A Tutorial on MRI Image Segmentation Using Fuzzy C-Means in MATLAB

Brisaac Johnson and Chris Crawford

College of Engineering, University of Alabama, Tuscaloosca, USA

ABSTRACT

Image segmentation is challenging because an image has to be partitioned into regions with pixels that contain similar attributes. To accomplish this task, many use MATLAB, Python, or other environments that allow them to set thresholds for the image, to group similar pixels together, or remove undesirable pixels from the image. When approaching the image segmentation problem, many use the fuzzy c-means algorithm to group similar pixels with the same features in an image. Hence, it makes it easier to analyze specific areas in the image, such as a tumor that may be present. This paper presents a tutorial on performing image segmentation using MATLAB and Fuzzy C-Means. Through this tutorial, we examine how fuzzy c-mean is used to solve the image segmentation problem and the methodologies used to cluster data properly.

Keywords: MATLAB, Fuzzy C-means, Clustering, MRI, Image segmentation, Tumor

INTRODUCTION

Image segmentation is challenging, because an image has to be partitioned into regions with pixels that contain similar attributes. To accomplish this task, many use software such as MATLAB, or software that allows them to set thresholds for the image, to group similar pixels together or remove undesirable pixels from the image. When approaching the image segmentation problem, many use the fuzzy c-means algorithm to group similar pixels with the same features in an image.

However, when using fuzzy c-means, data we are trying to cluster can belong to more than one cluster, this is different from the k-means clustering method, which has stricter clustering methods. One of the methods used to approach the image segmentation problem was to use the fuzzy c-means with weighted hues and intensity. By using the tuned weights, it indicates how accurate each color component is. Doing this allows one to cluster the color components more effectively. Though, a problem that was faced in this approach was selecting the correct color space. HSI space was the solution to this problem, using this space, allows for better accuracy. Selecting the two components of HSI such as the Intensity and Hue is more suitable then selecting two components of RGB to perform image segmentation.

Using these image components reduces the complexity and using weights allows for a faster convergence speed, which helps prevent the creation of improper cluster centers (Rajaby et al., 2016; Sai Kumar et al., 2011).

Moreover, the weighted image patch based FCM (WIPFCM), algorithm views each image patch, instead of each pixel, that is traditionally used to cluster an image. With the WIPFCM algorithm, one can reduce the presence of noise in an image that may affect the way the image segmentation is performed. Therefore, the WIPFCM algorithm can section images more precisely and is capable of handling any noise that may be present in an image, allowing one to cluster the image patch instead of a pixel itself (Ji et al., n.d.).

When using the fuzzy c-means in MATLAB, the function FCM is used to perform the algorithm. In MATLAB it is written as follows, [center, U] = fcm (data, clusters). The variable data is the data we are wanting to cluster. The variable clusters represent the number of clusters we are wanting to create. The function then returns the variable center, which is the matrix of cluster centers, that contains the individual cluster centers. The variable U is the results matrix (Elena, 2013).

In addition, we can use this function to perform image segmentation on MRI images in MATLAB. Fuzzy c-means is favored more than hard segmentations for anatomical structures, due to fuzzy c-means retaining more data from the original image. For example, if one wanted to segment a brain MRI into different categories based on tissue, we can do this by using fuzzy c-means, to take into account that more than one tissue class may be present in a single pixel, consequently having a class with more than one membership.

The problem that has been raised in this approach is the complication of integrating spatial information into fuzzy c-means (He et al., 2000; Selvakumar et al., 2012). When performing segmentation, identifying objects in the image is our objective. We can classify segmentation into two properties, discontinuity, and similarity. Designs based on discontinuity are called boundary-based methods. Methods that are based on similarity are referred to as region-based methods. However, these methods oftentimes fail to produce accurate segmentation. Due to a single method being unable to produce the expected results.

We can concur that, this is due to images having different properties and having other factors present in them such as noise (Argawal et al., 2010; Banaem et al., 2013). As previously stated, when we want to cluster our data, we want to divide the data into different sets. Another clustering algorithm we can use is k-means, which is an unsupervised clustering algorithm, that uses iterative methods, the value of each clustering center is updated until the most desirable clustering results are gotten. Yet, this clustering algorithm strictly identifies each data set into a certain class. Therefore, most objects do not have clear attributes.

The fuzzy k-means (FKM) clustering algorithm (Zhang et al., 2010) is better adapted to deal with a problem such as this. Even though FCM provides better clustering when performing image segmentation, it is not immune to having its flaws. As stated early a pixel can belong to more than one membership. When computing the Euclidean distance of a pixel, this can prove to be difficult, this can cause the centers of the clusters to be off, or our pixels to be clustered improperly (Hung et al., 2011).

This paper will introduce another way to use the fuzzy c-means algorithm to perform image segmentation in MATLAB to run the fuzzy c-means algorithm to perform image segmentation on MRI images.
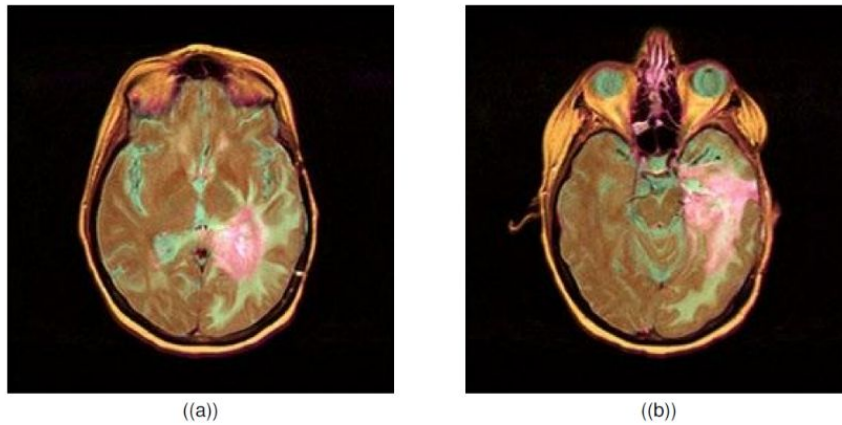
Figure 1: Brain MRI images.

FUZZY C-MEANS CLUSTERING ALGORITHM

What is Fuzzy c-Means

Fuzzy c-means is a way of dealing with inconclusiveness and unpredictability. This algorithm works by assigning memberships to each data point that corresponds to each cluster center in respect to the distance between the cluster center and the data point. FCM allows an object to belong to a set with a degree of membership ranging from 0 to 1. However, we can set restrictions in the algorithm to restrict memberships to be 0 or 1. In FCM, many instances arise that clusters are not well separated. Therefore, the classification of each object will be assigned in an arbitrary manner.

The process of FCM:

1) Specify a number of clusters N

   • Then assign them randomly to each point in the clusters.

2) Repeat the formula until the max number of iterations has been reached, or when the algorithm has converged

   • Compute the centroid for each cluster, using the formula below (Figure 2)

$$Y_m = \sum_{i=1}^{N} \sum_{j=1}^{C} M_{ij}^m \left\| x_i - c_j \right\|^2$$

Where
m- any real number greater than 1,
M_{ij}- degree of membership of x^i in the cluster j,
x_i- data measured in d-dimensional,
R^j- d-dimensional center of the cluster,

Figure 2: Mathematical representation of FCM (Selvakumar et al., 2012).

Application of FCM in the Image Segmentation Problem

In practice, FCM is a popular algorithm in image segmentation. For example, if we wanted to segment a medical image such as an MRI brain image, we can use FCM to do this. But how? As sated previously FCM allows data to belong to two or more clusters. With that being said, one can separate an image, such as a brain MRI, into clusters based on tissue membership. This becomes useful due to certain parts of image belonging to more than one tissue cluster (He et al., 2000), this allows one to better detect masses or tumors that may be present in the brain.

Method

The proposed approach was to use MATLABs built in FCM function to perform image segmentation, on the images that were provided. With MATLAB having this built-in function the primary goal was figure out how to handle clusters after the FCM function was ran. The images shown in Figure 1 were used to run the algorithm against. Once ran, the following images are produced as seen in Figure 3 and 4. In this section a breakdown of the code will
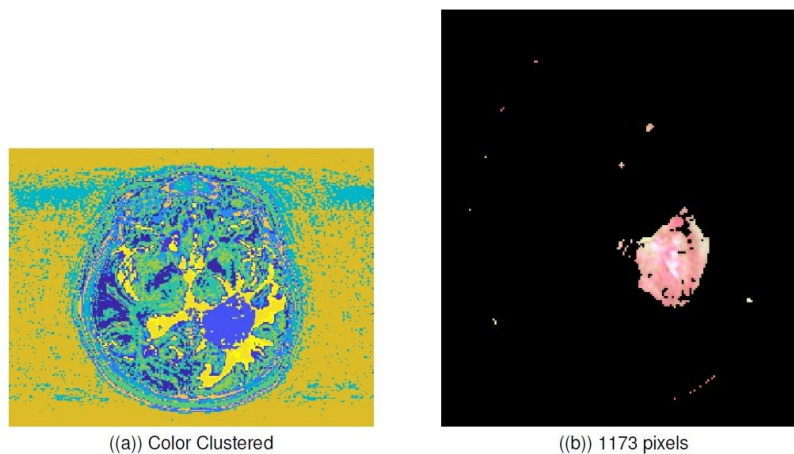

((a)) Color Clustered   ((b)) 1173 pixels

Figure 3: Brain MRI 1 clustered and segmentation.

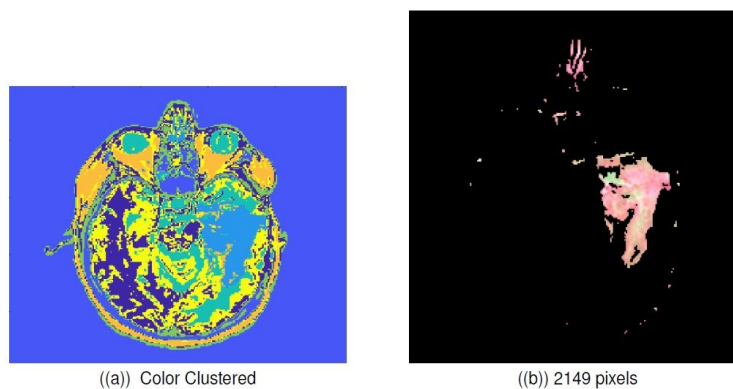((a)) Color Clustered   ((b)) 2149 pixels

Figure 4: Brain MRI 2 clustered and segmentation.

be discussed to provide better clarity of the proposed approach. Comments have been added to each line of code to better explain its function.

MATLAB Implementation

```matlab
1   function fuzzcmeans(imgName , numClust)
2
3   % fcm is random, so sometimes the tumor will fall into cluster ...
        4, other times into cluster 3, etc., so this just seeds the ...
        random number generator to have consistent output, which we ...
        will need later.
4   rng(0)
5
6   % Read in the image.
7   tumor = imread(imgName);
8
9   % Reshape the image into a matrix that has 3 columns, where each ...
        column represents a color band. Then do fcm with numClust ...
        clusters.
10  [¬, U] = fcm(double(reshape(tumor,[],3)), numClust);
11
12  % fcm outputs a matrix U which has as many rows as clusters. ...
        Here we want to assign the largest membership
13  [¬,clusters] = max(U);
14
15  % clusters is a vector that tells the cluster each pixel belongs ...
        to, so we convert it back into the shape of the original image.
16  clusters = reshape(clusters,size(tumor,1),size(tumor,2));
17
18  % This shows us a colormap of all the clusters. Each cluster ...
        will have a different color, so we can plot this and use it ...
        to find the cluster number of the tumor. If we change the ...
        number of clusters or the seed, we will need to check the ...
        cluster number of the tumor.
19  figure
20  imagesc(clusters)
21
22  % Get the indices of all pixels that are not part of the tumor ...
        cluster and turn them black.
23  [i,j] = find(clusters ≠ 3);
24  for k = 1:numel(i)
25      tumor(i(k),j(k),:) = 0;
26  end
27
28  % Converting the image to black and white to count the number of ...
        pixels in the tumor(easiest way)
29  bwimg = rgb2gray(tumor);
30
31  % Counting the pixels and ignoring the black pixels present in ...
        the image.
32  correctpix = length( bwimg(bwimg≠0) );
33
34  % Show the final image.
35  figure
36  imshow(tumor)
37
38  % Print in the console the number of pixels present in the tumor
39  fprintf('This tumor contains %d pixels ', correctpix)
40  end
```

RESULTS AND DISCUSSION

When running this function there are multiple clusters generated for each image. For figure 3(a), 7 clusters are needed to get the tumor. For figure 4(a),

we need 14 clusters to get the tumor. Therefore, the function must have an extra field to define the number of clusters. The function has a few caveats to note but the most important is the output of the clusters. When running the function, it can be seen that clusters are not outputted individually. In this approach we generated an image that colors all the clusters in a single image (Fig. 3 (a) and Fig. 4 (a)) to show how this algorithm constructs each cluster. Additionally, there is no image filter function before we run the function on these images. Having this function would denoise the image allowing a better center so our algorithm could be more efficient.

CONCLUSION

In this paper, we examined how fuzzy c-mean is used to solve the image segmentation problem, and the methodologies used to properly cluster data. A proposed solution to the image segmentation problem was also introduced. By using MATLAB and the built in functions one can isolate a tumor present in the MRI image. Then identify the cluster that the tumor exists in and then count the number of pixels in the tumor region.

REFERENCES

Argawal, P., Shriwastava, S. and Limaye, S. (2010). MATLAB Implementation of Image Segmentation Algorithms.

Banaem, H., Kermani, S., Sarrafzadeh, O. and Khodadad, D. (2013). An Improved Spatial FCM Algorithm for Cardiac Image Segmentation. *13th Iranian Conference on Fuzzy Systems*, pp. 1–4.

Elena, M. (2013). FUZZY C - MEANS CLUSTERING IN MATLAB. *The 7th International Days of Statistics and Economics*, 1(2), pp. 1–10.

He, S., Wang, X., Yang, Y. and Yan, W. (2000). MRI Brain Images Segmentation. *International Journal of Science and Research (IJSR)*, pp. 113–117.

Hung, C., Kulkarni, S. and Kuo, B. (2011). A New Weighted Fuzzy C-Means Clustering Algorithm for Remotely Sensed Image Classification. *IEEE Journal of Selected Topics in Signal Processing*, 5(3), pp. 543–553.

Ji, Z., Xia, Y., Chen, Q., Sun, Q., Xia, D. and Feng, D. (2019). *Fuzzy c-means clustering with weighted image patch for image segmentation*. K. Elissa, "Title of paper if known," unpublished.

Rajaby, E., Ahadi, S. and Aghaeinia, H. (2016). Robust color image segmentation using fuzzy c-means with weighted hue and intensity. *Digital Signal Processing*, 51, pp. 170–183. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp. 68–73.

Sai Kumar, T., Chandra M, M. and Murthy P, S. (2011). COLOUR BASED IMAGE SEGMENTATION USING FUZZY C-MEANS CLUSTERING. *International Journal on Intelligent Electronic Systems*, 5(2), pp. 47–51.

Selvakumar, J., Lakshmi, A. and Arivoli, T. (2012). Implementation of Brain Tumor Segmentation in brain MR Images using K-Means Clustering and Fuzzy C-Means Algorithm. *International Conference on Advances in Engineering, Science and Management*, pp. 186–191.

Zhang, J., Ha, M. and Wu, J. (2010). IMPLEMENTATION OF ROUGH FUZZY K-MEANS CLUSTERING ALGORITHM IN MATLAB. *Ninth International Conference on Machine Learning and Cybernetics*, pp. 2084–2088.