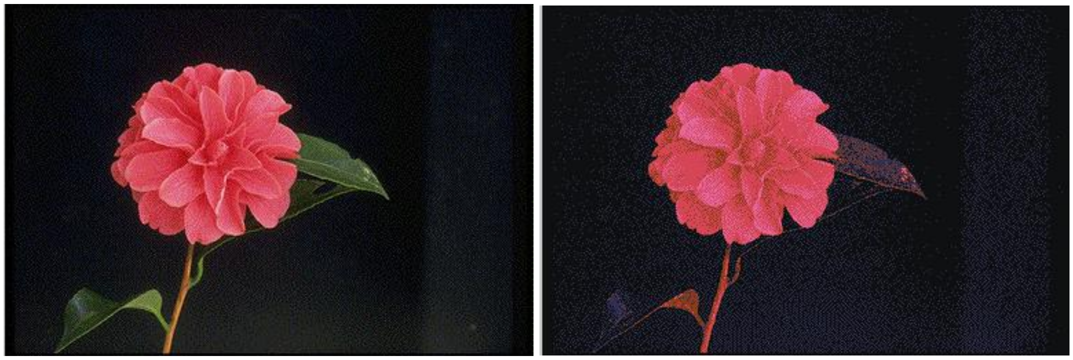
(a) (b)

Fig 2: color image segmentation using K-Means Clustering with different random initial values
(a) Original color images (b) Segmented color images

(a) (b)

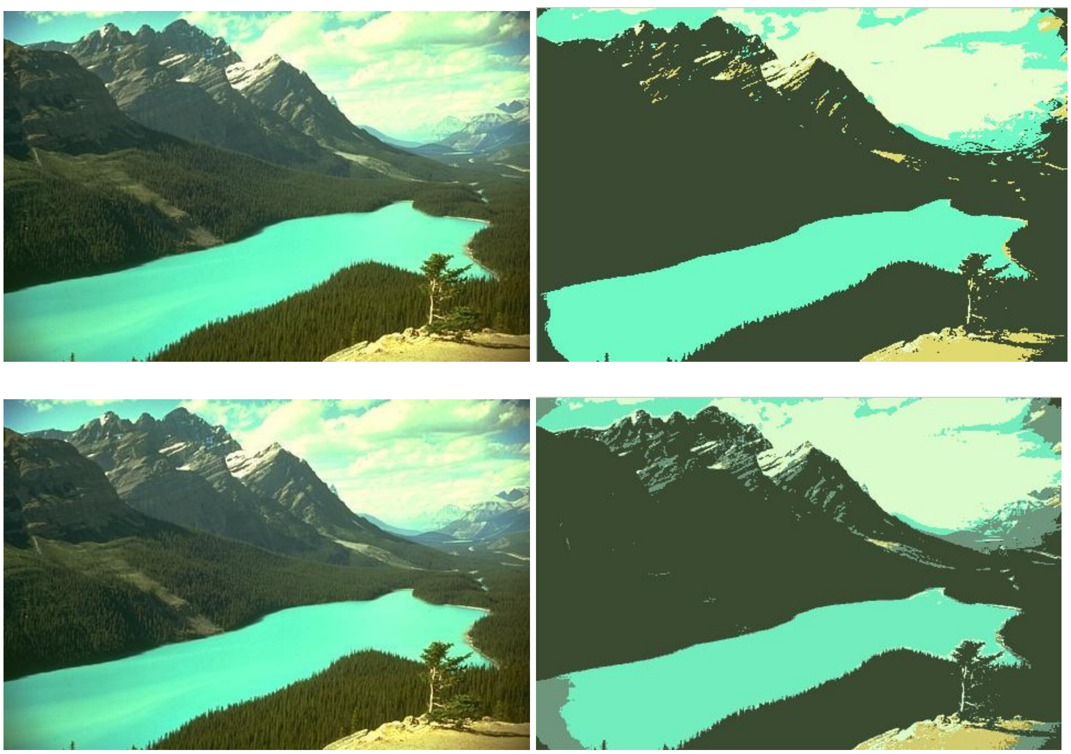
Fig 3 color image segmentation using EM algorithm given k=8
(a) Original color image (b) Segmented color image





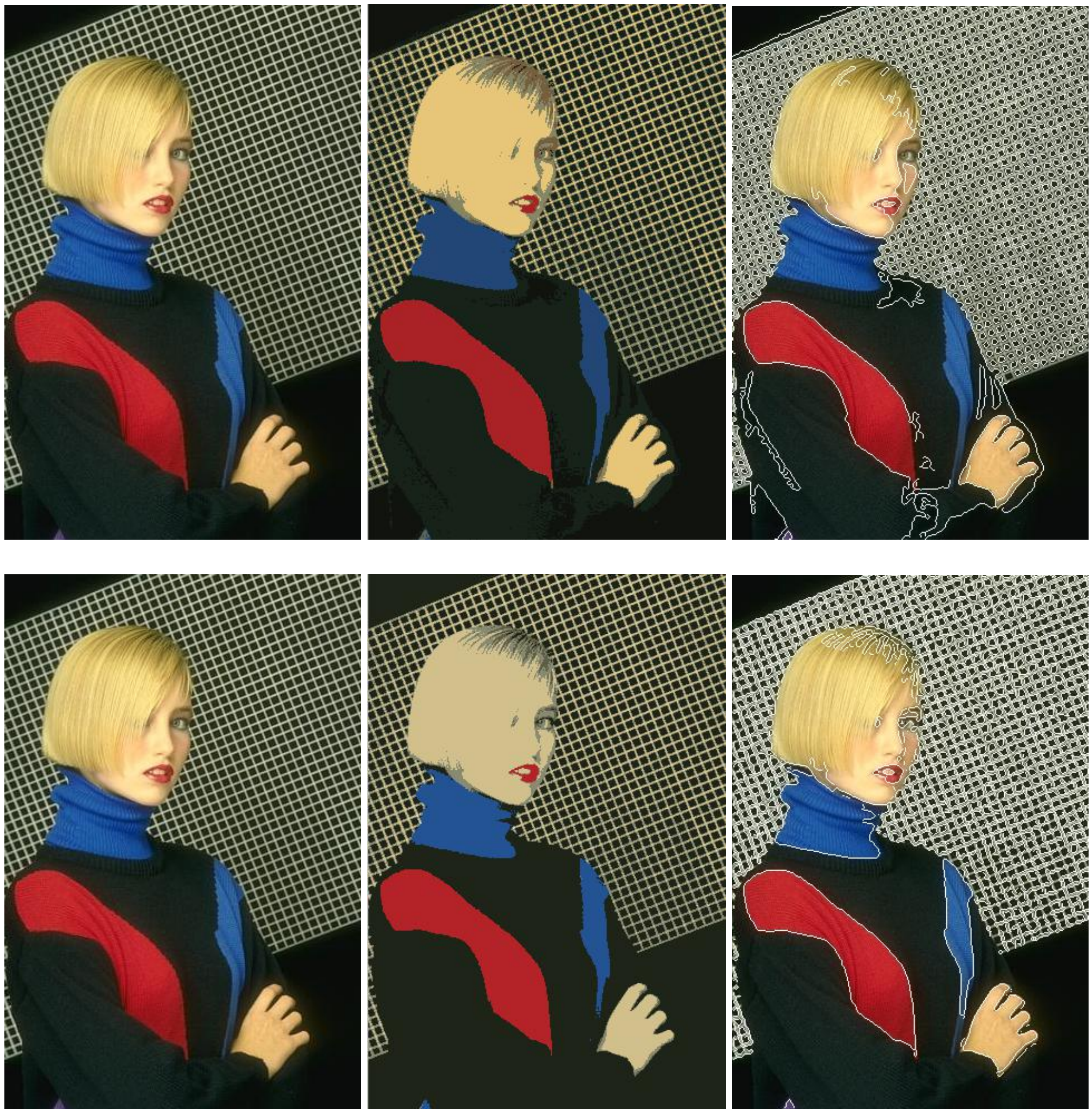
(a) (b)

Fig 4 color image segmentation using EM algorithm with different random initial values
(a) Original color images (b) Segmented color images



(a) (b)

Fig 5: color image segmentation using three different kinds of Mean-Shift methods
(a) Original color images (b) Segmented color images by Mean-Shift given k=10, Mean-Shift considering only range values without prior knowledge ($h_r = 0.1$) and Mean-Shift considering both spatial and range values without prior knowledge ($(h_s, h_r) = (16, 0.1)$) respectively from top to down



(a) (b) (c)

Fig 6: comparison between K-Means clustering, EM algorithm and Mean-Shift method
(a) Original color images (b) Segmented color images by K-Means, EM, Mean-Shift algorithms respectively from top to down (c) Detected boundaries from the results of the three methods

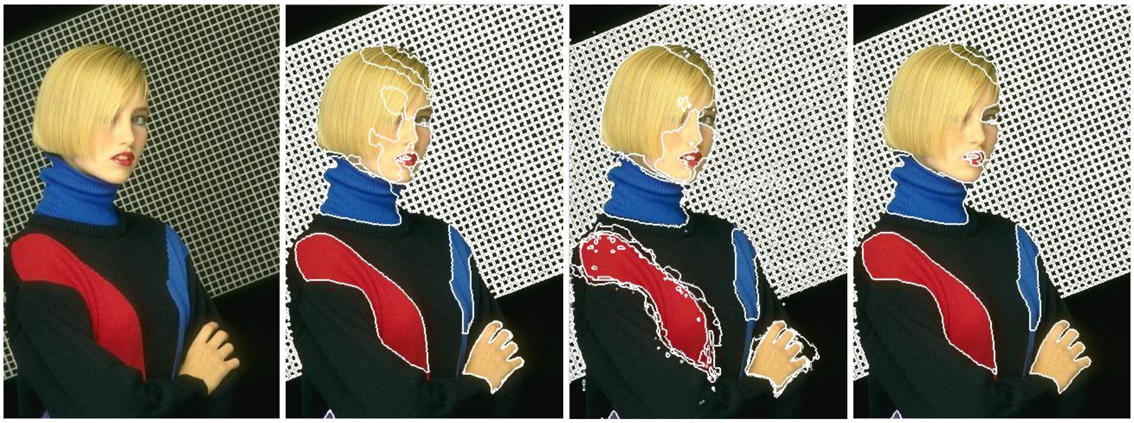


Fig 7: comparison of color image segmentation in RGB, HSI and CIE Luv color spaces respectively (a) Original color images (b) Segmented color images in RGB, HSI and CIE Luv color spaces from left to right respectively using Mean-Shift method with parameter $(h_s, h_r) = (16, 0.1)$