# EE 634-Digital Image Processing Term Project

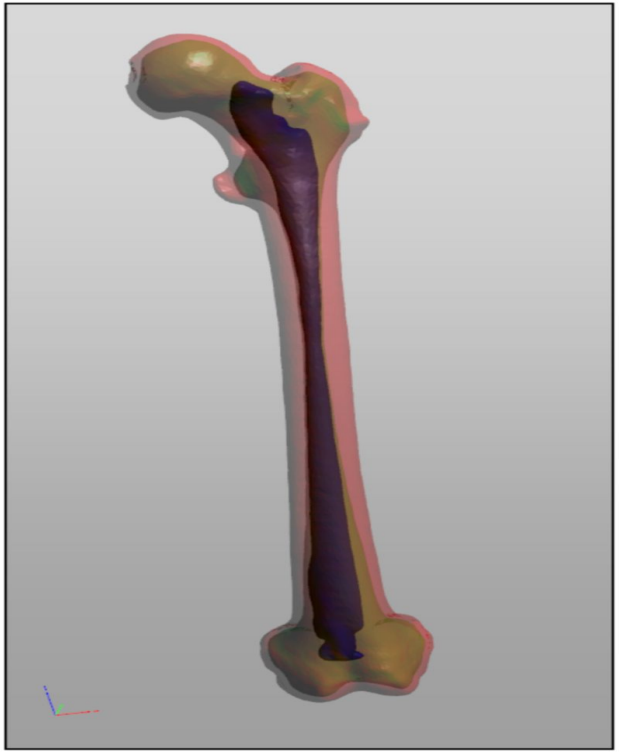
Alican Hasarpa
Bora Baydar
Volkan Okbay

# Outline

- Image Segmentation
- Edge Based Segmentation
- Color Based Segmentation Using K-means Clustering
- Watershed Segmentation
- Summary

# Image Segmentation

- The purpose is to change and simplify the representation of the image for better understanding and making easy to analyze
- The pixels in the same segment show similar characteristics
- Basic process of image analysis; finding the region of interest of the more complicated analysis processes; thus it is very important



Picture taken from: https://en.wikipedia.org/wiki/Image_segmentation

# Image Segmentation

Applications:

- Machine Vision
- Medical Imaging
- Object Detection
- Traffic Systems
- Analysis of Astronomical Data

# Image Segmentation

Some image segmentation techniques:

- Thresholding
- Edge detection
- Region based methods
- Histogram based methods
- Clustering
- Partial differential equation based methods
- Watershed transformation
- Graph based methods
- Pixon based methods

# Edge Based Segmentation

In this image segmentation technique, region of interests are determined by finding the boundaries within the image. Possible edge detection algorithms:

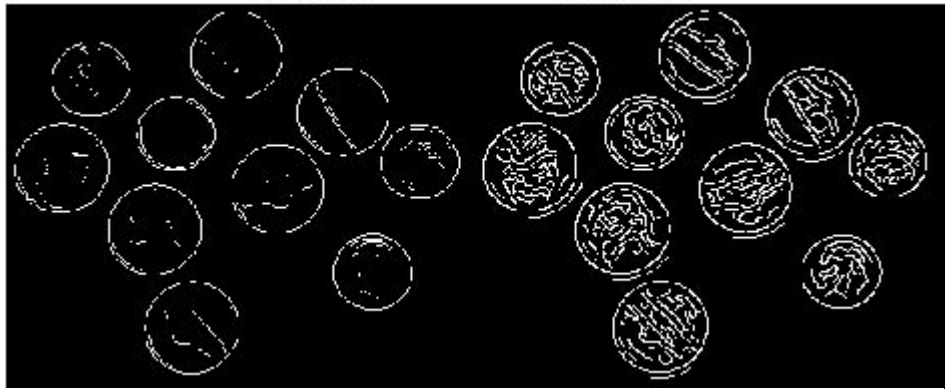
- Sobel
- Canny
- Roberts
- Prewitt
- Fuzzy Logic Methods
- Laplacian of Gaussian

# Edge Based Segmentation

Original Image

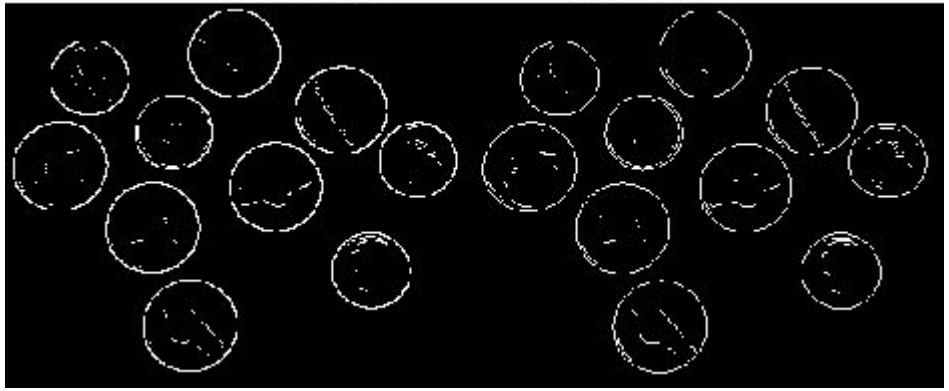
Sobel Filter

Canny Filter

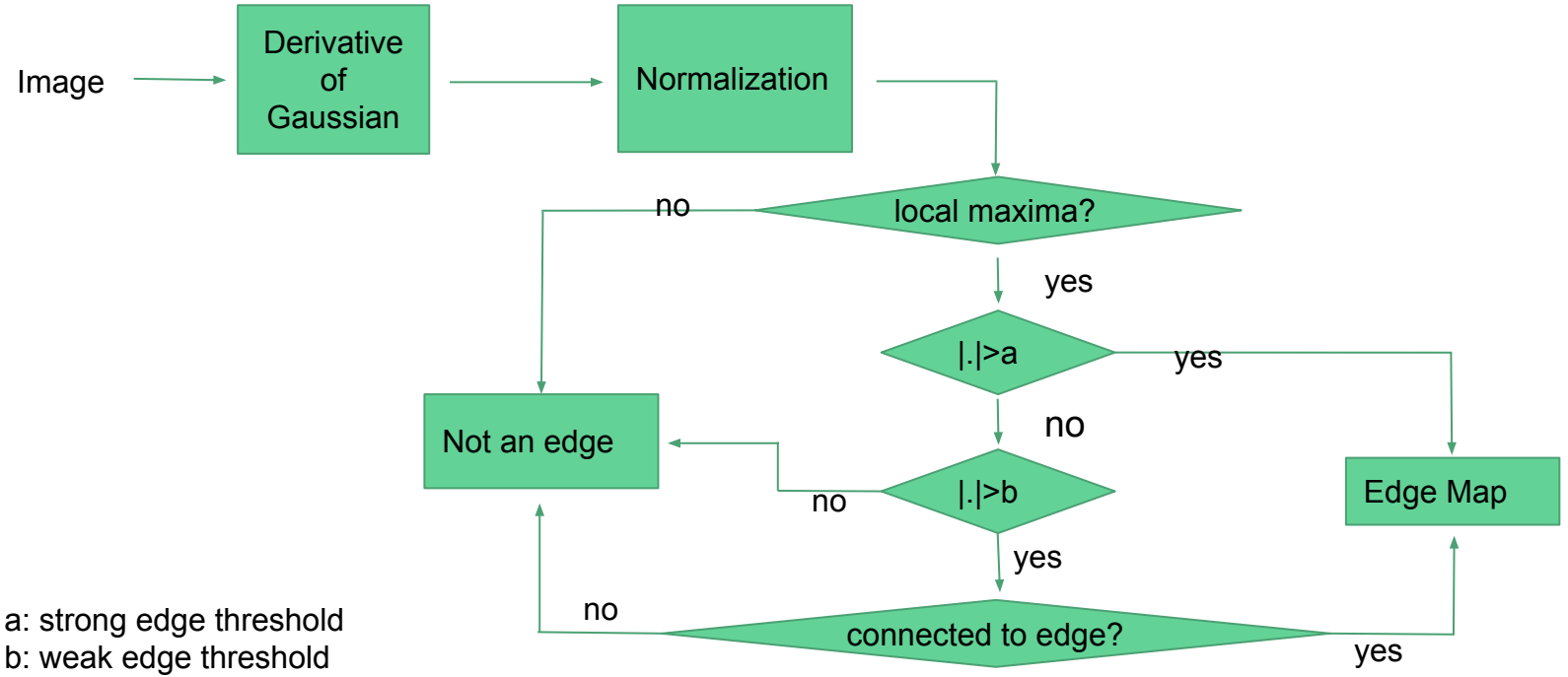


Roberts Filter

Prewitt Filter



# Edge Based Segmentation - Canny




a: strong edge threshold
b: weak edge threshold

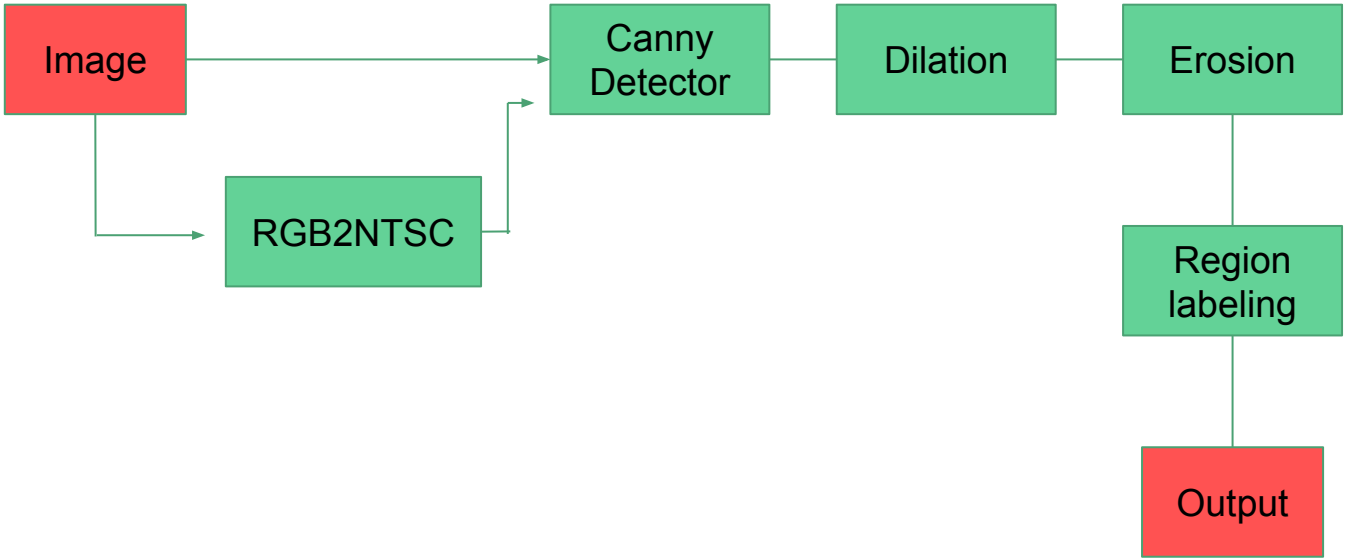
# Edge Based Segmentation

The biggest problem is open contours which are formed when the edge detection algorithm fails. We used 3 steps for getting better performance:

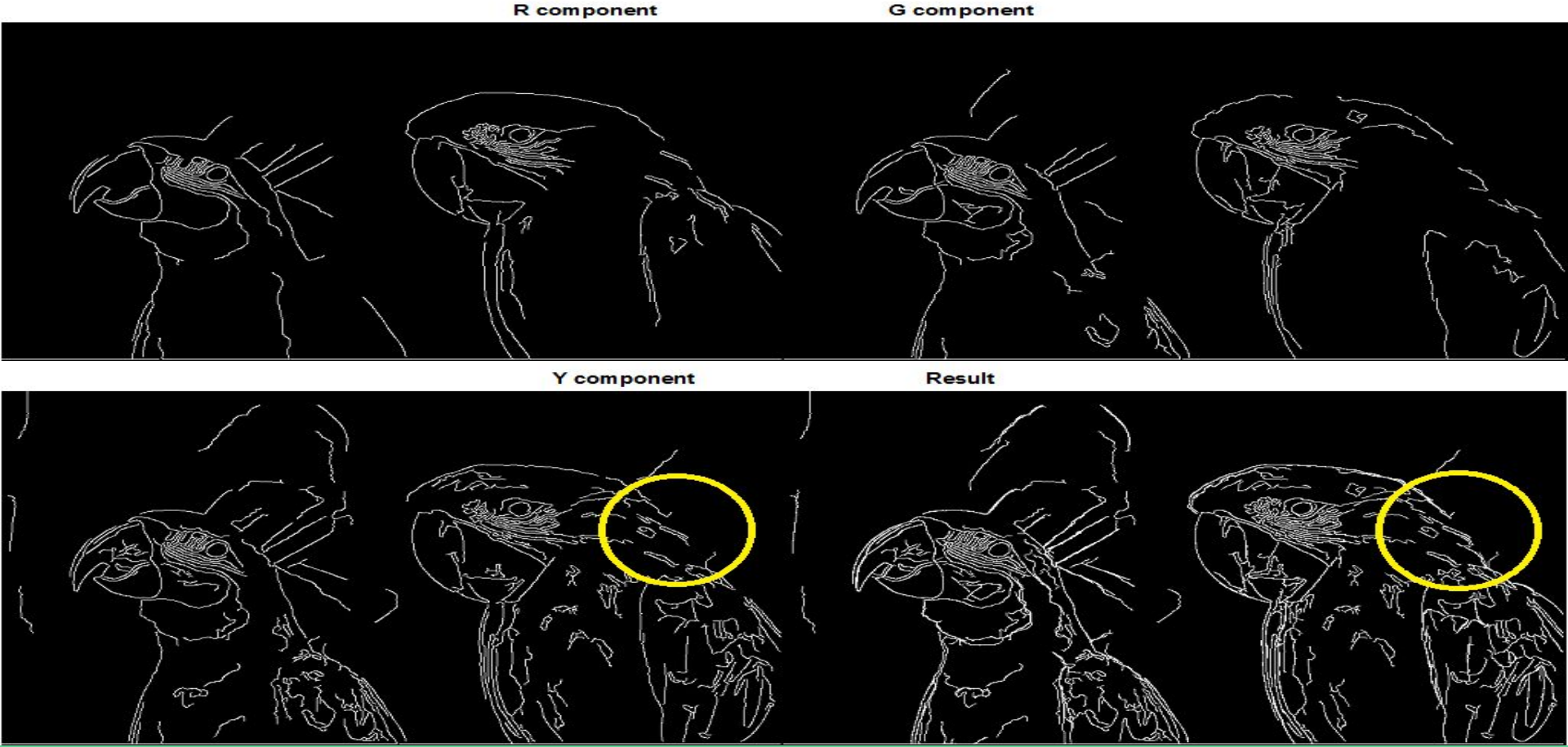
1) Use R, G, B and Y components of the same image instead of only one of them. Apply Canny filter for every matrices. Merge every result for less edge loss.
2) Use first dilation then erosion; but use larger structure element for dilation and smaller element for erosion which results with emphasizing edges and improvement in the open contours.
3) Use 4 -pixel -connectivity for deciding which pixel belongs which segment

# Edge Based Segmentation

Overall algorithm:

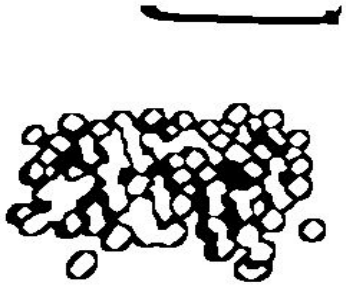


# Edge Based Segmentation

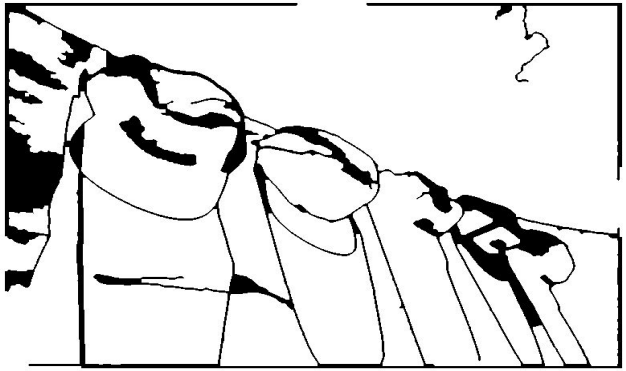


# Edge Based Segmentation

# Edge Based Segmentation



# Edge Based Segmentation




# Edge Based Segmentation

PROS:

- Relatively fast algorithm
- Successful for object detection

CONS:

- No automatic algorithm is possible for all images
- Problem with edges which are not continuous

# Color Based Segmentation Using K-means Clustering

**What is k-means clustering ?**

It is basically a way to partition n data points into k clusters by solving the equation:

$$\arg\min_{\mathbf{c}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in c_i} d(\mathbf{x}, \mu_i)^2 = \arg\min_{\mathbf{c}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in c_i} \|\mathbf{x} - \mu_i\|_2^2$$

1. First place k center points randomly
2. For each data point x, find the nearest center point
3. After each point is assigned to a cluster, find the new centroid of that cluster by averaging its member data points
4. Repeat 2nd and 3rd steps until centroids stop changing.

# Color Based Segmentation Using K-means Clustering

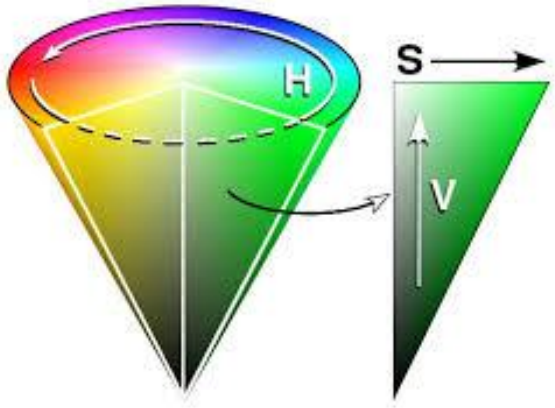
**A simple interactive example:**

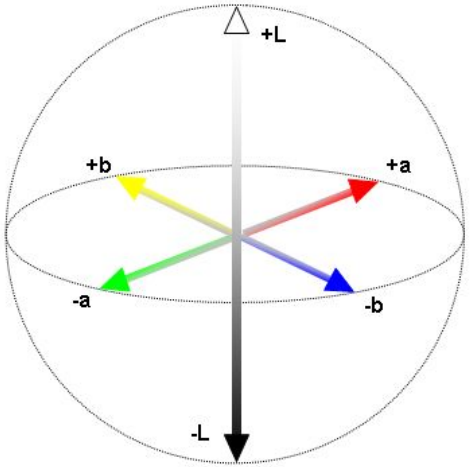
http://www.onmyphd.com/?p=k-means.clustering&ckattempt=2

# Color Based Segmentation Using K-means Clustering

In the image segmentation, k-means clustering is used such that 2 channels of the image is selected and treated as coordinates.

In our algorithm, both HSV and LAB color spaces have been tried since these two are the most widely used color spaces in k-means clustering in the literature.



# Color Based Segmentation Using K-means Clustering

Finding number k automatically:

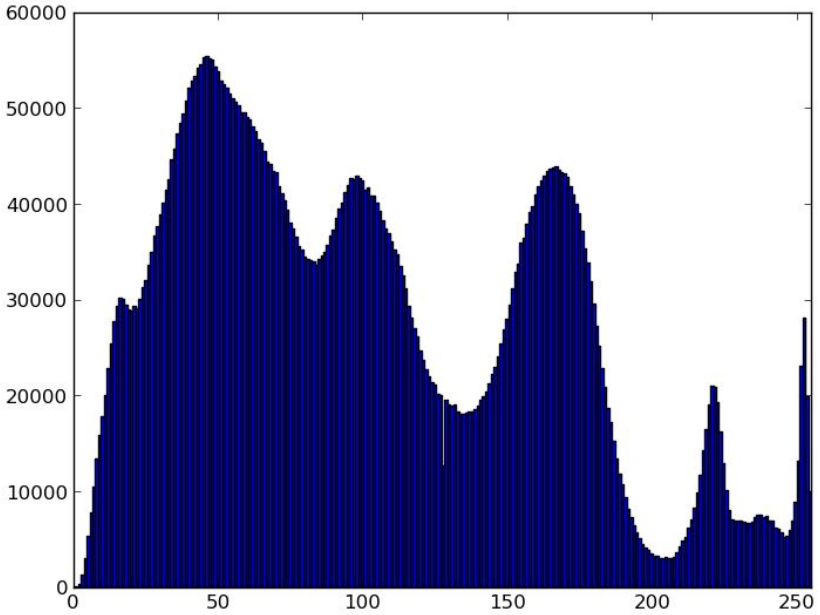
Find histogram, find peak points

in every channel, if a or b is smaller

than lightness, use lightness

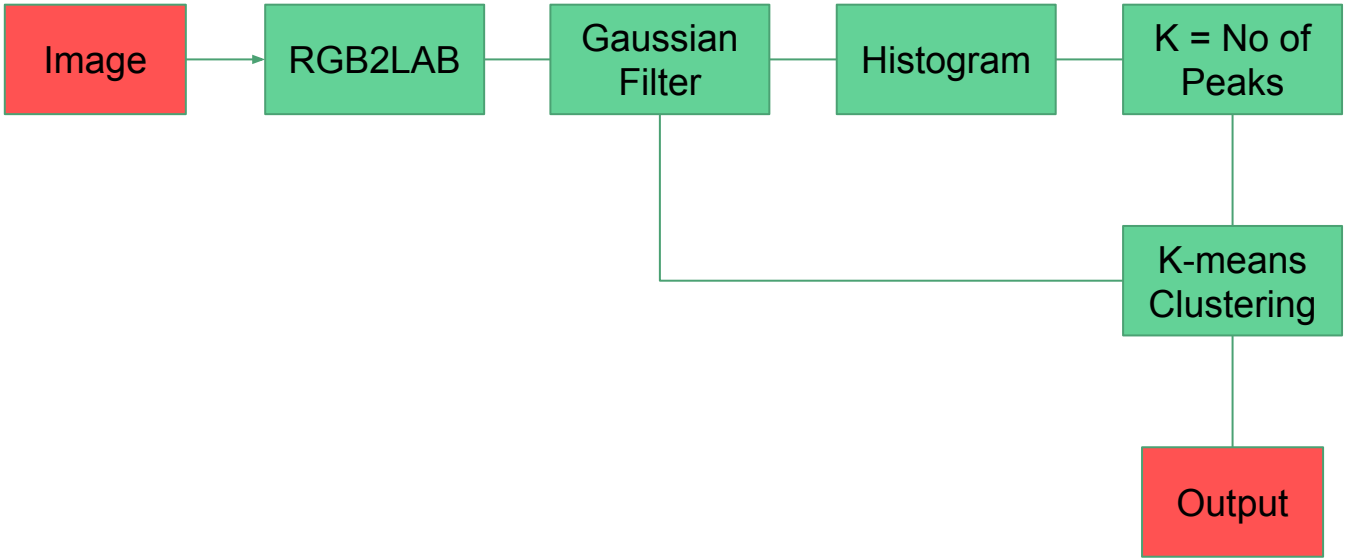
instead of a or b (such as l and b or

l and a)



# Color Based Segmentation Using K-means Clustering

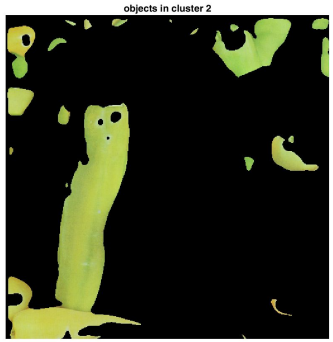
Overall algorithm:

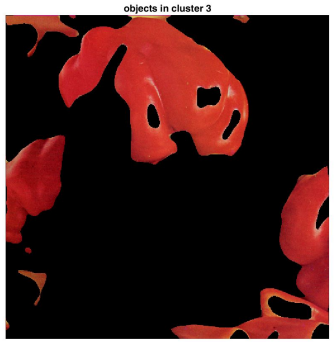


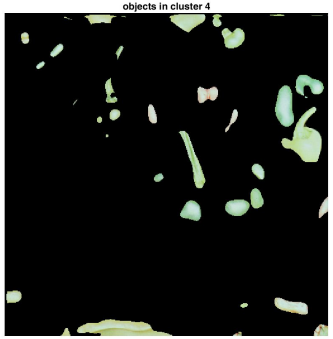
# Color Based Segmentation Using K-means Clustering

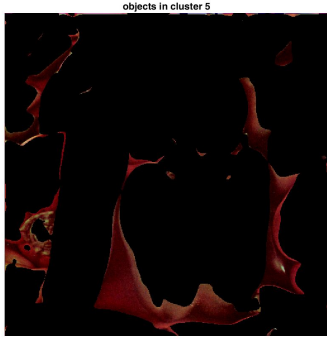
## Results of Some Sample Images:

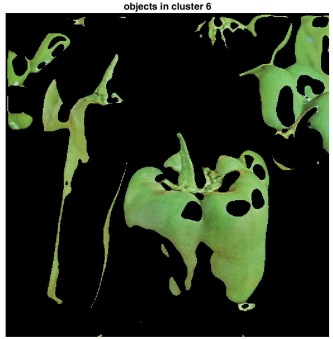




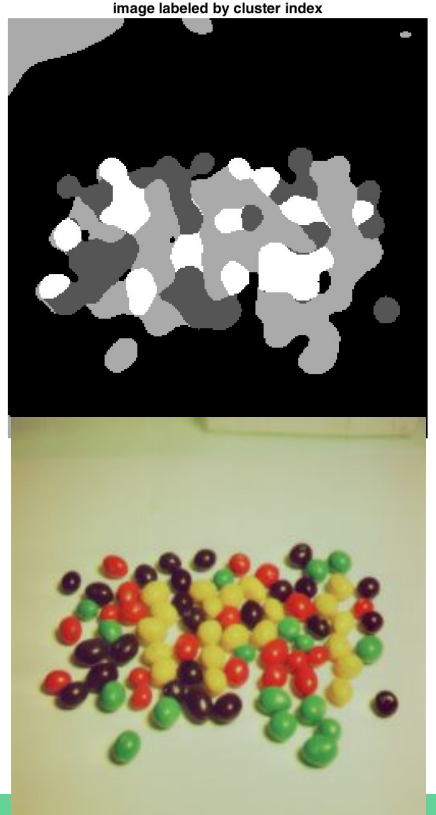

# Color Based Segmentation Using K-means Clustering






# Color Based Segmentation Using K-means Clustering





# Color Based Segmentation Using K-means Clustering

PROS:

- Generally very successful with multi-channel images

CONS:

- Can not segment objects that have same color with background or with other objects
- Computationally complex:

  O(iterations*clusters*dimensions*instances)

# Watershed Segmentation with Marker Control

## Theory:

- Definition
- Algorithm

## Practice:

- Over-segmentation
- Marker Control
- Flowchart
- Implementation Steps

## Results:

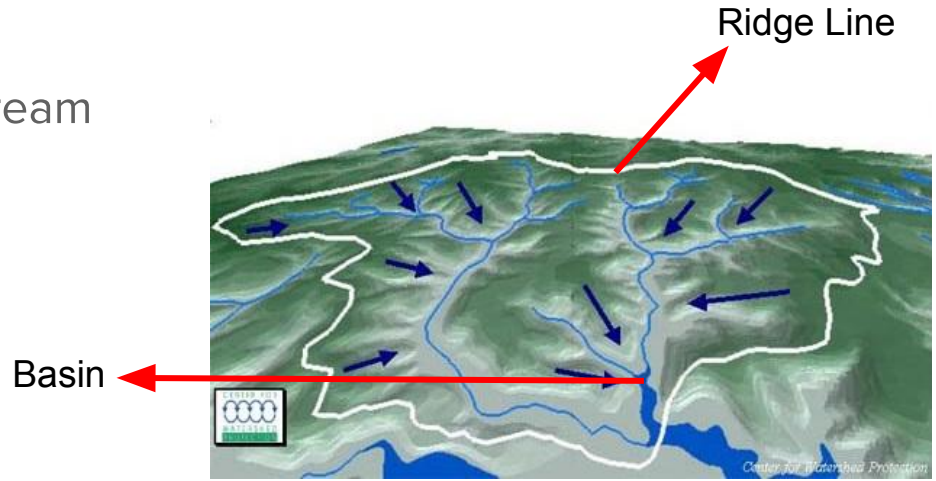
- All Resulting Examples
- Pros & Cons

# Watershed Segmentation with Marker Control

**What is a watershed ?**

Geographical area represents a stream to a low level point.

- Catchment Basins
- Ridge Lines




Picture taken from: http://www.wm.edu/

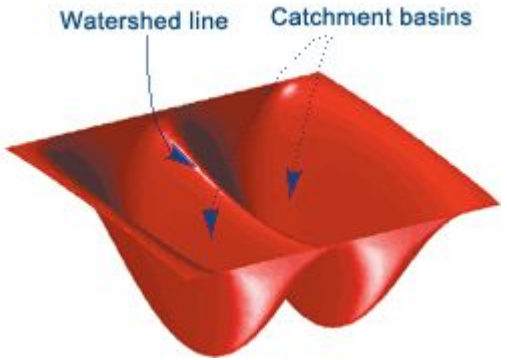
# Watershed Segmentation with Marker Control

**Image Processing Analogy**

Object features to be segmented corresponds to basins.

Separating ridge lines are object edges/perimeters.

Depends on gradient but the most importantly **LOCAL MINIMA**.




Picture taken from: http://www.mathworks.com/

# Watershed Segmentation with Marker Control

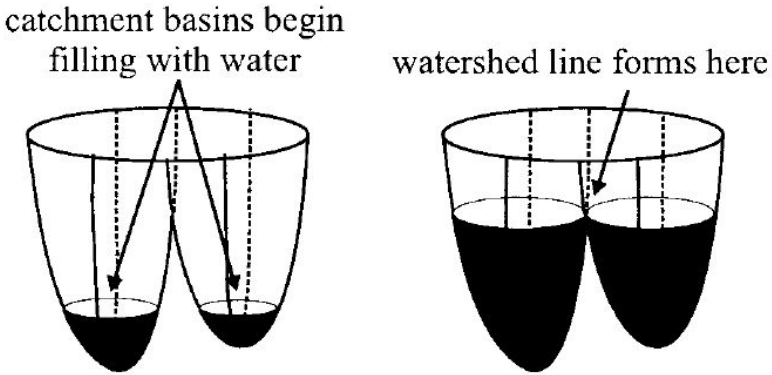
**Algorithm**

Suppose holes at each local minimum.

Flood water from below, incrementing by a uniform rate.

Water from distinct basins meet at ridge lines.

Continue until whole image is segmented.




Picture taken from: http://iacl.ece.jhu.edu/

# Watershed Segmentation with Marker Control

**F.Meyer's Flooding Algorithm**

Inter-pixel watershed concept (S. Beucher, F. Meyer, 1991)

1. Label local minima.
2. Non-labeled neighboring pixels to a priority queue.
3. Take out the pixel with least priority from the queue and check its neighborhood.
4. If all labeled pixels have the same label, give that label to current pixel. If there are more than one label then it is ridge line pixel.
5. Return step 2 if queue is not empty.

# Watershed Segmentation with Marker Control

## F.Meyer's Flooding Algorithm


Label each minimum.


Choose low priority pixel from queue.


Check neighborhood and label.


Whole segmented image.

# Watershed Segmentation with Marker Control

## Over-segmentation

Directly using ‘watershed(Grayscaled Image)’.

# Watershed Segmentation with Marker Control

**Pre/Post-processing**

- Noise removal
- Morphological operations (gradient, contrast, area filtering etc.)
- Low-pass or Gaussian filtering
- High-pass filtering, Edge-detection
- Local operations (mean, variance etc.)
- Distance Transform, Fourier Transform etc.

- Region Merging

# Watershed Segmentation with Marker Control

**Marker Control as a Pre-processing Method**

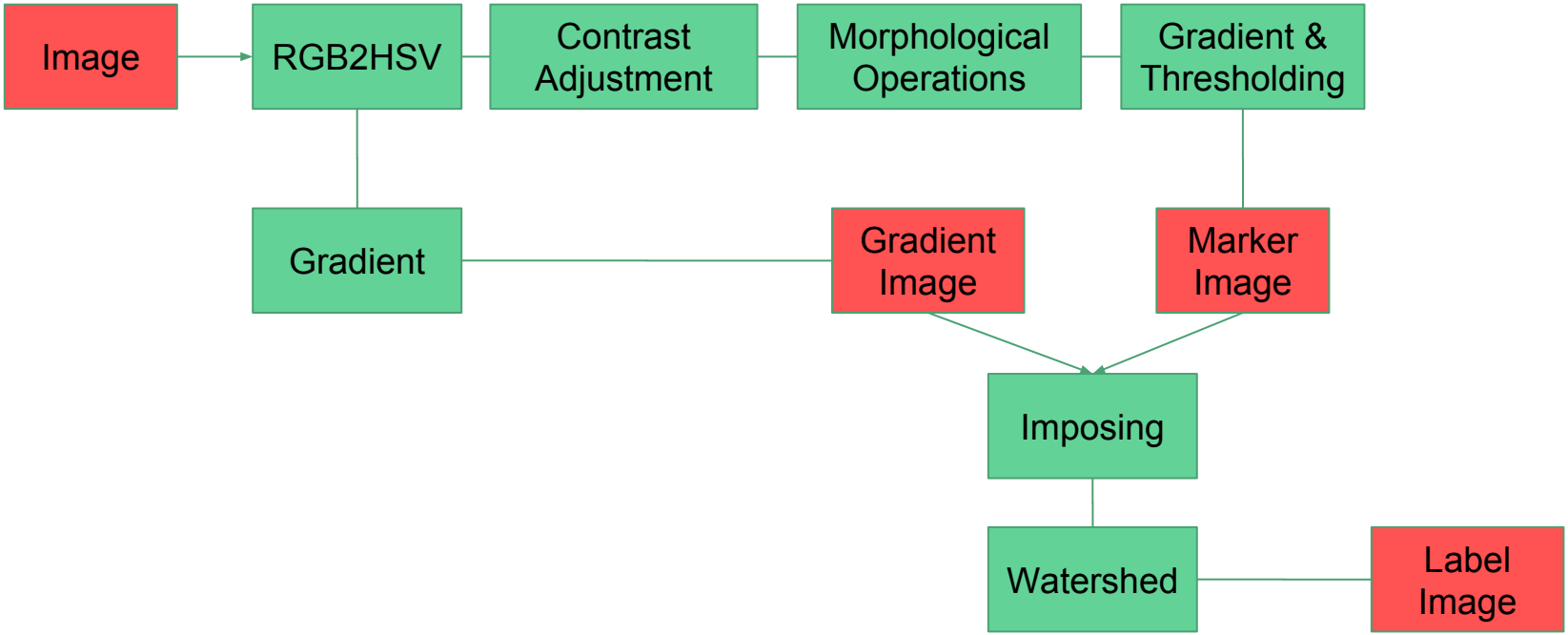
Either before or after watershed, choosing processing methods is very image properties dependant and purposeful.

Choosing which area has local minima preserved. Beside markers no other minima remains.

Markers represent object feature to be segmented.

# Watershed Segmentation with Marker Control

## Flowchart of the Practical Implementation



# Watershed Segmentation with Marker Control

**Some Important Functions in MATLAB**

**imtophat**: Stands for Original Image-Opened Image (used for background elimination)

**imbothat**: Stands for Closed Image-Original Image (enhance valleys, low levels)

**imimposemin**: Fundamental for Marker Control. Imposes minima only in marked area and eliminating all other minima in the image.

**label2rgb**: Gives out an RGB image with coloring labels with specified color chart.

# Watershed Segmentation with Marker Control

## Implementation Steps

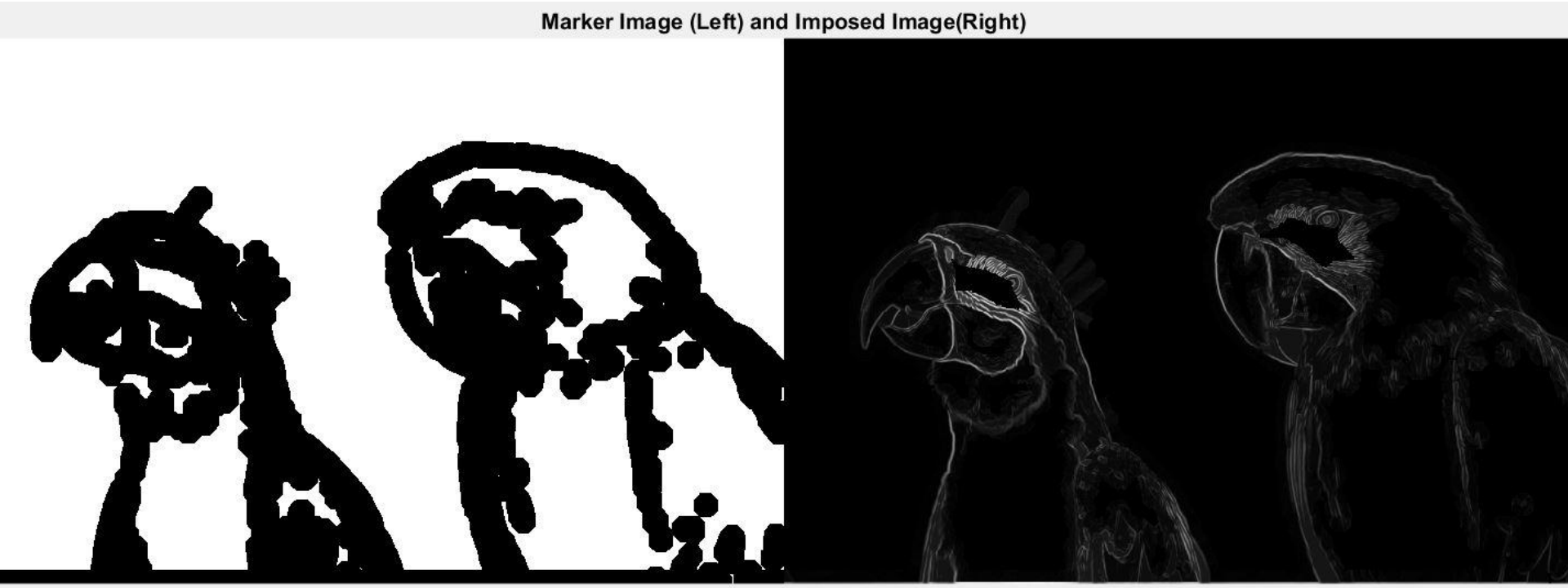
**Contrast Adjusted Image (Left) and Morphological Modified Image(Right)**

Original Image + Top-Filtered Image - Bottom-Filtered Image

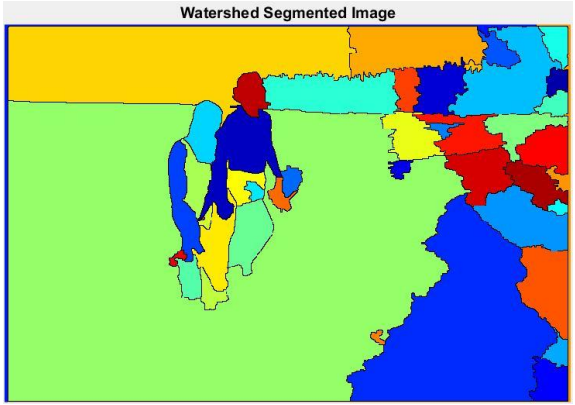
Top-Filtered Image + Bottom-Filtered Image

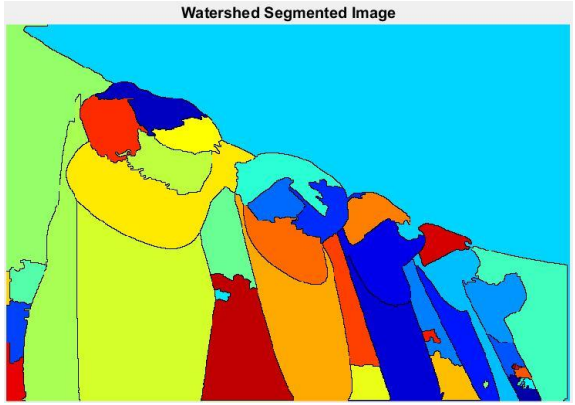
# Watershed Segmentation with Marker Control

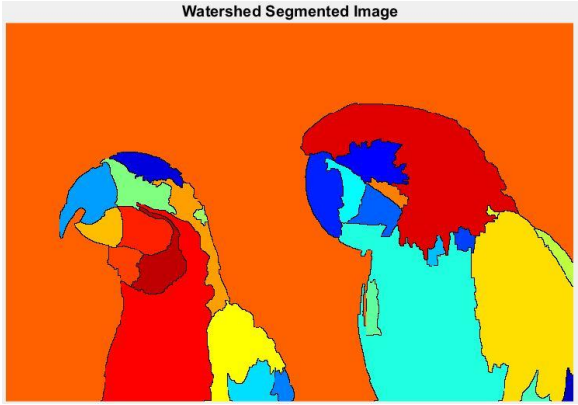
## Implementation Steps

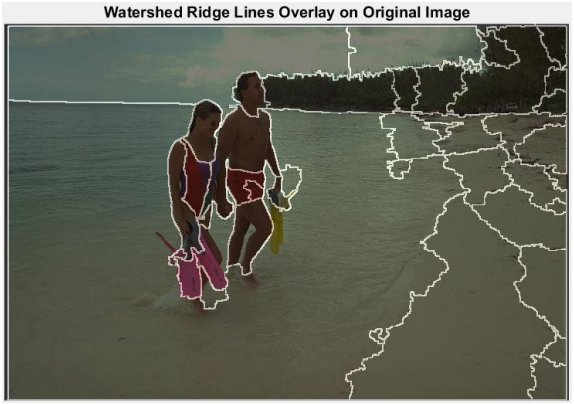


# Watershed Segmentation with Marker Control



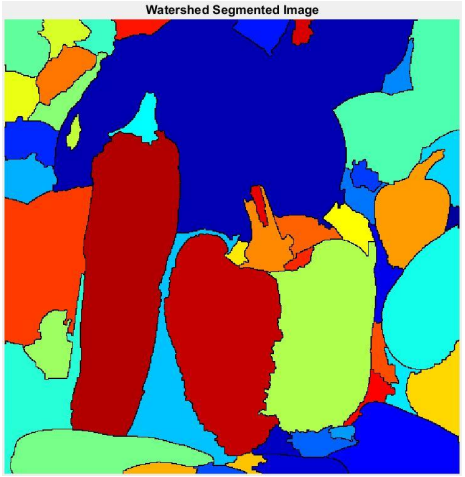



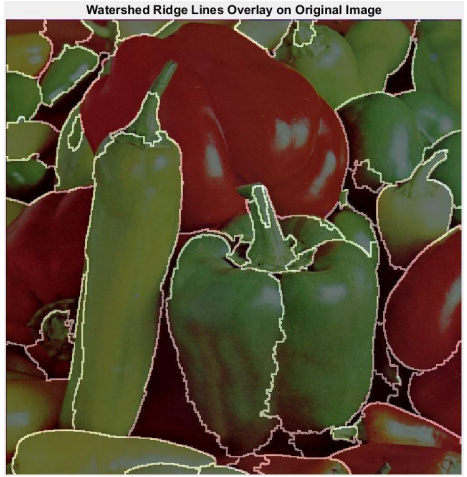



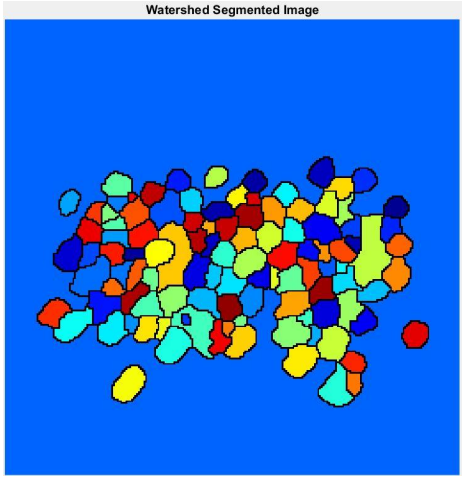



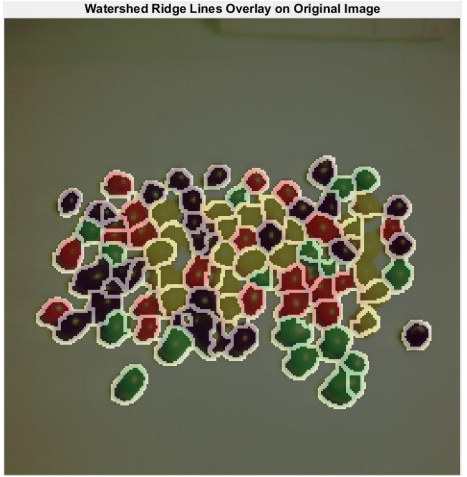


# Watershed Segmentation with Marker Control



# Watershed Segmentation with Marker Control

PROS:

- Fast
- Can separate objects with similar colors
- Can separate overlapped objects
- Can identify desired object features defined by user
- Can be combined almost every segmentation method
- Works for gray-scale images

CONS:

- Pre- or post-processing is image-dependant
- There are many methods to take into account

# Conclusion

Choosing the method depends on the application, image features to be detected, scene illumination and recording media specifications.

- Faster performance: Edge Detection
- Detecting detailed features separately: Watershed
- Smoother separation in terms of colors: K-means Clustering

... may be a better choice.

# Questions ?