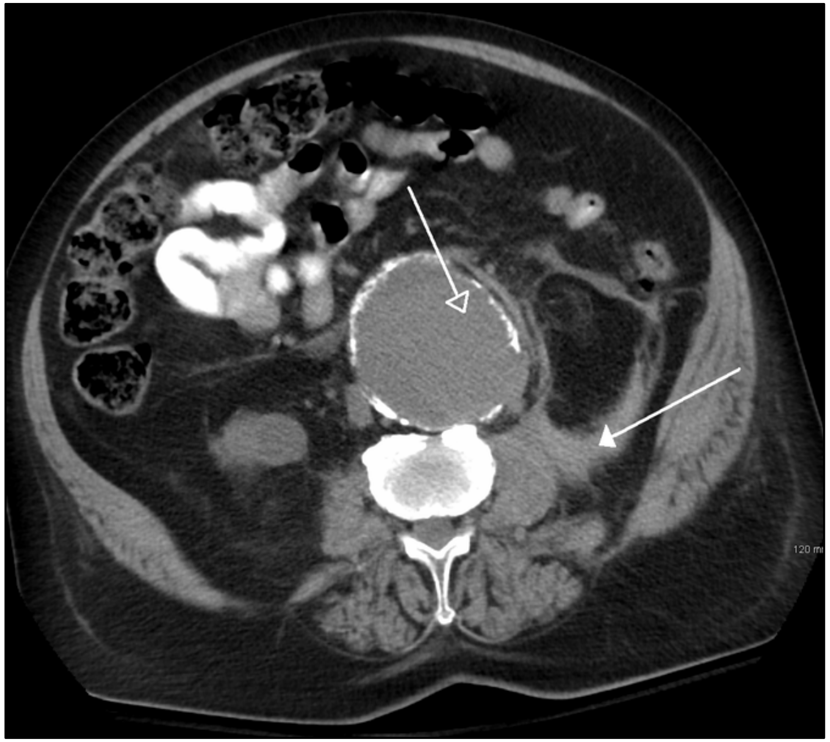
NON-PARAMETRIC BAYESIAN METHOD FOR IMAGE REGION SEGMENTATION

David Esparza Alba
Dr. Tadahiro Taniguchi

Image Segmentation

- Image segmentation refers to the act of grouping pixels that share certain characteristics like color, intensity, texture, etc.
- The main goal is to be able to identify different objects in an image.
- Each region represents an object or a part of an object.

Applications



Medical Image, tumor detection, abnormalities, heart disease etc.

Face Recognition. Digital Cameras, Facebook, etc.

Object tracking. Security cameras, satellites, etc.

Proposed Method

- The proposed method consists in three steps:
  - **Clustering step:** Associate pixels to a specific cluster based on their color.
  - **Split step:** Separate group of pixels which are connected and associated to the same cluster.
  - **Merge step:** Eliminate those objects composed of a number of pixels bellow of a threshold value and associate those pixels to the nearest object.

Clustering Step

Ritsumeikan University

David Esparza Alba

Clustering Step

- Associate each pixel to a specific cluster.
- Each cluster represents a Gaussian distribution with parameters μ_k and Δ_k.
- Also, each Gaussian as a mixing proportion parameter π_k associated to it.

Clustering Step

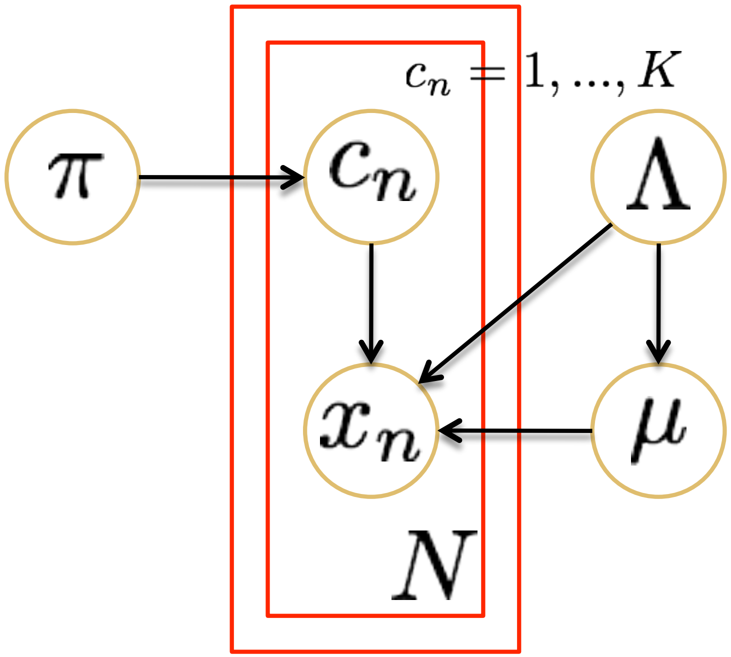
- The problem consists in estimate the values of the parameters for each Gaussian.

- One way to estimate the parameters in each iteration is using Gibbs sampling.

- It is required the conditional distribution of each parameter given the values of the other parameters.

Clustering Step

- The dependencies between parameters are illustrated with the following graphical model (Finite model)



$$P(X, C, \pi, \mu, \Lambda) = p(X|C, \mu, \Lambda)p(\mu|\Lambda)p(\Lambda)p(C|\pi)p(\pi)$$

Clustering Step

- According to our graphical model, the posterior distributions are given by

$$q(C) \sim p(X|C,\mu,\Sigma)p(C|\pi)$$

$$q(\pi) \sim p(C|\pi)p(\pi)$$

$$q(\mu) \sim p(X|C,\mu,\Lambda)p(\mu|\Lambda)$$

$$q(\Lambda) \sim p(X|C,\mu,\Lambda)p(\Lambda)$$

Clustering Step

- According to our graphical model, the posterior distributions are given by

$$q(C) \sim p(X|C, \mu, \Sigma)p(C|\pi)$$ Multinomial

$$q(\pi) \sim p(C|\pi)p(\pi)$$ Dirichlet

$$q(\mu) \sim p(X|C, \mu, \Lambda)p(\mu|\Lambda)$$ Normal

$$q(\Lambda) \sim p(X|C, \mu, \Lambda)p(\Lambda)$$ Wishart

Clustering Step

- If we take $K \to \infty$ the prior distribution of the latent variables is defined by

$$p(c_{ij} = 1) = \begin{cases} \frac{N_{-i,j}}{N-1+\alpha} & \text{components where } N_{-i,j} > 0 \\ \frac{\alpha}{N-1+\alpha} & \text{all other components combined} \end{cases}$$

- The probability of assign certain observation to an already existing cluster is proportional to the number of observations associated to that cluster.

- The probability of assign certain observation to a new cluster is proportional to the concentration parameter.

- Observations tends to join to the "most popular" cluster.

Clustering Step: Illustration

- To illustrate the clustering step, we used a set of 300 observations generating three identifiable clusters.

- The hyperparameters used for this example are listed bellow

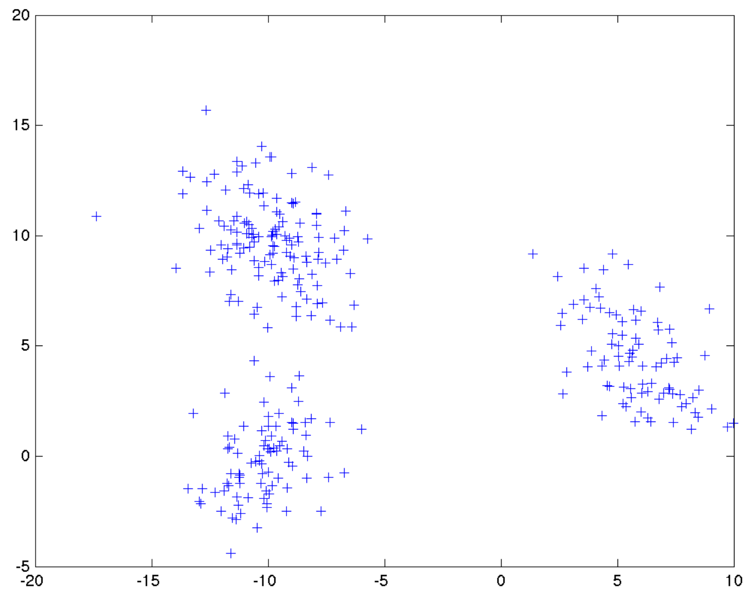
$$\alpha = 3.5$$

$$W_0 = 0.01 * d * I$$

$$v_0 = d$$

$$m_0 = \overline{x}$$

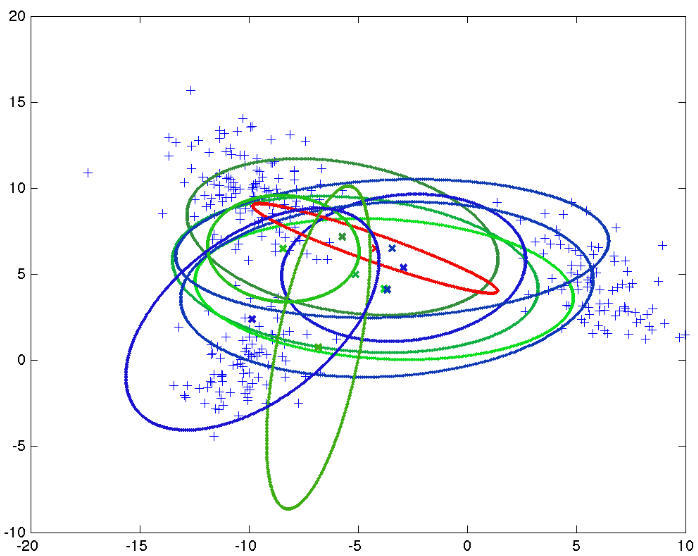
$$\beta_0 = 1$$



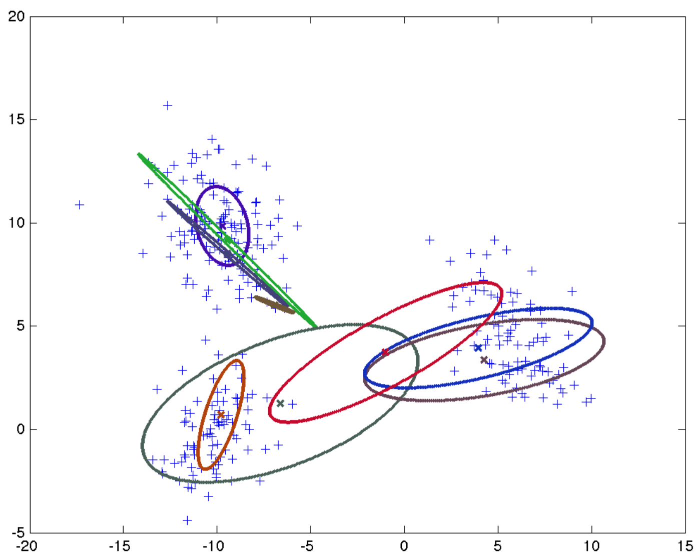
Data set with 300 observations

Clustering Step: Illustration

After 1 iteration	After 5 iterations



9 clusters

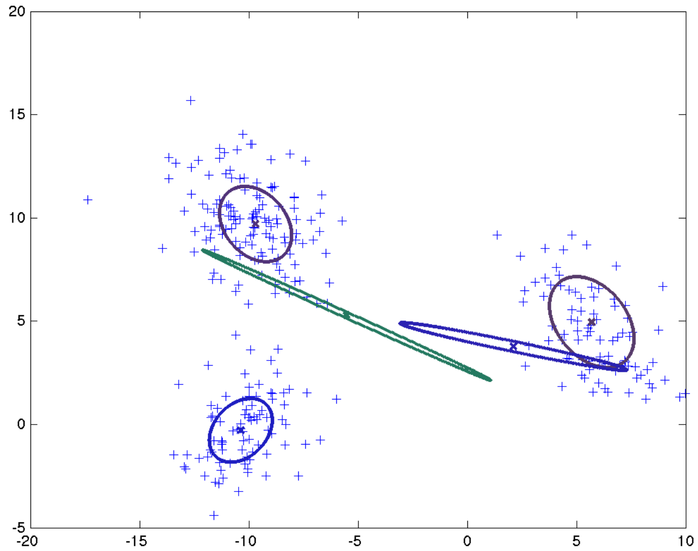


9 clusters

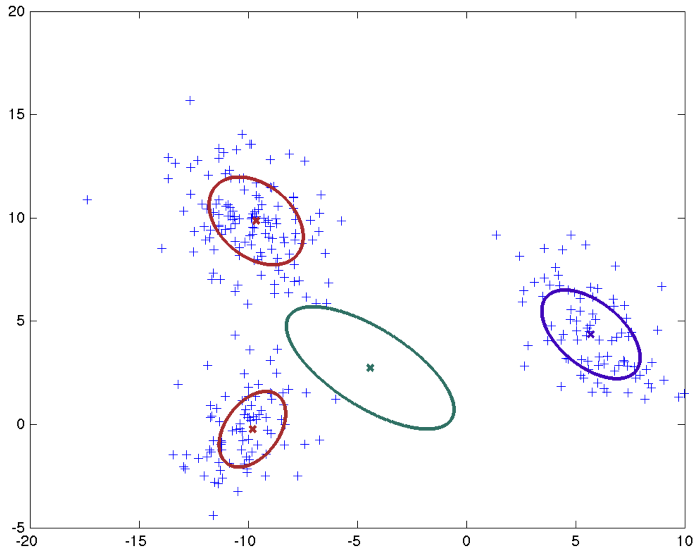
Clustering Step: Illustration

After 25 iterations

After 50 iterations



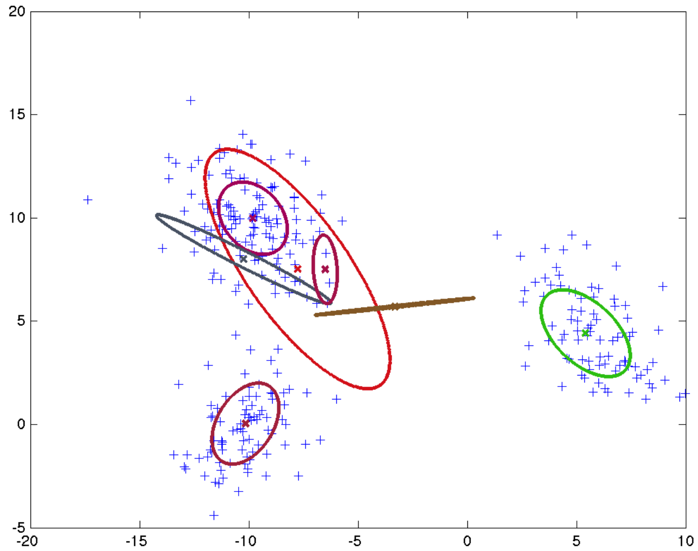
6 clusters



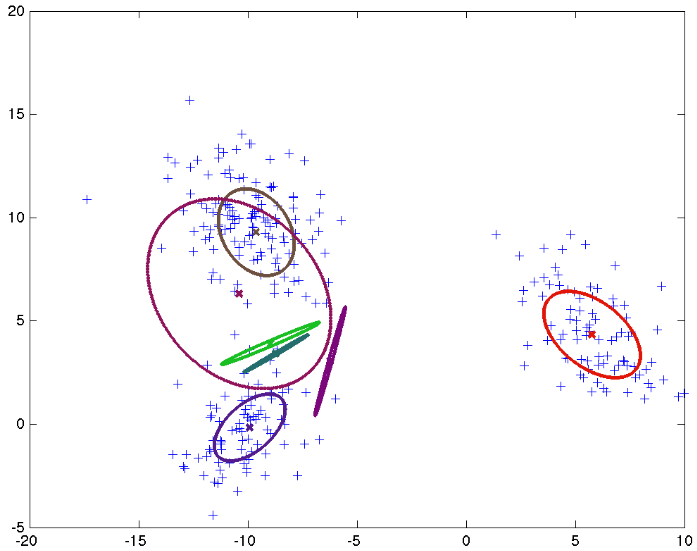
5 clusters

Clustering Step: Illustration

After 100 iterations	After 150 iterations



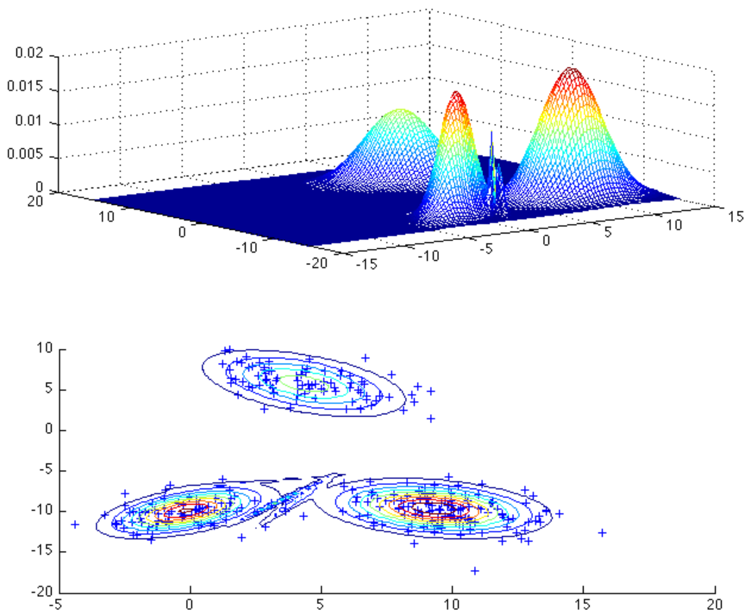
6 clusters



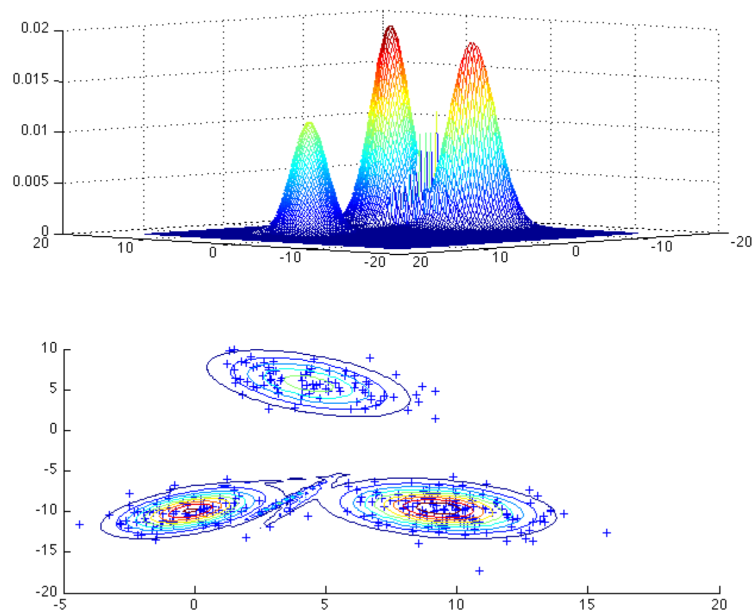
6 clusters

Clustering Step: Illustration

PDF after 150 observations



PDF after 150 observations



Probability density function using a non-parametric Bayesian mixture model wit concentration parameter $\alpha = 3.5$ for a set of 300 observations.

Clustering Step

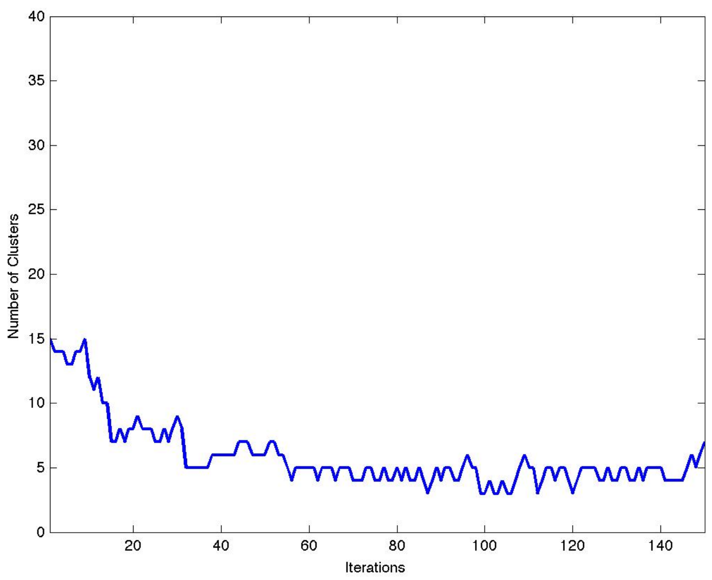
- What happens if we change the value of the concentration parameter?

  - If we increase the concentration parameter it is expected that the number of observable components will be bigger, but at the end the number of components will oscillate around the expected value.

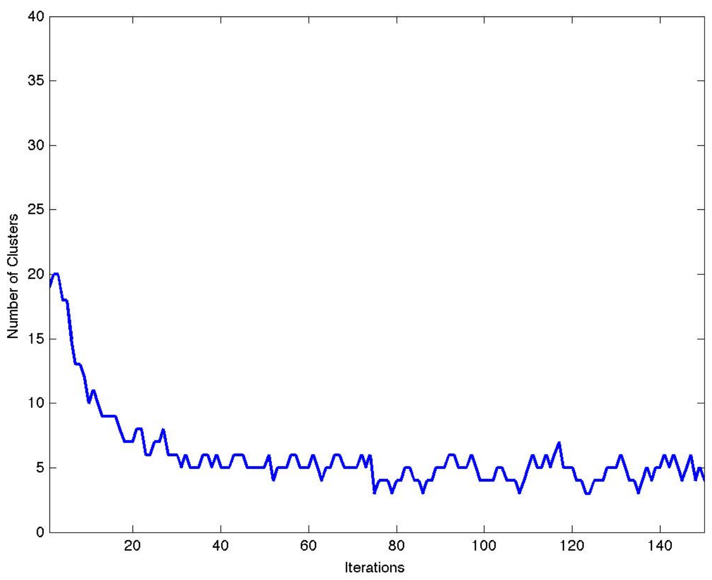
  - On the other hand, if we decrease the concentration parameter it is will be more difficult to create new components.

Clustering Step

With α = 3.5 and 150 iterations

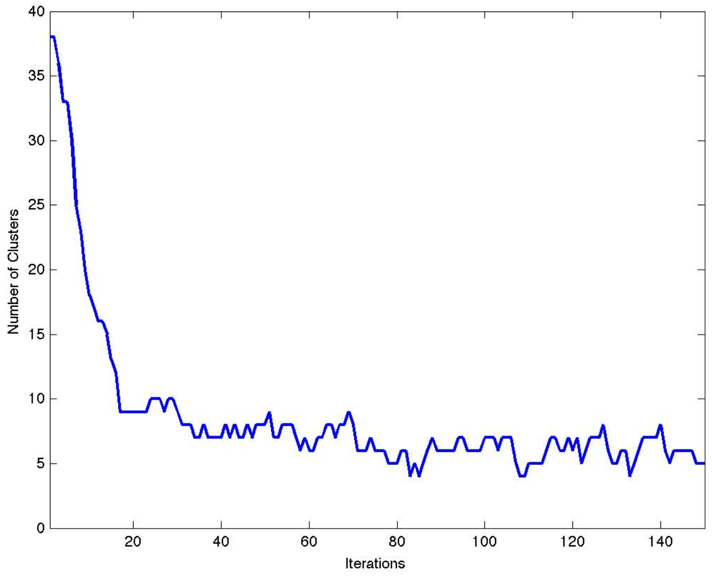


With α = 5 and 150 iterations

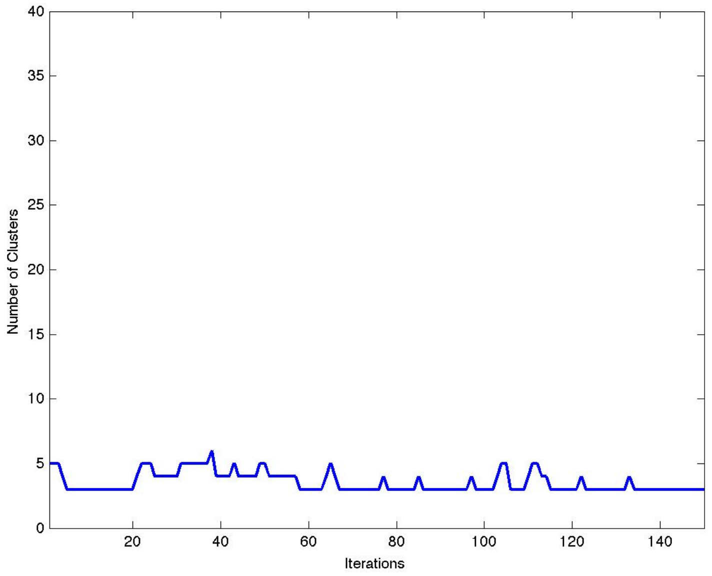


Clustering Step

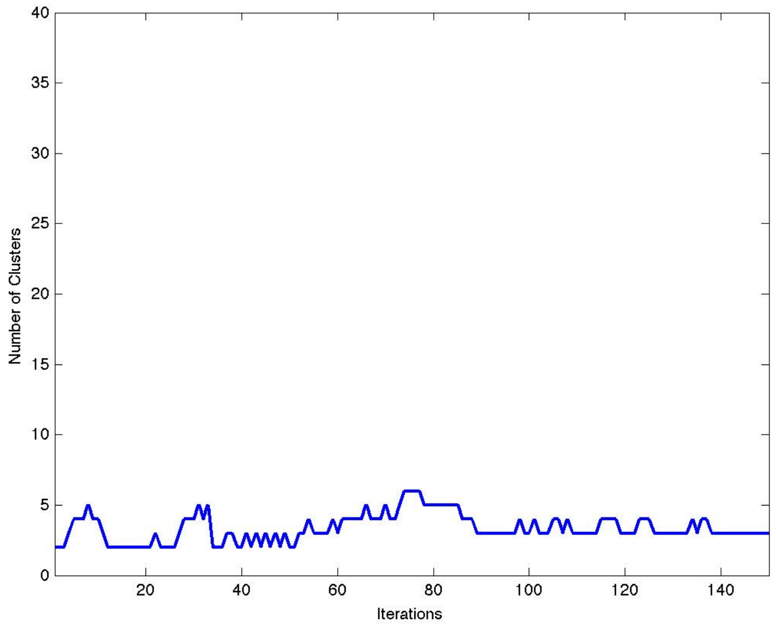
With α = 10 and 150 iterations



With α = 0.5 and 150 iterations



Clustering Step



- There are a total of three identifiable clusters.

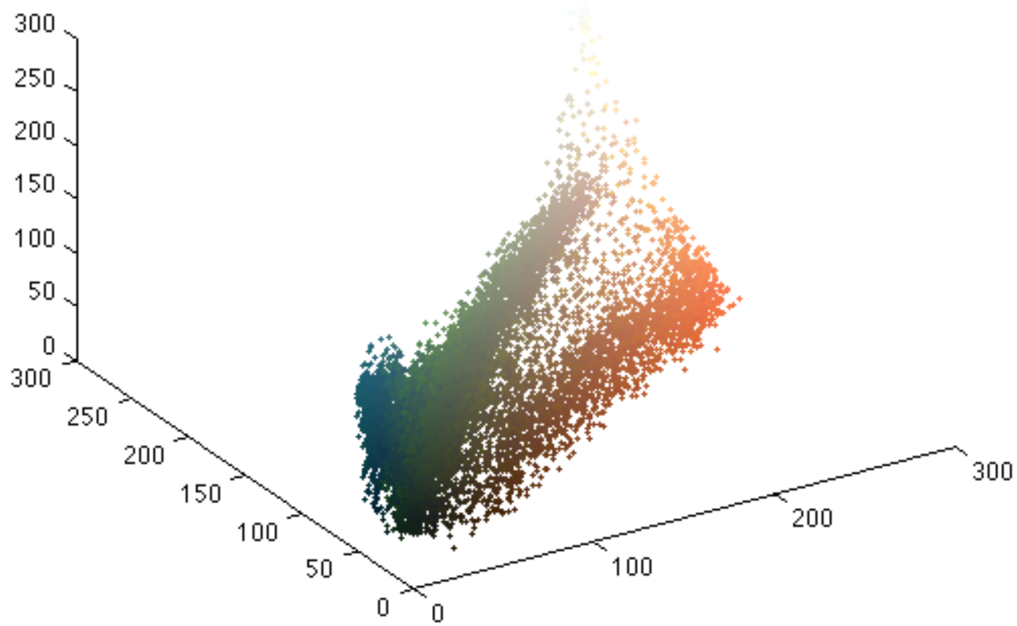
- Using $\alpha = 0.5$ The initial number of clusters is bellow three, but after around 60 iterations, the number of cluster established around three.

Clustering Step

- Using RGB values as points in a 3-dimensional space.
- Associate each point to the cluster which is more probably to belong to.
- There will be one cluster for red pixels, another for purple, another for yellow, etc.
- Try to associate pixels with similar color to the same cluster.

Clustering Step

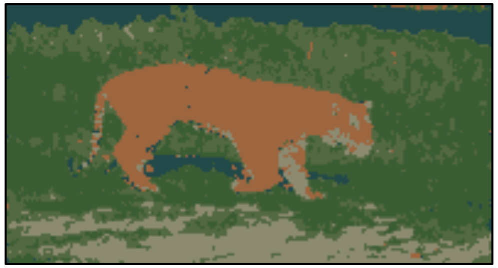
- Consider the following image



Representation of an image in the 3-dimensional space. In the left the original image showing a tiger in a nature environment. At right the representation of the image in a 3-dimesional space, where each dimension corresponds to a color level.

Clustering Step

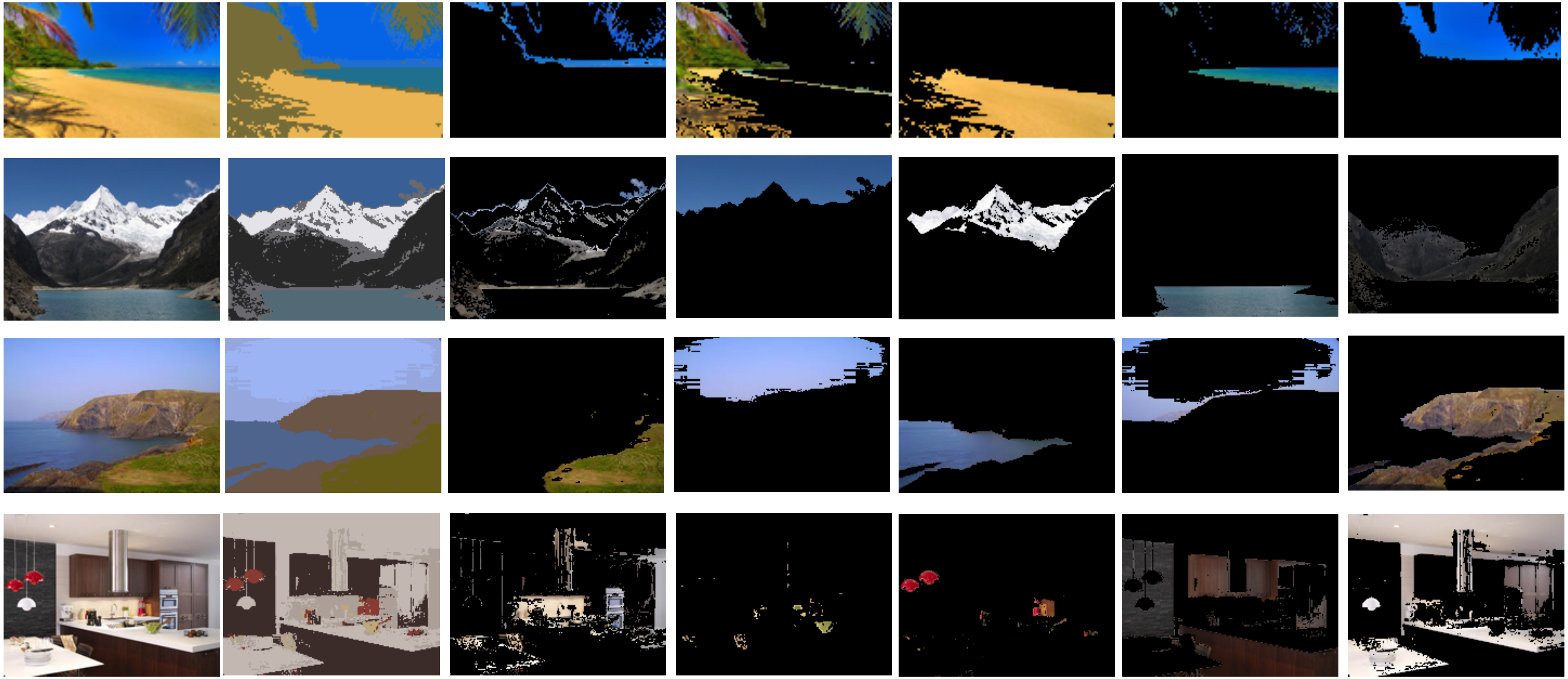
- After applying the clustering algorithm using 5 Gaussians we obtain



- Now we can identify easily the pixels that are associated to each cluster

Clustering Step

Following we show some examples of clustering using 5 Gaussian functions.



Clustering step using 5 Gaussian functions. At left the original image, then the segmented image showing the 5 clusters, finally 5 images displaying the pixels from each one of the clusters.

Split Step

Ritsumeikan University

David Esparza Alba

Split Step

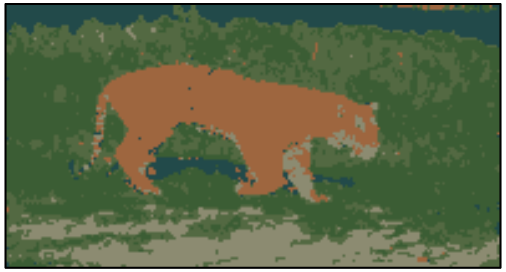
- Isolate group of pixels that are connected and belong to the same cluster.

- The flood fill algorithm compare the current pixel with each one of its 8 neighbors.

- If the neighbor pixel and the current pixel are associated to the same cluster, then they are part of the same object.

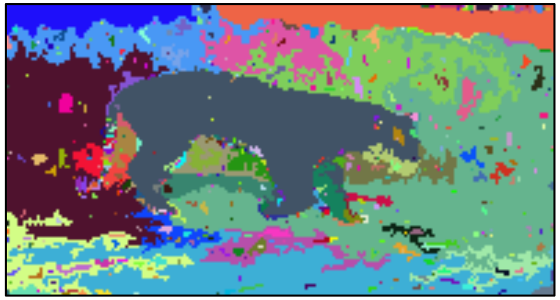
Split Step

- Using the following image



Original image at left. Resulting image after clustering step at right.

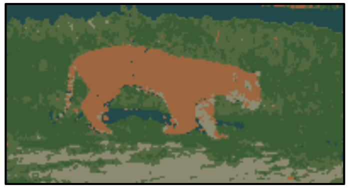
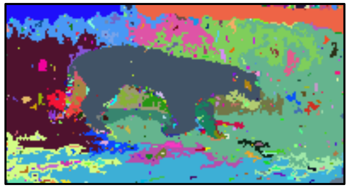
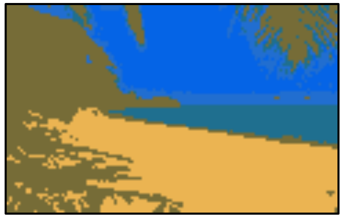
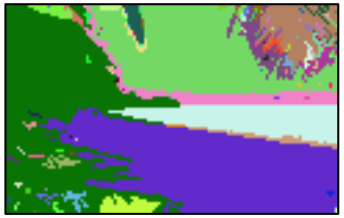
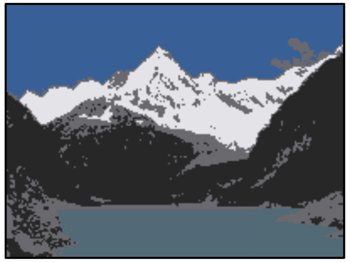
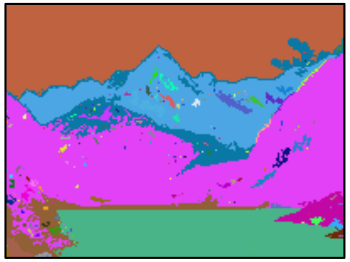
- After the split step, we get the image is compounded by 507 isolated areas or objects.



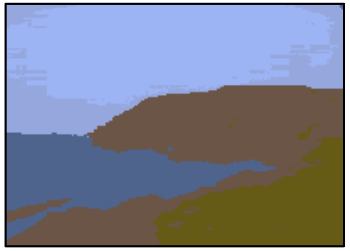
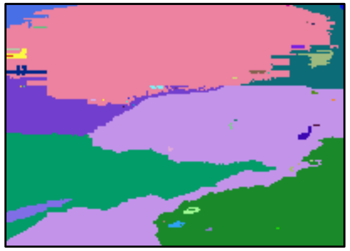
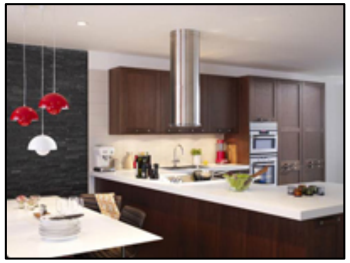
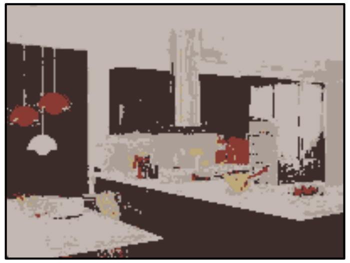
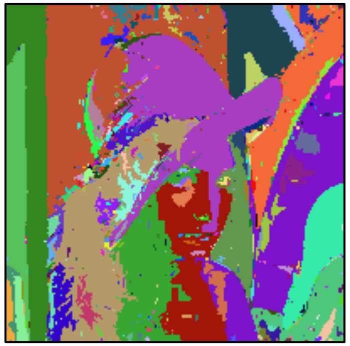
Resulting image after split step

Split Step

- Applying the split step in the same test images used before, we obtain

Original Image	Clustering Step	Split Step	Objects
			507
			128
			197

Split Step

Original Image	Clustering Step	Split Step	Objects
			48
			470
			718

Merge Step

Ritsumeikan University

David Esparza Alba

Merge Step

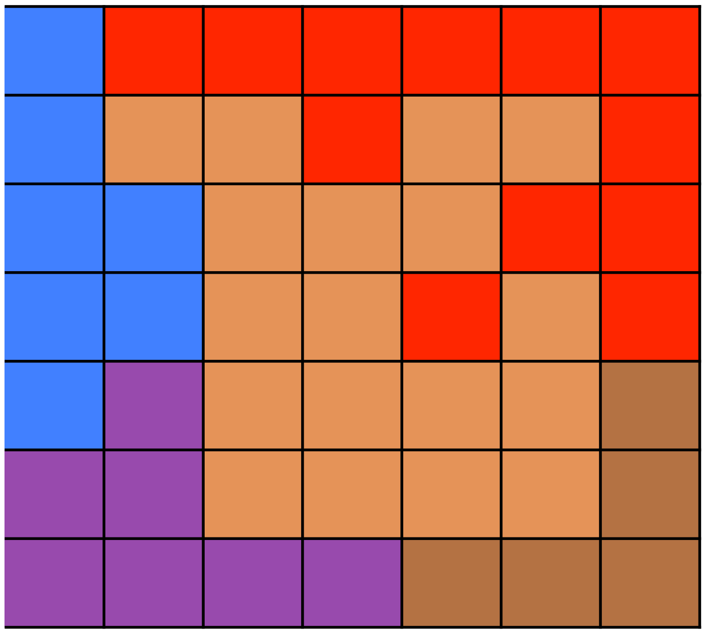
- Eliminate small objects.

- Those objects with a number of pixels bellow of certain threshold value are eliminated.

- The pixels of the removed objects are associated to the most similar neighbor not removed object.

Merge Step

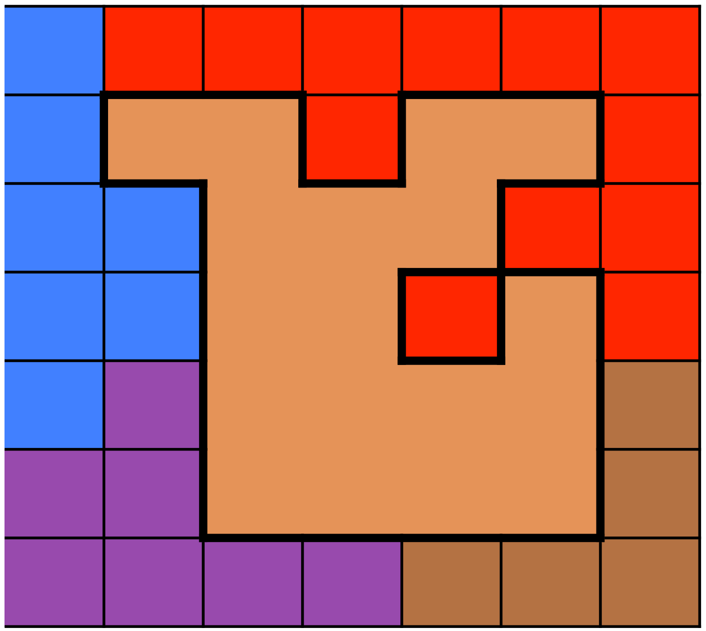
Consider the following portion of an image



Merge Step

Consider the following portion of an image

Identify the object to be removed

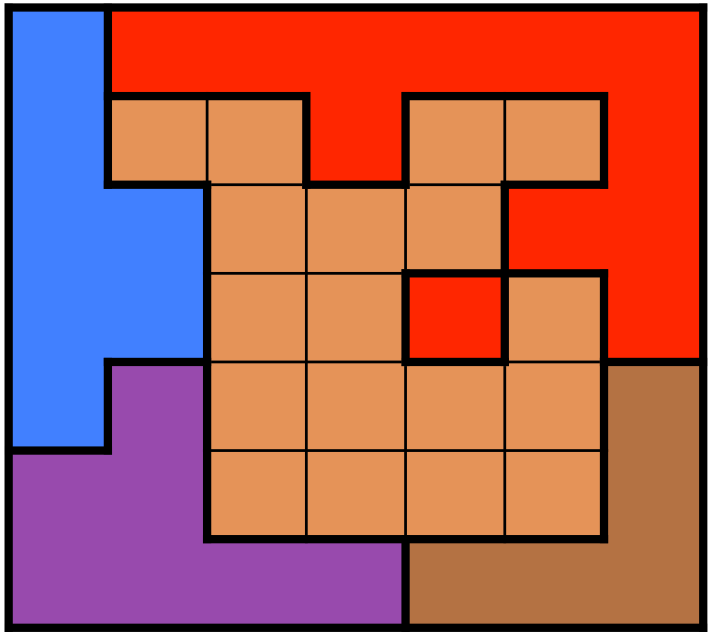


Merge Step

Consider the following portion of an image

Identify the object to be removed

Identify its neighbors

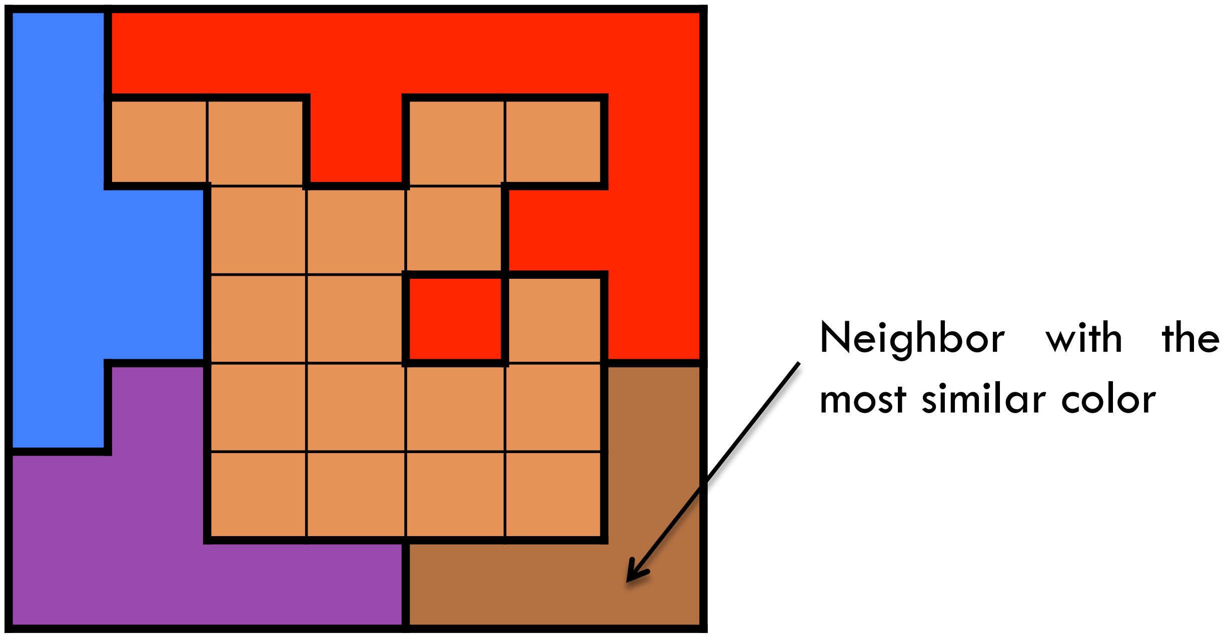


Merge Step

Consider the following portion of an image

Identify the object to be removed

Identify its neighbors



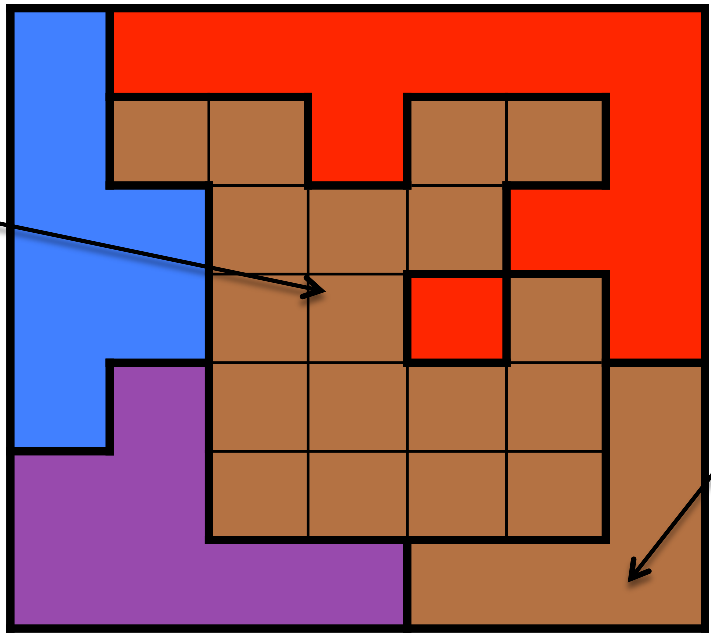
Merge Step

Consider the following portion of an image

Identify the object to be removed

Identify its neighbors

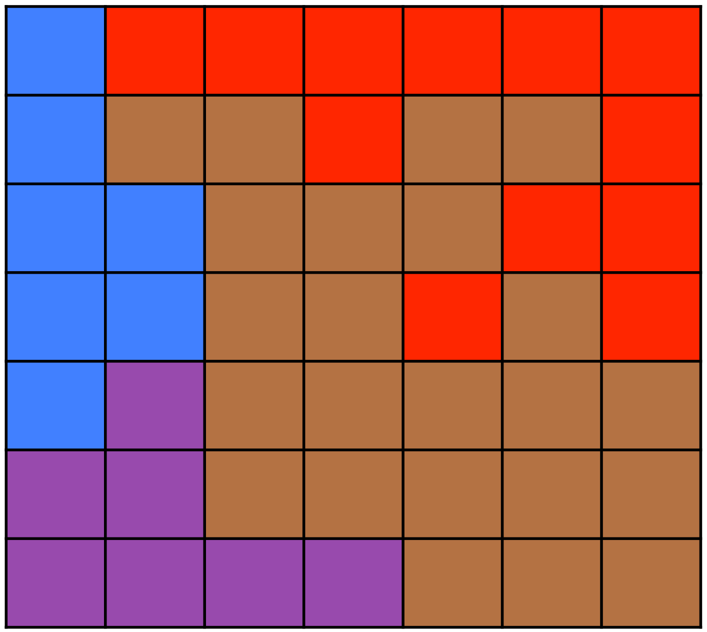
Pixels are associated to the object with the most similar color



Neighbor with the most similar color

Merge Step

Resulting image!!



Merge Step

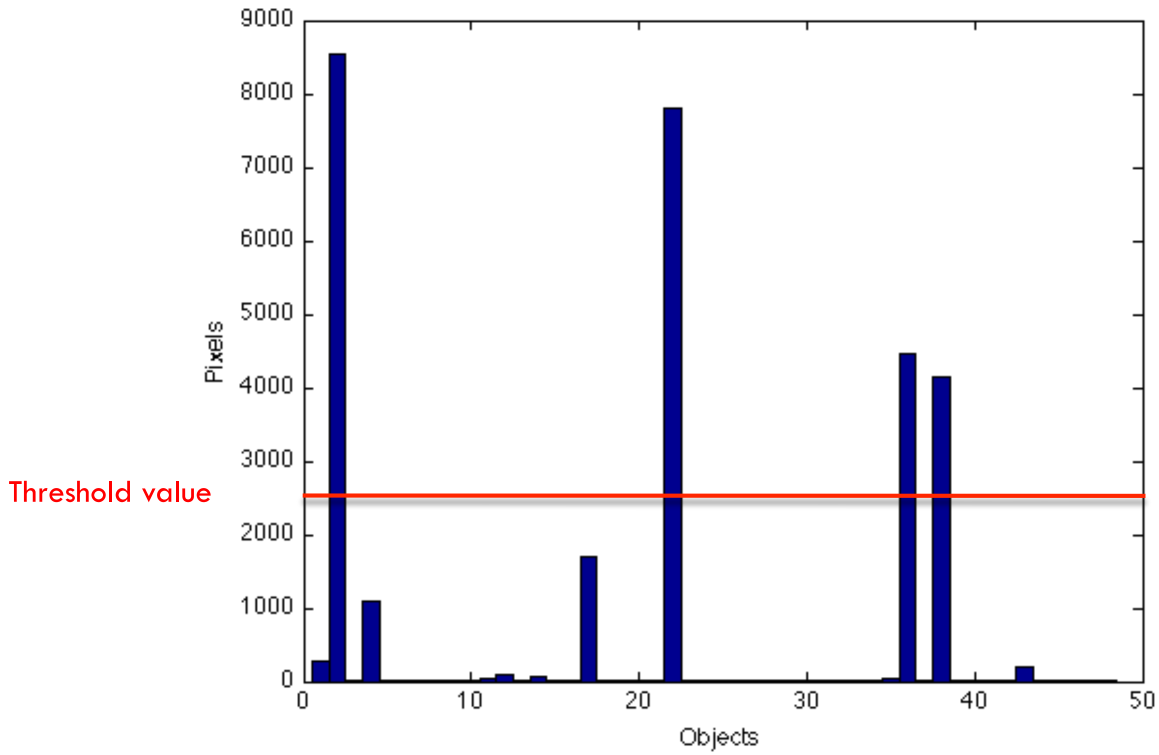
Consider the following image

In the left the original image, in the right the resulting image after the split step.

After the split step we have a total of 48 objects in the image, some of them bigger than others. Following is a histogram showing the number of pixels in each object.

Merge Step

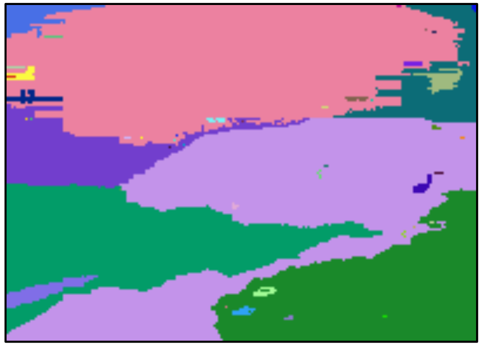
We can see the threshold value as a line, an those bins bellow that line are removed and their pixels pass to be part of bigger objects.



Objects with number of pixels bellow the threshold values are removed ant their pixels

Merge Step

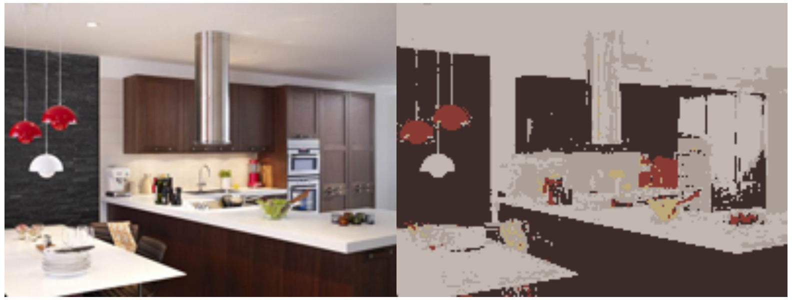
After applying the merge step with a threshold value TH = 100, we obtain

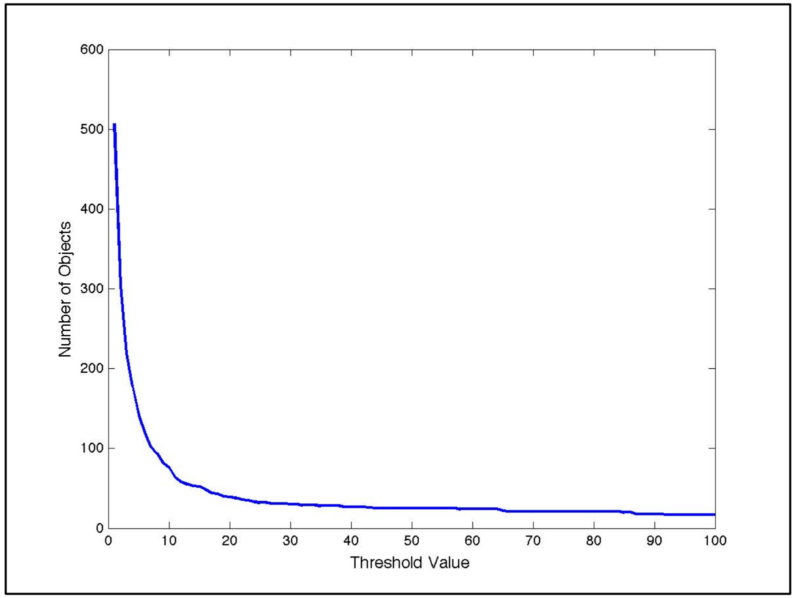
Original Image	**Split Step**	**Merge Step**	**Objects**
			8

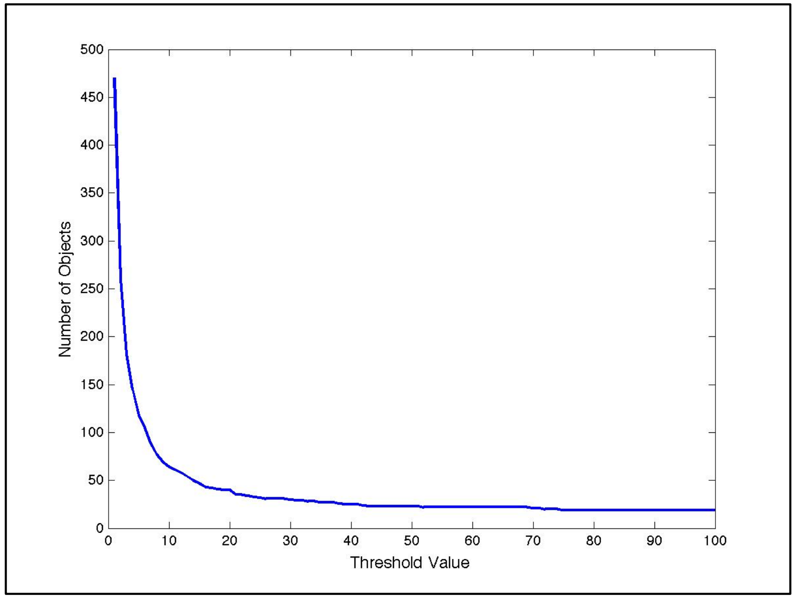
The image at the left is the original image, the second image is the result after applying the split step, the third image is the resulting image after the merge step, the last column indicate the number of objects founded after the merge step.

The number of object decrease depending on the threshold value, resulting in bigger and more easily identifiable regions

Merge Step

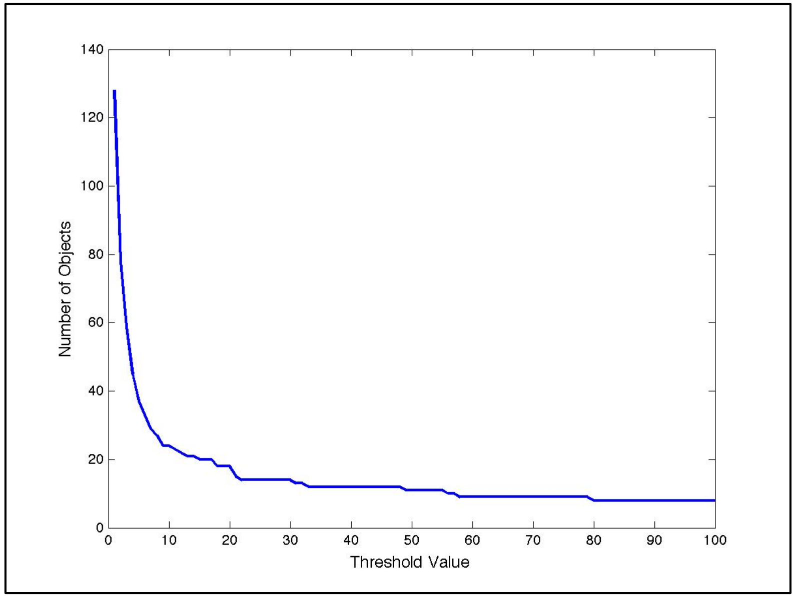


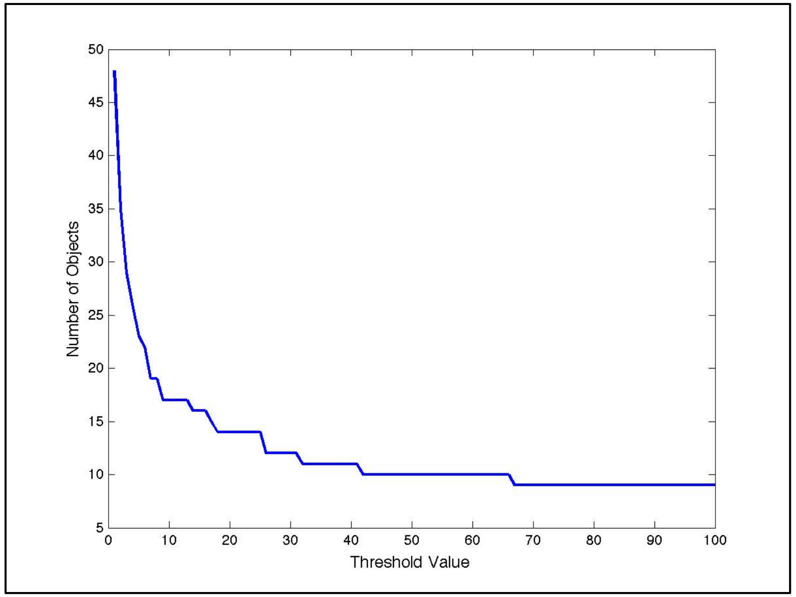




The image at the right corner represents the original image, in the left corner is the resulting image after the clustering step. The graph shows a relation between the threshold value and the number of objects after the merge step.

Merge Step





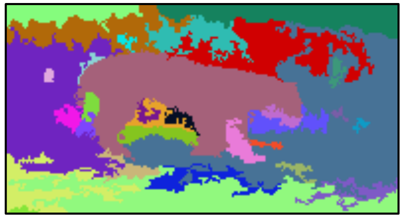
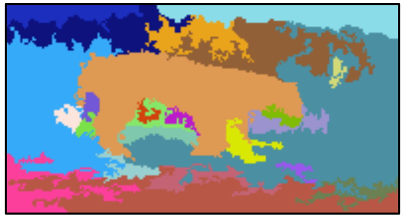
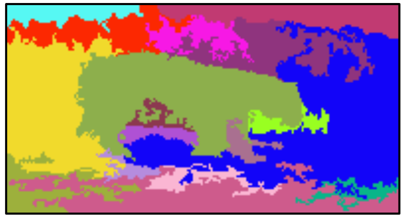
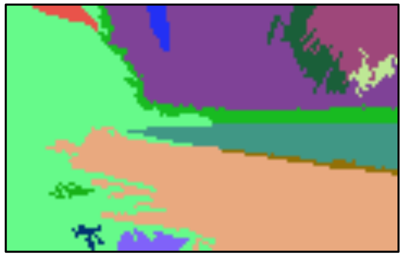
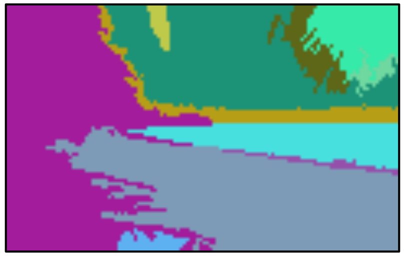
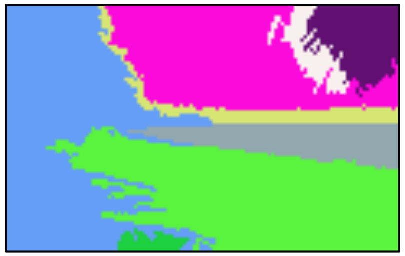
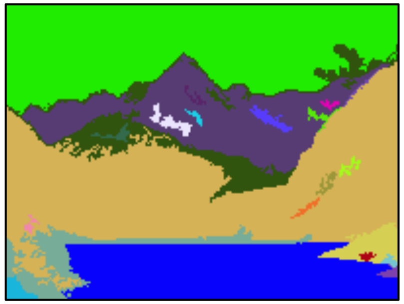
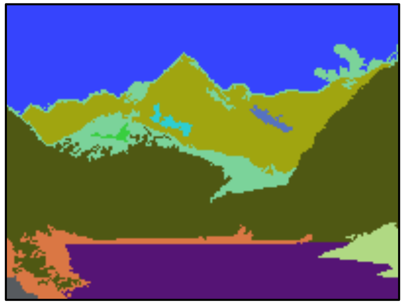
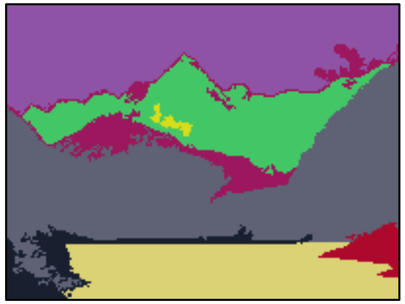
The image at the right corner represents the original image, in the left corner is the resulting image after the clustering step. The graph shows a relation between the threshold value and the number of objects after the merge step.

Results

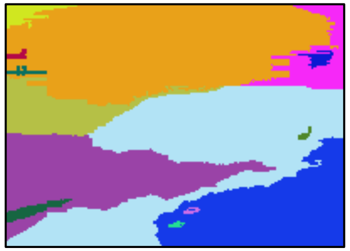
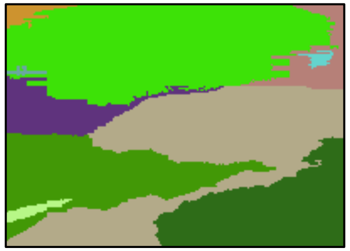
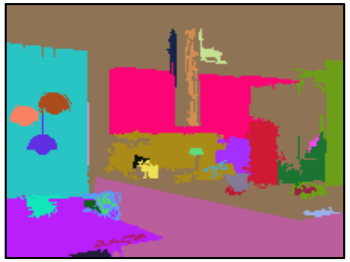
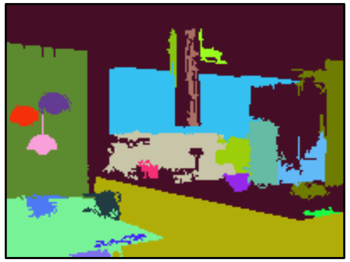
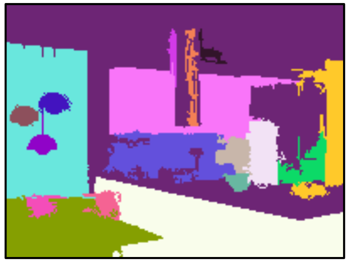
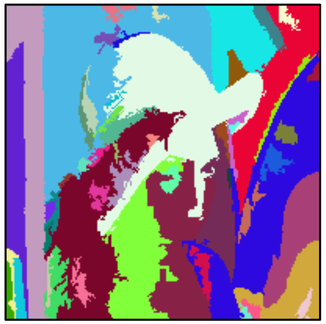
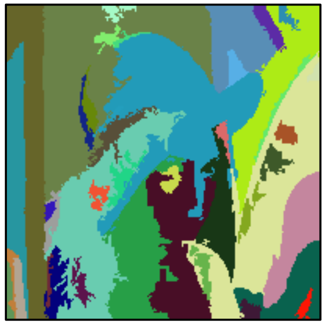
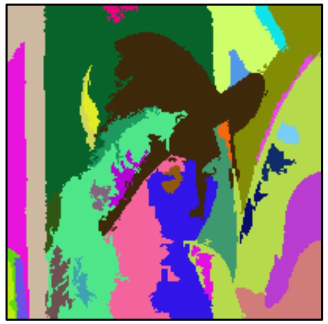
Ritsumeikan University

David Esparza Alba

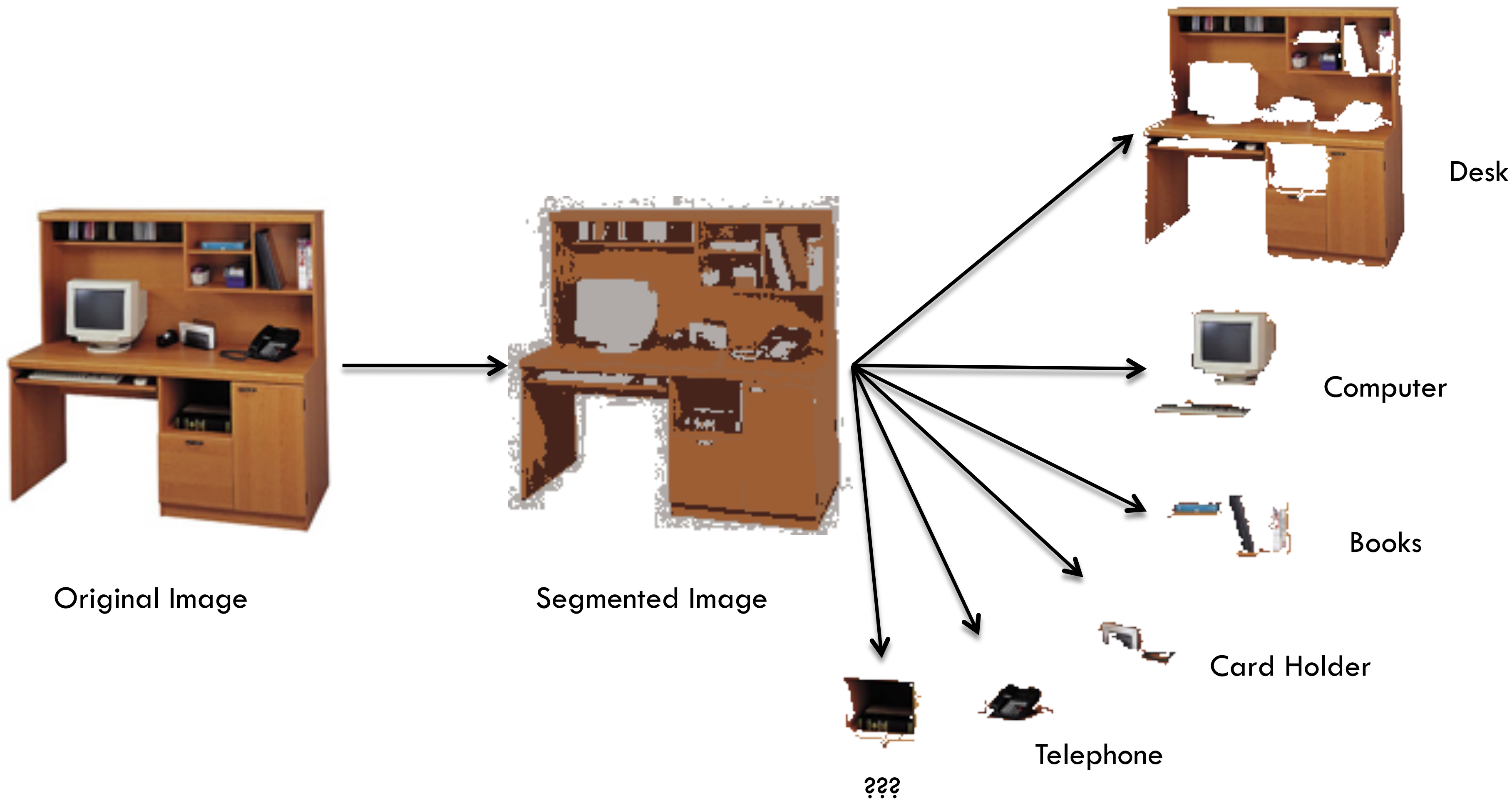
Results

Original Image	After Merging (TH = 25)	After Merging (TH = 50)	After Merging (TH = 100)
			
	32/507 Objects (93.7% reduced)	25/507 Objects (95.0% reduced)	17/507 Objects (96.6% reduced)
			
	14/128 Objects (89% reduced)	11/128 Objects (91.4% reduced)	8/128 Objects (93.7% reduced)
			
	23/197 Objects (88.3% reduced)	11/197 Objects (94.4% reduced)	8/197 Objects (96% reduced)

Results

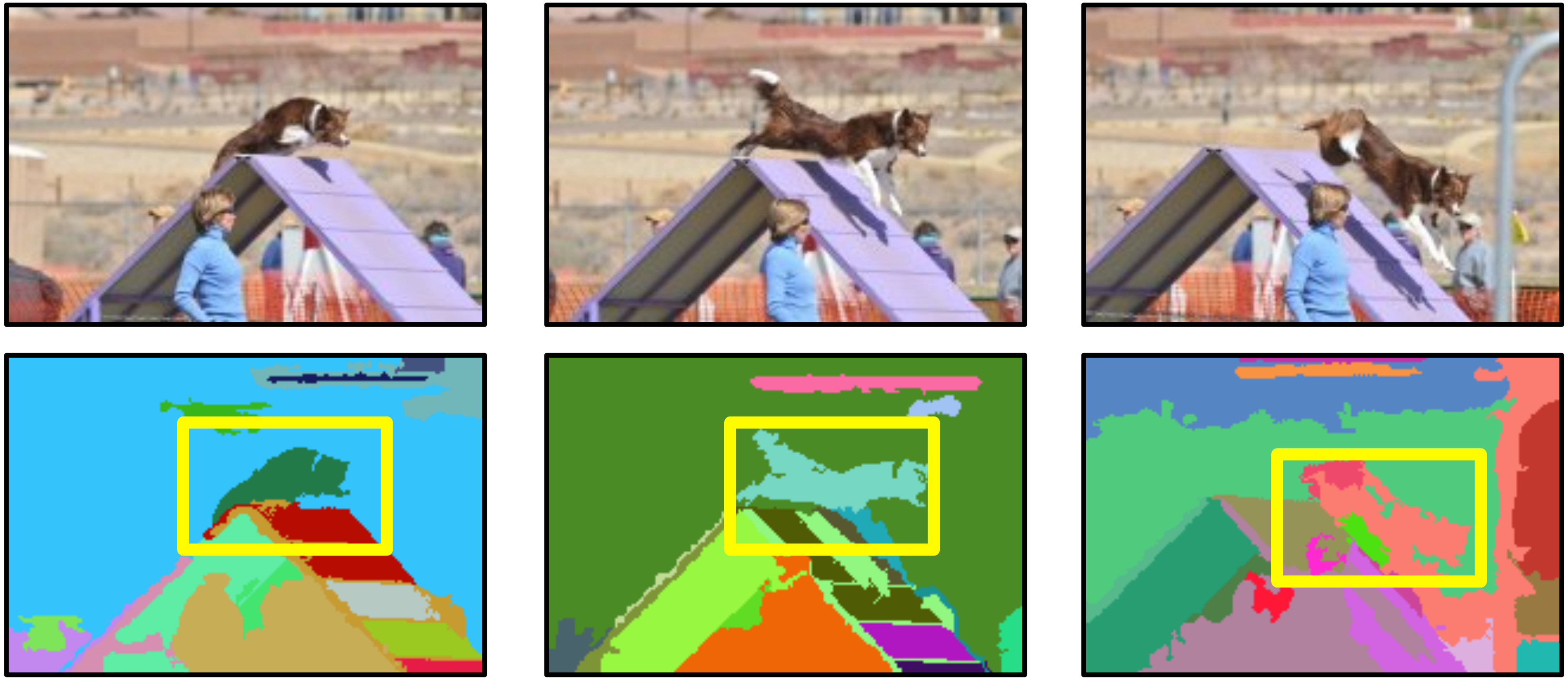
Original Image	After Merging (TH = 25)	After Merging (TH = 50)	After Merging (TH = 100)
	 14/48 Objects (70.8% reduced)	 10/48 Objects (79.2% reduced)	 9/48 Objects (81.25% reduced)
	 32/470 Objects (93.2% reduced)	 23/470 Objects (95.1% reduced)	 19/470 Objects (96.0% reduced)
	 55/718 Objects (92.3% reduced)	 43/718 Objects (94.0% reduced)	 34/718 Objects (95.3% reduced)

Possible Application



Identify different object in image. This also can be used un security systems, robotic vision, etc.

Possible Application



Once the amount of information of the image has been reduced to only a small number of objects, it is easier to identify the location of certain objects in a video sequence.

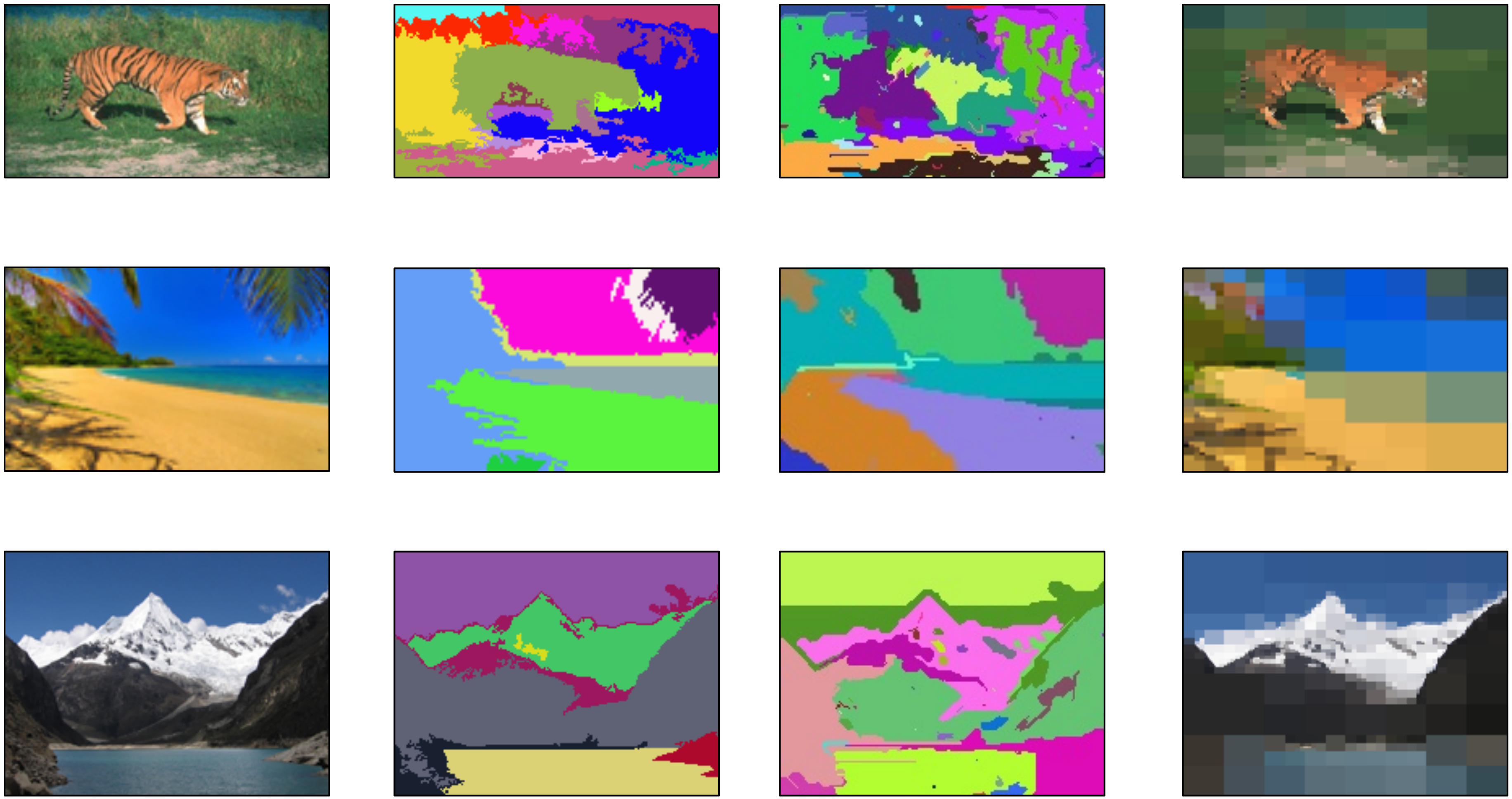
Other Implemented Methods

Ritsumeikan University

David Esparza Alba

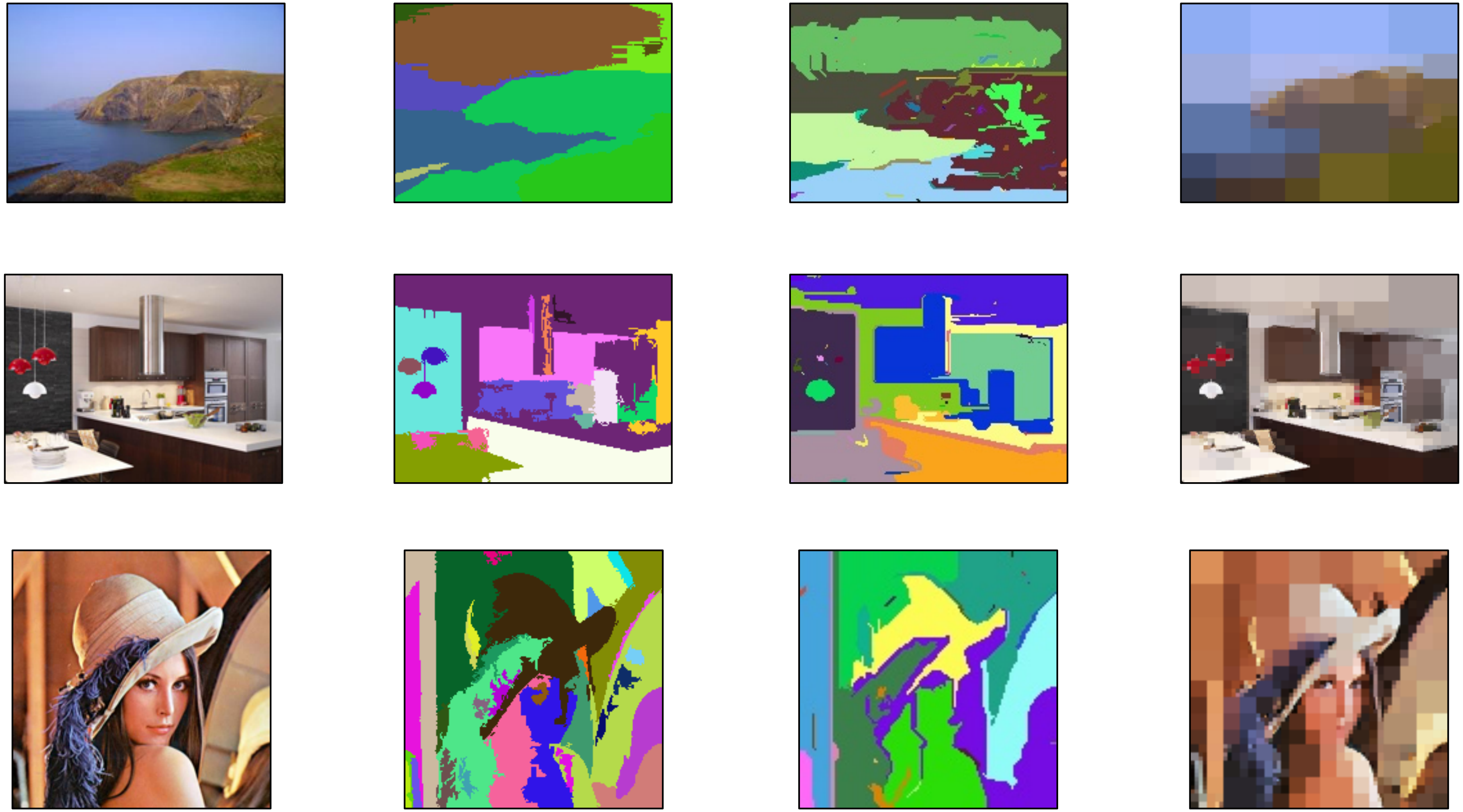
Other Implemented Methods

Original Image	Proposed Method	MST Segmentation	Quad Tree Segmentation



Other Implemented Methods

Original Image	Proposed Method	MST Segmentation	Quad Tree Segmentation



Future Work

David Esparza Alba

Ritsumeikan University

Future Work

1. Implement a robust algorithm using a Bayesian approach instead the split and merge steps.

2. Optimize the proposed method in order to reduce the execution time.

3. Improve the algorithm to be capable of deal with a big amount of data.

Thank You Ritsumeikan!!

Thank You JICA!!