Project Module 1 : Image Segmentation

Bahnisikha DUTTA, Mathieu LAPEYRE, (Yi-Pu ZHAO)

Georgia Institute of Technology, ECE 6780, Group 9

Atlanta, GA

***Keywords : Colormap Normalization; LDA ;Level Set; Graph Cut; Rand Index; Inter Index; Intra index.***

# CLINICAL NEED

Labeling a histopathology image as having cancerous regions or not is a critical task in cancer diagnosis [2]. Existing image classification techniques require detailed manual annotations for the cancer pixels, which are time-consuming to obtain. With the dramatic improvements in computational power, computer-assisted diagnosis has been made easier. Today, pathologists need the support of a clinical decision system for diagnosis and prognosis of cancer.

# PROBLEM STATEMENT

The first thing to deal with for the project is the image segmentation. The goal of segmentation is to simplify the representation of an image into something that is more meaningful and easier to analyze for a computer [1]. Humans can easily interpret the semantic meaning of an image and distinguish objects between each other. This is something a computer can not achieve without some pre processing. We need to design algorithms to help the computer assign label to each pixel of the image such that pixels with the same label share certain characteristics and represent a meaningful object.

The main goal of this first module is to develop an automatic segmentation of nuclear structures which can detect three different regions :

* ribosome and nuclei
* cytoplasm
* background

To achieve such an automatic segmentation, we need to perform a preprocessing, such as getting rid of the outliers and normalizing the images according to a common representative color distribution. We then, need to implement two categories of color segmentation techniques and then compare them using objective performance metrics.

# RELATED WORK

Our segmentation approach is based on several papers that have been published for the last 10 years. We focused on papers which present differents techniques about normalization, supervised segmentation, unsupervised segmentation and performance evaluation. We reviewed several papers to be able to choose the most efficient techniques for each of these domains. You can find the results of these reviews, and the selected method for each domain in the Figure 1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Strategies** | **Papers** | **Location** | **Strength/Weakness** | **Selected** |
| Normalization | • Reinhard’s Method [11]  • CDV-L [12]  • CDV-MM [12]  • Colormap Normalization [4] | Hoffman et al 2014  Reinhard et al 2001  Magee et al. 2009 | University of UTATH, IEEE Computer Graphics and Application Journal.  University of Leeds, MICCAI Workshop. | Weakness: 1)highly dependent on reference batch selection  2)First two approaches introduces high background color distortion  Strengths:Easy and efficient algorithms | Our own method: Statistical Colormap Normalization  Rationale: Does not require manual selection of reference image/ batch.  Does not produce background artifacts |
| Supervised | • LDA classification, LDA reclassification in sample color space [16]  • Supervised grayscale conversion, learning based marker detection and classification, watershed segmentation[20]  • Bayesian/kNN/ SVM training, Color watershed [18] | Kothari et al.  2011  Mao et al. 2006  Meurie et al.  2003 | Journal of the American Medical Informatics Association.  IEEE Transactions on Biomedical Engineering Journal.  Signal Processing and Information Technology Journal. | Weakness: 1)Highly dependent on ground truth selection  2)Requires a lot of manual effort and time  3)Training classifiers takes significant time  Strengths: Produces better segmented results than unsupervised | LDA classification followed by Level Set method  Rationale: fast, worked well for small reference dataset |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Strategies** | **Papers** | **Location** | **Strength/Weakness** | **Selected** |
| Unsupervised | • Hierarchical Self-Organizing Map [17]  • EM algorithm with the Fisher-Rao criterion as its kernel [16]  • K-means clustering in the La\*b\* color space  • Kernel graph cut [8] | Datar et al.  2008  Jun et al. 2007  Sertel et al. 2009  Salah et al. 2011 | IEEE International Symposium on Biomedical Imaging.  IEEE Transactions on Image Processing.  J. Signal Proc.. Systems Journal. | Weakness: 1) Lack of in-depth prior knowledge for the segmentation task  2) Heavily rely on strong assumptions on data distribution, e.g. Gaussian | K-Means followed by Kernel Graph Cut Method  Rationale: K-means generates good initial segmentation fast; Kernel Graph Cut perform well generally (better than traditional Graph Cut) |
| Performance Evaluation | • Foreground/Background variance [15]  • Rand Index [9]  • Squared color error [14]  • Entropy based evaluation [13] | Unnikrishnan et al.  2007  Zhang et al. 2008  Wagner et al, 2007  Chabrier et al. 2004  Yang et al. 2007  Ouattara et al., 2011 | IEEE Transactions on Pattern Analysis and Machine Intelligence.  Computer Vision and Image Understanding Journal.  Cranio-Maxillo-Facial Surgery Journal.  Asian Journal of Scientific Research. | Weakness: 1) Rely on strong assumption on the distribution. Can lead to negative value.  2) Many foreground/background evaluation.  3) Features error (color, texture, …) on the overall region instead of region boundary. | Supervised : Clusters similarity with Rand Index.  Unsupervised : individual per-region measures, weighted by boundary length and region size (Zeb). |

Figure 1: Paper review

# OUR APPROACH

Our approach to the problem at hand can be described by the following flow chart:

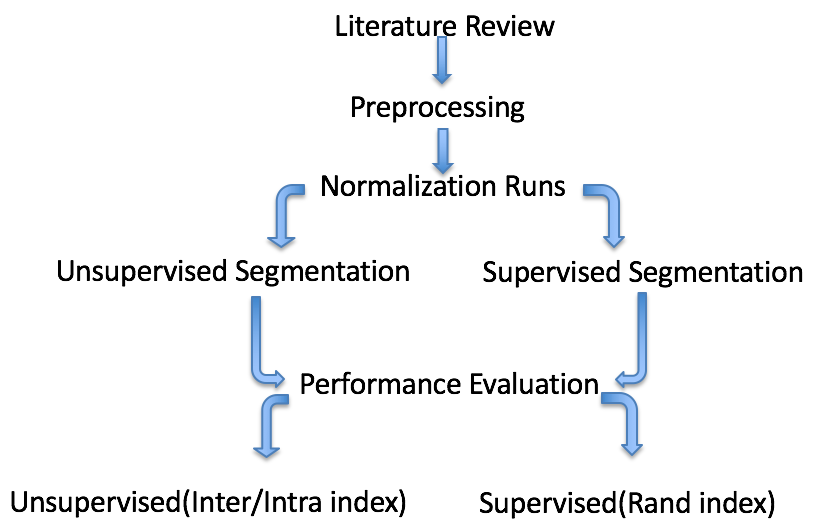


Figure 2: System design flowchart

Each of these steps are explained at length in the following sections:

## PREPROCESSING

Before dealing with the segmentation itself, there is some preprocessing to do. We need to get rid of the images that differ too much from the other images. We also need to normalize the remaining images to guarantee a similar color distribution between all of them.

To get rid of the outliers, we implemented an objective statistical method. For each image of a dataset, we computed its colormap and evaluate the number of unique colors which make up the image. Then, we compute the mean value of this number as well as the standard deviation across the dataset.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| outlier | - std |  | m |  | * std | outlier |

Figure 3: How to get rid of the outliers

Every image whose number of unique color does not belong to the interval [m - std; m + std] is considered as an outlier and the decision making algorithms like ground truth preparation/normalization will not take into account such an image. For Dataset 2 and 3, we carried out the above process per batch of images assuming that in the images named as TCGA-B0-4693\_15.png(for example); TCGA-B0-4693 denotes the name of the batch and all the images starting with this prefix belongs to the same batch.

## NORMALIZATION

After careful study of the existing literature, we drew inspiration from the colormap normalization technique from Kothari et al 2011[4] and came up with our own technique referred to as Statistical Colormap Normalization from now onwards.

For Dataset 2/ 3: Assuming as before, that in the images named as TCGA-B0-4693\_15.png (for example); TCGA-B0-4693 denotes the name of the batch, we have batches of 16 images each in these datasets. After outlier removal, not all batches will have 16 images left. However, we can find the colormap out for each of these batches and then the unique number of colors contained in each of these batches. This no. of unique colors is then, divided by the total no. of images in each of these batches to give a normalized value. Then, we find out the mean and deviation of this normalized value across all the batches. Let’s call this mean and std as mean\_target and std\_target. Now, for a source image which is to be normalized, we find out the total no. of unique colors in its batch and divide it by the total no. of images in the batch(source\_uniquecolor) . Now, if source\_uniquecolor belongs to the interval [mean\_target - std\_target; mean\_target + std\_target] , we map the source image to the mean\_target using color normalization technique introduced in Kothari et al 2011[4]. Rest of the images are mapped to their closest deviation of mean. The flowchart for the normalization process is as shown below:

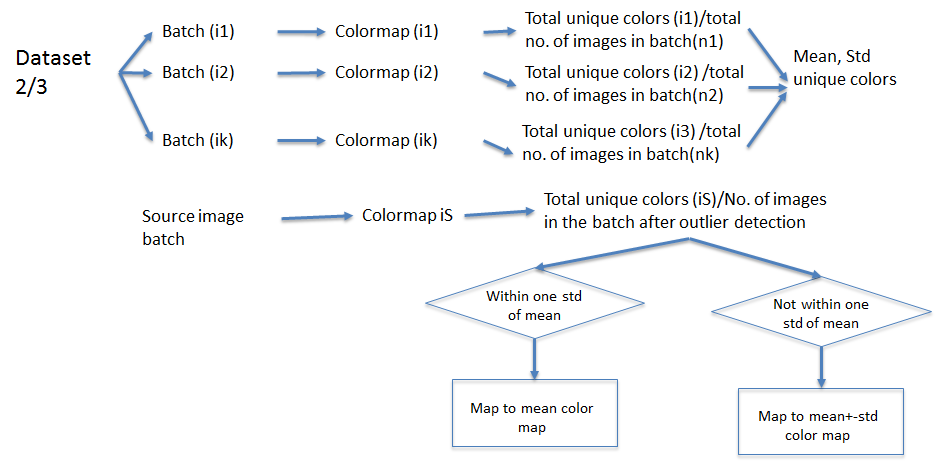


Figure 4: Batch normalization approach

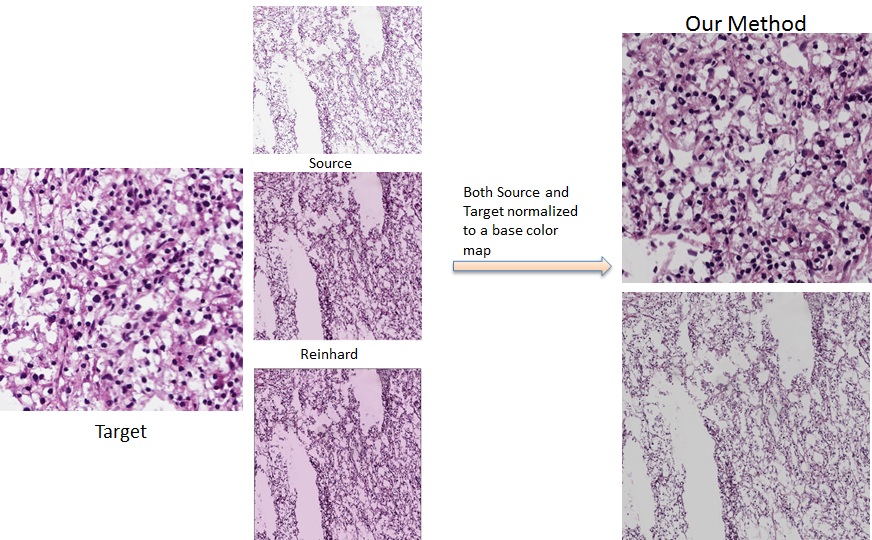
For Dataset1, since we had no prior knowledge about the batches, we computed the colormap of each images in the entire dataset and then computed the mean and deviation of the same and followed the same procedure for normalization as before but mapped it to the mean colormap across the entire dataset rather than the batches. Our normalization technique does not need the careful selection of reference images/ image batch and does not introduce artifacts like the ones we can see from the other methods as shown below:

Figure 5 : Normalization results

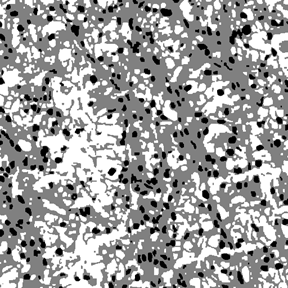
## SEGMENTATION

We carried out two segmentation algorithm for the data provided. Each of these methods are described in details in the following sections.

### SUPERVISED SEGMENTATION METHOD

Our Supervised segmentation algorithm uses an LDA classifier to generate classifiers from a set of ground truth labels (called Training Set). The classifier is then used to test on another set called the Testing Set. The generated segmented images are then smoothed out using Level Set algorithm. Each of these steps are explained in details below:

**Ground Truth Generation:** Ground truth is generated in a semi-automatic way. Ideally, one can manually define all the segments in one image: which are the segments, how many pixels in each segment, etc. However, it is extremely time-consuming and tedious. Instead, based on our experiment on color distribution of each segment component, there is a clear difference in term of color for different segments. Therefore, we decide to build a tool to manually select the typical color for each segment component, e.g. nuclei, cytoplasm and background. After that, pixels in the same image can be assigned to different segment component based on the corresponding distance (e.g. L2 distance) to typical color. However, the color-distance based method has the same limitation as other local segmentation such as K-means: the result is not smooth enough due to lack of consideration on context information. To cope that, we add a graph cut algorithm to smooth out the result of color-distance based method. The combination of these two gives us superior results in term of accuracy and smoothness, as illustrated in the following images. We use them as ground truth.



Original Image Generated Ground Truth

For each dataset, we generate ground truth for ~10% from all images (50 for dataset 1, 160 each for dataset 2 and 3). It is worth mentioning that, even with manual input during the generation of ground truth, there are still some hard cases with poor ground truth results. In order to keep the quality of ground truth data, we can manually review all the generated ground truth and remove those ones with poor quality. However, since we had already removed the outliers, our generated ground truth were decent enough. We then, performed LDA classification on these generated ground truth images. Linear discriminant analysis (LDA) is a generalization of Fisher's linear Discriminant [16], a method used in statistics, pattern recognition and machine learning to find a linear combination of features that characterizes or separates two or more classes of objects or events. The resulting combination may be used as a linear classifier. For each Dataset, ~ 10% images were used for ground truth generation. Amongst these images, some of them were used to train the classifier and some were kept as a testing batch.

For example, For Dataset1:

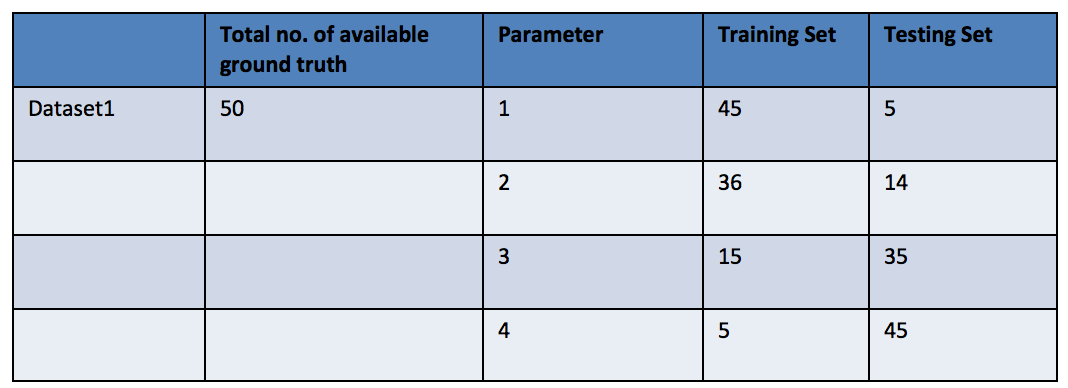
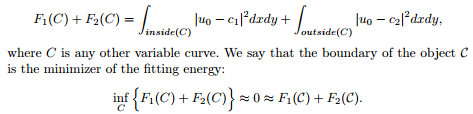


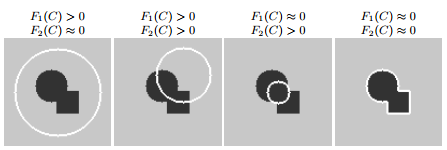
Figure 6: Parameters for the supervised segmentation method

The trained classifiers were used to do the initial segmentation on the images. This was followed by a level set algorithm to smooth out the segmented images.

**Level Set Method:** Level set method has been well developed for medical image segmentation for decades. Unlike most of the image segmentation methods that operate on discrete image domain, level set method perform segmentation on the continuous spatial domain, therefore achieving a sub-pixel accuracy. This advantage is especially favored by medical image processing, since the accuracy of segmentation results is highly desired. Various extensions of level set have been proposed for specific tasks. In this project, we choose the Chan-Vese method [5], which is an extension of level set without using edge explicitly. The main reason of choosing Chan-Vese method here is to avoid explicit detection of edges on microscopy image, which could be quite trivial and challenging. Instead, the Chen-Vese method is minimizing an energy function:



One example on how Chan-Vese method works is illustrated below. Given some initial segmentation curve C, Chan-Vese method tries to minimizing the energy function above. Specifically, by minimizing the inside term F1(C), the curve C tends to shrink; meanwhile, by minimizing the outside term F2(C), the curve C tends to expand. These two tendency reach to a balance when the curve stays at the boundary between image foreground and background. Therefore, the segmentation of 2-phase can be solved in this way.



In our project, we refer to Vese et al. 2012 [6] to extend the Chan-Vese methods for multi-phase segmentation (since there are more than 2 components to be segmented in the image).

### UNSUPERVISED SEGMENTATION METHOD

We use k-means clustering followed by kernel graph cut for unsupervised image segmentation [7]. K-means clustering is a fast and mature segmentation approach, especially when the number of k is known. However, K-means has its limitation: there is a lack of consideration of context for image segmentation. For example, neighboring pixels in one image tend to belong to the same segment component such as nuclei.

In order to embed the context into image segmentation, we turn to the graph cut image segmentation, which is a well-known state-of-art method with strong modeling of context. Graph cut has been well developed for image segmentation task. In our application, graph cut is favored for three reasons:

1) it has very fast inference, e.g. ~5 iterations to converge in our experiment;

2) incorporate context information explicitly and

3) be able to incorporate recognition or high-level priors, e.g. nuclei detection results.

However, there is a need of special design on unary terms for graph cut. If we look into the general formula of graph cut [19] :



where L is the set of segmentation labels for all pixels in the image. Unary term R(L) incorporates the local (pixel) information into the segmentation, and piecewise term B(L) incorporates the context (between-pixel) information into segmentation, α is the relative importance factor between these two terms. An example of unary and piecewise term definition can be found below:

One can also check the illustration below to get a general idea on graph cut image segmentation:

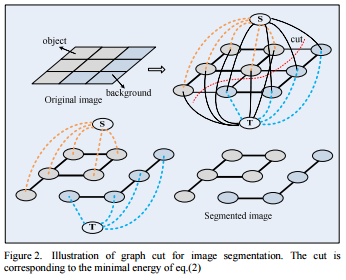


Figure 7 : Illustration of graph cut for image segmentation

In this project, we choose an extension of general graph cut, named kernel graph cut [8], for image segmentation. Specifically, a RBF kernel function is added to the unary term as well as the piecewise term of general graph cut. Compared with general graph cut, kernel graph cut releases the constraint of Gaussian distribution in piecewise model (e.g. assume the color between adjacent pixels follow a Gaussian distribution). Therefore, we can obtain better segmentation performance with such an extension.

## PERFORMANCE METRICS

When dealing with image processing, it is always difficult to assess whether one segmentation algorithm produces better results than another. The overall result of such algorithm is highly object-dependent. According to what the end-user wants, so we must choose the most appropriate performance metric to objectively evaluate the output of a segmentation algorithm.

Today, the most common evaluation tool is the human eye, which can visually analyze the result of a segmentation method. However, this is a tedious and time-consuming process which limits the evaluation of segmentation algorithm to only a small number of segmented images. This is why we decided to implement both a supervised performance metric, which compare a ground-truth segmented image to our own segmented image, and an unsupervised method which does not rely on any reference image. We referred to a really well built review paper, Hui Zhang et al. [3] which did a good job of listing out the existing unsupervised evaluation methods and their strength and weaknesses.

The main issue when evaluating a segmentation process is the fact that the segmentation itself highly depends on the purpose of the segmentation. Some metrics described in [3] deals with grayscale images, while others focus on foreground/background delimitation.

### SUPERVISED EVALUATION METHOD

We decided to implement the Rand Index evaluation metric [9]. The Rand Index is a purely statistical metric that compare the ratio of the number of pairs of points having the same label relationship in cluster S (from ground-truth method) and cluster S’ (obtained through segmentation method).

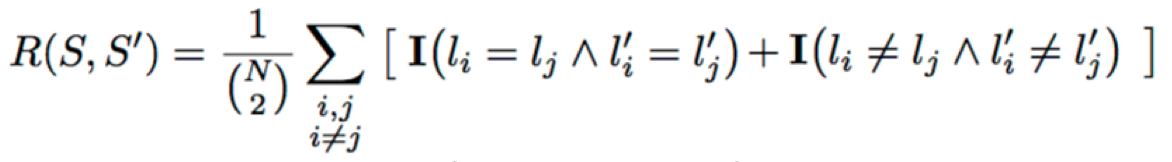


Figure 8: The Rand Index formula

### UNSUPERVISED EVALUATION METHOD

When we do not have any ground-truth to compare our results with, we can rely on the unsupervised evaluation metrics. We decided to implement two unsupervised evaluation metrics. The first one is called the intra-region uniformity criterion of Levine and Nazif [10]. The intra-region uniformity is translated here by the normalized variance of colors inside each region. Each region has a proper contribution, depending on the number of pixels which make up the region. Let the value of feature F at each pixel i (the color feature) belonging to region Ri be fi. Aj is the region size of the segment j. The variance of the feature F is given by sigma. Then we can compute the uniformity of the image according the sum of the contribution of each region.

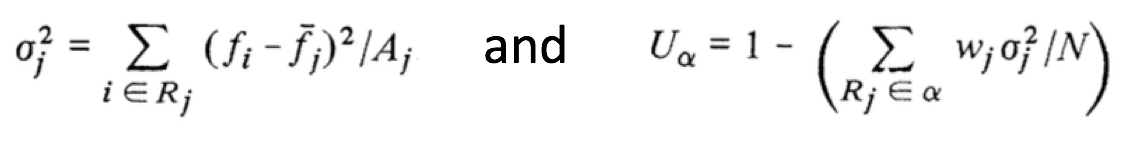


Figure 9: The Intra-Region uniformity

Assuming that all regions are perfectly homogeneous, the test is 1 since the variance of each region is zero. The criterion decreases in presence of inhomogeneity.

The second metric is called the inter-region contrast of Levine and Nazif [10]. Contrast can be computed on the basis of the average color values of pixels from different regions at the boundary. Each region contributes to the inter-region contrast according to the length of its boundary. The contrast cij between two regions Ri and Rj can be express as below:

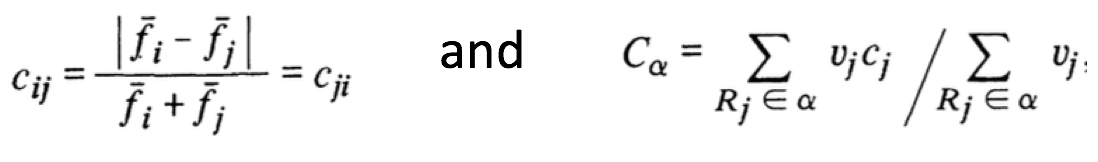


Figure 10: The Inter-Region Contrast

# RESULTS

This section will report some representative images of our results.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Reference | Unsupervised seg. | Supervised seg. |
| Necrosis\_1 |  |  |  |
| Stroma\_18 |  |  |  |
| Tumor\_20 |  |  |  |

Our implementations are comparatively fast and the speed for each step is tabulated as below:

|  |  |
| --- | --- |
| **Outlier Detection** | **~3images/sec** |
| Normalization | ~3images/sec |
| LDA classifier Generation | ~1.25images/sec |
| LDA based segmentation | ~0.77images/sec |
| Level Smooth Algorithm | ~0.19images/sec |
| Unsupervised Method | ~0.2images/sec |
| Performance Metric | ~0.5images/sec |

# EVALUATION

This section will report the performance of our segmentation algorithms on Dataset1. We started with plotting the scatter plots for our manually segmented ground truth for the different classes in two different colorspace. We then plotted out the scatter plots for the supervised/unsupervised segmented images. These plots are shown below:

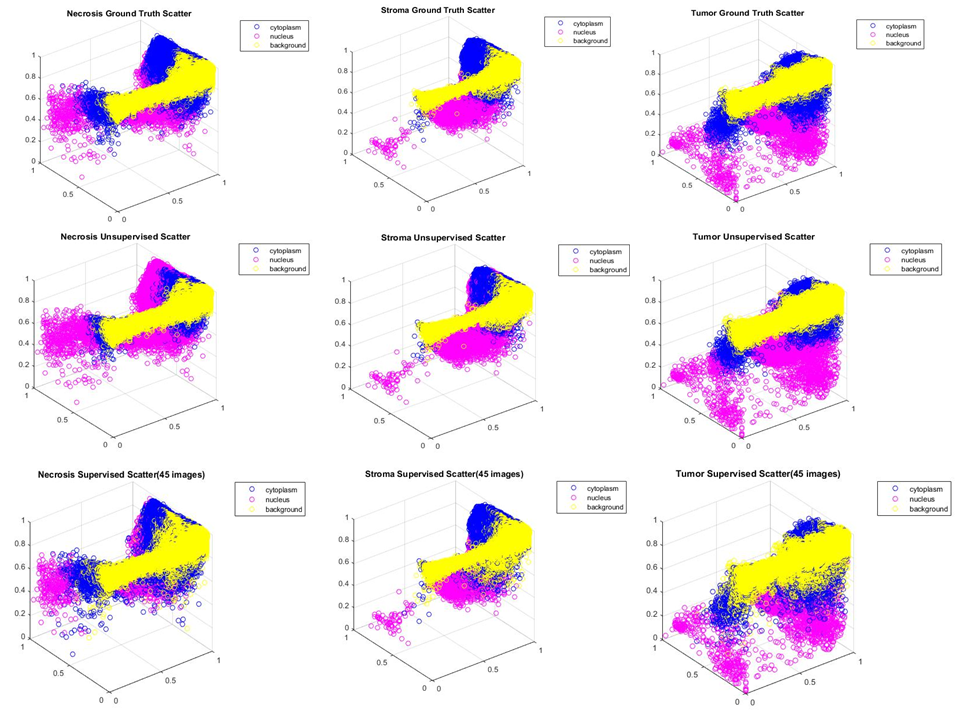
## RGB COLOR SPACE

## 

As you can see from the above figures, our ground truth images for all the three classes i.e Necrosis, Stroma and Tumor has minimal overlap between the three different clusters. The overlap of regions is the least in the case of Tumor as compared to other classes for unsupervised segmentation. However, the resemblance to ground truth is lower in the case of unsupervised than in supervised for all the classes. A lot more cytoplasm color values has been assigned to the nucleus in case of unsupervised or in other words, there is significant under segmentation. On the other hand, the overlap in between regions is comparatively high for the case of supervised in this color space. This might be due to the limited availability of ground truth images for training.

## HSV COLOR SPACE

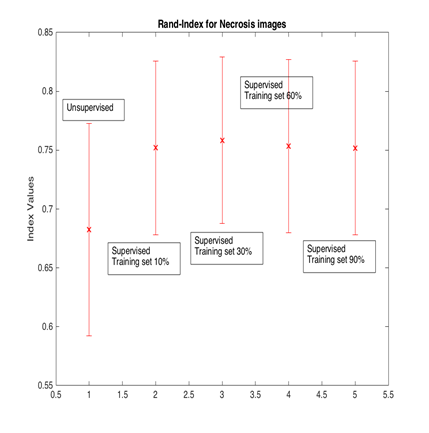
The same analysis was carried in the HSV space and showed similar features:



## RAND INDEX

The Rand index evaluation over the entire classes and algorithms generates the following results:





## INTERPRETATION

It is quite evident from the above figures that the overall performance of all the algorithms (with varying parameters) is the best for Tumor images and the worst for Necrosis on an average. The supervised algorithms gives consistent performance over all data types and parameters and thus can be considered stronger than the unsupervised algorithms. The unsupervised algorithms give high variation for the Stroma class whereas minimal variation for Tumor class and can be thought of as class dependent.

# THE GUI

The targeted end-users of this project are the pathologists who want to get support when diagnosing cancer images. To this end, we need to design a GUI that can help the pathologist analyzing several images in an appealing way. The parameters described above for each segmentation technique must be accessible and the interface should be user friendly.

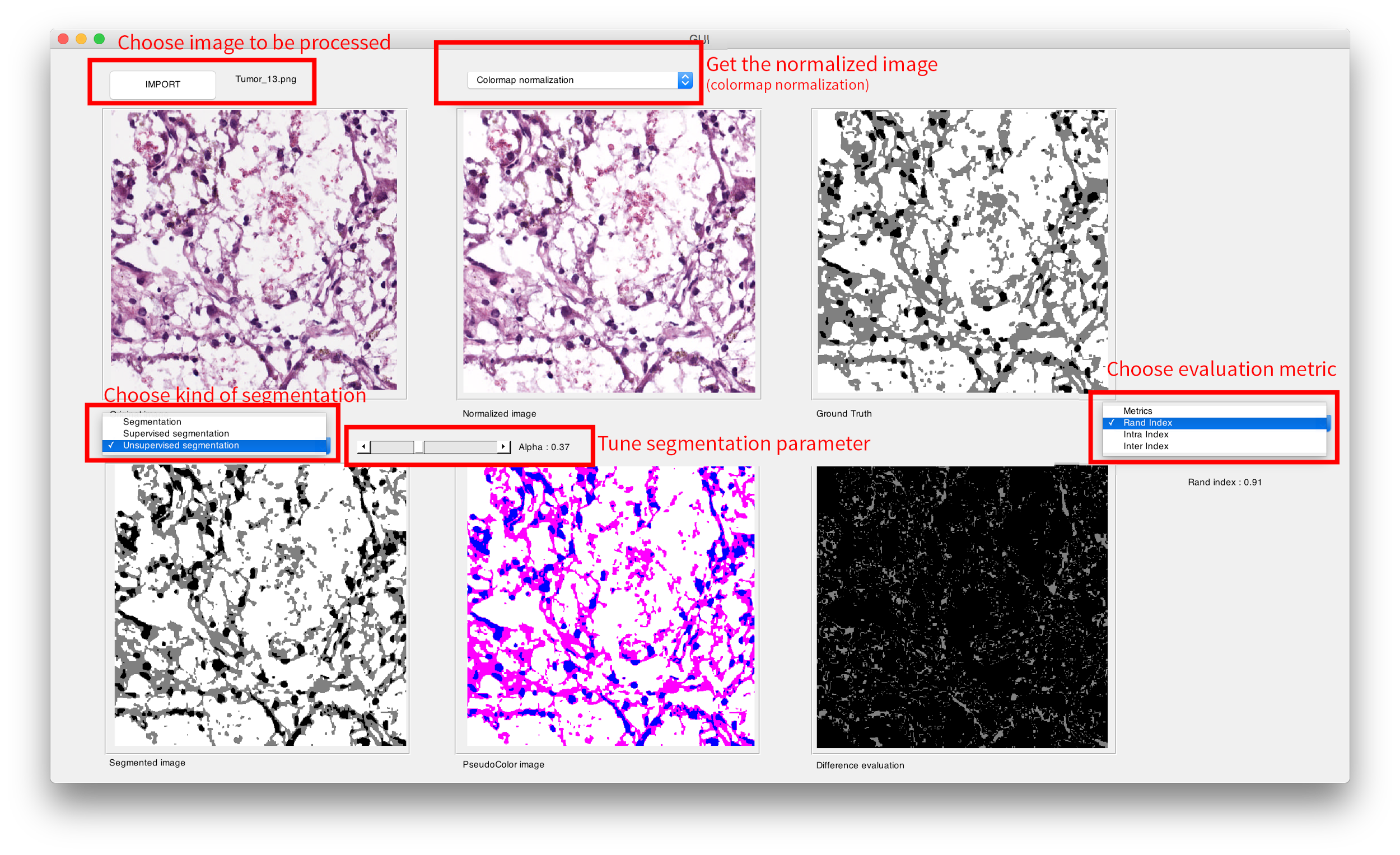


Figure 11: The interactive GUI

The interactive feature of the GUI let you choose the segmentation or tune the parameter. For each change, the data automatically updates and a new result is displayed.

# FUTURE WORK

Labeling a histopathology image in cancer diagnosis, and manual image segmentation is a highly time-consuming task. We implemented two color-based segmentation methods, according to research that have been made over the past decade.

The complexity of the approach to make histological images results in variable staining and illumination variations. This is why some preprocessing is needed before applying the segmentation algorithms. We understood that histopathological image segmentation raises more complex challenges than classic image segmentation. This is why robust image segmentation algorithms, which can be really effective on common images, do not always produce good results for histopathological images.

We found that segmentation algorithms can be class-dependent, which mean that one algorithm may perform better on one type of cancer, but not on another type. To improve our algorithms, we must rely on accurate metrics. These metrics also depend on the kind of images we want to analyze. Both our segmentation algorithm and evaluation method rely on image color. However, we can find new way to process an image, by paying attention to object-based algorithm. For example, we can rely on a semantic metric (such as shape) that would perform better for our specific need (circular nuclei, large cytoplasm regions, …).

# References

1. Linda G. Shapiro and George C. Stockman, “Computer Vision”, pp 279-325, New Jersey, Prentice-Hall, 2001
2. Yan Xu, Weakly supervised histopathology cancer image segmentation and classification, 2014
3. Hui Zhang et al., Image Segmentation Evaluation : a survey of unsupervised methods, 2007
4. Kothari S, Phan JH, Stokes TH, Wang MD. Pathology imaging informatics for quantitative analysis of whole-slide images. Journal of the American Medical Informatics Association. 2013
5. Tony Chan, Luminita Vese, “An active contour model without edges”, Scale-Space Theories in Computer Vision 1999
6. Tony Chan, Luminita Vese, “A multiphase level set framework for image segmentation using the Mumford and Shah model”, International journal of computer vision, 2012
7. Olcay Sertel et al., “Histopathological Analysis Using Model-Based Intermediate Representations and Color Texture”, Journal of Signal Processing Systems, 2009
8. Mohamed Salah et al., “Multiregion image segmentation by parametric kernel graph cuts”, IEEE Transactions on Image Processing, 2011
9. R. Unnikrishnan et al., “Measures of Similarity”, IEEE Workshop on Applications of Computer Vision, 2005
10. M. Levine, A. Nazif, “Dynamic Measurement of Computer Generated Image Segmentations” IEEE Transactions on Pattern Analysis, 1985
11. Reinhard E, Adhikhmin M, Gooch B, Shirley P. Color transfer between images. Computer Graphics and Applications, IEEE. 2001
12. Magee D, et al. Colour normalisation in digital histopathology images. 2009
13. H. Zhang, J. Fritts, S. Goldman, An entropy-based objective evaluation method for image segmentation, in: Proceedings of SPIE- Storage and Retrieval Methods and Applications for Multimedia, 2004.
14. J. Liu, et al, Multi-resolution color image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1994
15. J. Weszka, A. Rosenfeld, Threshold evaluation techniques, IEEE Transactions on Systems, Man and Cybernetics, 1978
16. R.A. Fisher, “The statistical utilization of multiple measurements,” Annals of Eugenics, 8:376-386, 1938.
17. Meyer, F, “Topographic distance and watershed lines”, Signal Processing, 1994
18. M. Datar et al., “Color and texture based segmentation of molecular pathology images using HSOMS”, Biomedical Imaging: From Nano to Macro, 2008
19. Faliu Yi et al,. “A survey of graph-cut methods”, International Conference on Systems and Informatics, 2012
20. K. Mao et al. “Supervised Learning-Based Cell Image Segmentation for P53 Immunohistochemistry”, IEEE Transactions on Biomedical Engineering, 2006