

Document Objective

This document is a technical report for the 'Object based Image Fusion and Change Detection tools for Opticks' GSOC 2013 project. This report gives detailed information and overview of the software system itself, design of the system, the implemented technologies, and the working Environment. This document will furnish the following:

- Modular Details
- Technical Details
- User Guidelines

Plug in Overview

Change detection is the process of obtaining satellite images at different time stamps of the same geographic area and comparing them to and any changes that might have taken place in that area. The aim of the product is to focus on analyzing the changes based on different criteria, such as the spectral and geometrical attributes of the objects in the images. This module requires registered images as input for analysis.

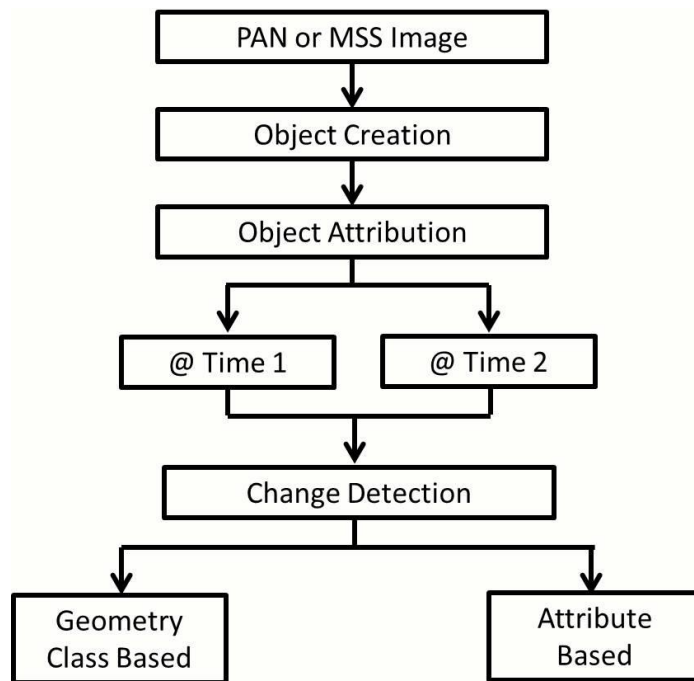


Figure 1: Change Detection Overview

Image Fusion is the process of combining high spectral information present in Multispectral data with high spatial detail present in panchromatic data. Multi-spectral data provides various spectral bands which helps in better discrimination between objects. But the spatial resolution is relatively lower than panchromatic data. High spatial resolution plays an important role in delineating very small features present especially in urban areas. Thus combining both datasets creates a new dataset which has high discrimination capabilities and better boundary delineation potential. This software requires accurately geo-referenced Multi-spectral and Panchromatic data of same time in order to create the fused image.

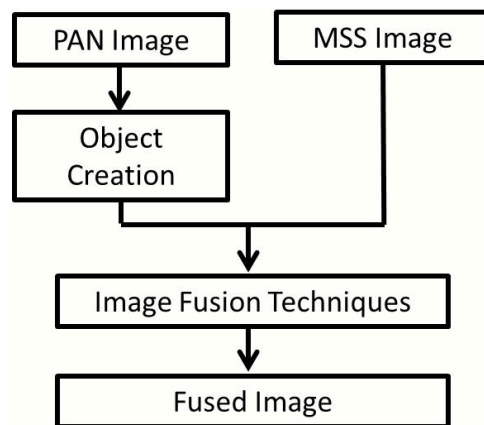


Figure 2: Image Fusion Overview

Scope and Perspective

The software provides the following facilities to the user:

- Segmentation
- Change Detection
 - Attribute Query Based
 - Geometry Class Based
- Image Fusion

1. Change Detection and Analysis

The Change Detection software is a tool to detect changes between two images of a place over time. This change analysis may assist in several applications like urban planning, assessing damages caused by natural disasters, defense monitoring activities.

2. Image Fusion

The fusion tool fuses a PAN image and a correspondingly registered MSS image. The fused image has salient information from both the images. The fused image can have complementary spatial and spectral resolution characteristics. Image fusion can have applications in image classification, medical imaging, digital camera applications etc.

Algorithms and Techniques

Mean Shift Segmentation:

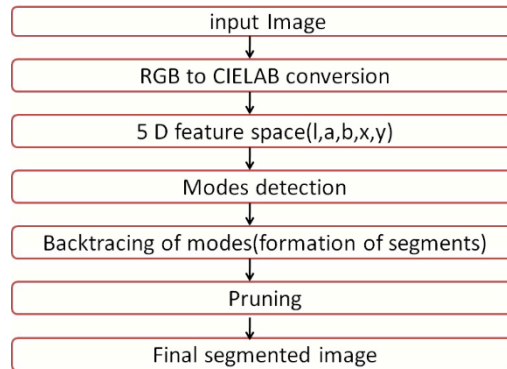


Figure 3: Mean shift segmentation workflow

The above flow diagram shows the steps involved in the mean shift segmentation^{1]} module. The first step is to convert the input images from RGB to CIELAB color space. RGB space is perceptually non-uniform while perceptually uniform spaces are better in approximating human vision. Both LUV and LAB color spaces were evaluated on their performance. But LUV did not give results as good as LAB space, which is a more uniform one than LUV. Hence, CIELAB is chosen for implementation. The data is then converted to 5 dimensional feature space, the other two features being the location of the pixel, x and y . Then clustering is done on this data which has both color components (L, a, b) and space component (x, y). This allows clustering of those points which are spectrally and spatially close. Dense regions in the feature space i.e. local maxima correspond to the modes. Points in feature space collapsing to one mode form an object. Backtracking the modes helps in object segmentation. After the objects are obtained by the previous step, the results are further pruned based on the following criteria:

- Spatial Bandwidth
- Color Bandwidth
- Minimum Region Area

Spatial Bandwidth signifies that how far two objects can be, so that they are allowed to merge. Color Bandwidth controls permissible difference in color values of two merging candidate objects. While, the minimum region area parameter controls the minimum size of the objects in the final result. Objects apart from each other by a value lesser than a normalized distance and having pixels lesser than a minimum number of pixels are merged into a single object. Thus, giving us the final segmentation result.

Change Detection Techniques

Change Detection is usually based on shape changes in the images at two time frames. Now, since here we also know the objects and their attributes, change detection and analysis is also possible on the basis of object attributes along with the geometry of the objects. In this process, once the segmentation is complete giving us the objects in the image, we calculate their attributes and the Change Detection Module takes over from here. The user is provided with the option of proceeding with Attribute Based Change Analysis or Geometric Class Based Change Analysis. The two techniques are discussed in the following sections.

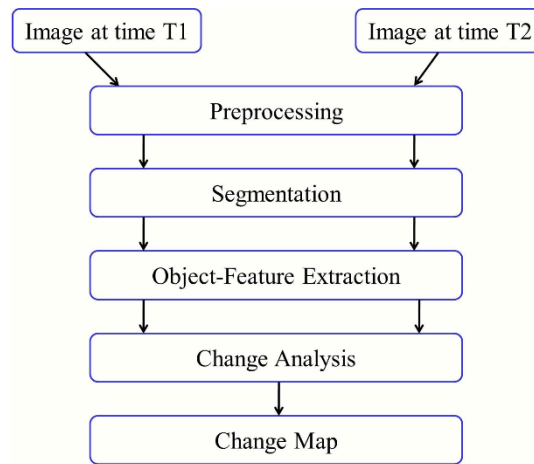


Figure 4: workflow of the algorithm.

a) **Attribute Based Techniques:**

For the objects given by segmentation module, we calculated the following attributes and prepared a feature vector for each of the object. For the following attribute notations, $I(i,j)$: Segmented Image (has object id at each pixel location)
 $Img(i,j,k)$: Original Image (has color values). $k = f1, 2, 3$ for red, green and blue bands

$$A = \sum_{i=1}^{rows} \sum_{j=1}^{cols} \alpha$$

where,

$$\alpha = \begin{cases} 1 & \text{If } I(i,j) = \text{current ObjectID} \\ 0 & \text{Otherwise} \end{cases}$$

$$2) \text{ Perimeter}$$

$$P = \sum_{i=1}^{i=\text{rows}} \sum_{j=1}^{j=\text{cols}} \alpha$$

where,

$$\alpha = \begin{cases} 1 & \text{If } I(i,j) = \text{current ObjectID and its boundary pixel of that object.} \\ 0 & \text{Otherwise} \end{cases}$$

$$3) \text{ Roundness}$$

$$R = \frac{4 * \pi * A}{P^2}$$

$$4) \text{ Compactness}$$

$$C = \frac{P}{3.53 * \sqrt{A}}$$

$$5) \text{ Centriod}$$

$$6) \text{ Contrast}$$

$$F_{\text{con}} = \frac{\sigma}{\alpha_4^n}$$

Where,

$$\alpha_4 = \frac{\mu^4}{\sigma^4}$$

μ^4 is the fourth moment about the mean, σ is std. deviation.

$$7) \text{ Coarseness}$$

$$8) \text{ Direction}$$

$$9) \text{ Roughness}$$

$$10) \text{ Mean red}$$

$$\mu R = \sum_{i=1}^{i=\text{rows}} \sum_{j=1}^{j=\text{cols}} C$$

where,

$$C = \begin{cases} \text{Img}(i,j,1) & \text{If } I(i,j) = \text{current ObjectID} \\ 0 & \text{Otherwise} \end{cases}$$

$$11) \text{ Mean Green}$$

$$\mu G = \sum_{i=1}^{i=\text{rows}} \sum_{j=1}^{j=\text{cols}} C$$

where,

$$C = \begin{cases} Img(i, j, 2) & \text{If } I(i, j) = \text{current ObjectID} \\ 0 & \text{Otherwise} \end{cases}$$

$$\mu_B = \frac{\sum_{i=1}^{i=rows} \sum_{j=1}^{j=cols} C}{A}$$

where,

$$C = \begin{cases} Img(i, j, 3) & \text{If } I(i, j) = \text{current ObjectID} \\ 0 & \text{Otherwise} \end{cases}$$

13) Standard deviation Red

$$\sigma_R = \sqrt{\frac{\sum_{i=1}^{i=rows} \sum_{j=1}^{j=cols} C}{A}}$$

where,

$$C = \begin{cases} (Img(i, j, 1) - \mu_R)^2 & \text{If } I(i, j) = \text{current ObjectID} \\ 0 & \text{Otherwise} \end{cases}$$

14) Standard deviation Green

$$\sigma_G = \sqrt{\frac{\sum_{i=1}^{i=rows} \sum_{j=1}^{j=cols} C}{A}}$$

where,

$$C = \begin{cases} (Img(i, j, 2) - \mu_G)^2 & \text{If } I(i, j) = \text{current ObjectID} \\ 0 & \text{Otherwise} \end{cases}$$

15) Standard deviation Blue

$$\sigma_B = \sqrt{\frac{\sum_{i=1}^{i=rows} \sum_{j=1}^{j=cols} C}{A}}$$

where,

$$c = \begin{cases} (Img(i,j,3) - \mu_B)^2 & \text{If } I(i,j) = \text{current ObjectID} \\ 0 & \text{Otherwise} \end{cases}$$

16) Class

The User is provided with a UI for supervised classification. The classification is based on Mahalanobis Distance [5]. The user can first give the number of classes he/she wants the objects in the image to be classified into. Then training samples (objects) have to be selected from the images and a class number is assigned to each of them with the help of the interface. Once a sufficient number of training samples (objects) have been selected for each class, the user can proceed to classification. step which then assigns each object in both the images a unique class number. The class id of an object i is calculated using the Mahalanobis Distance metric which is defined as

$$D^2_{i,j} = (X - \mu_j)^T S^{-1} (X - \mu_j)$$

Where,

$D^2_{i,j}$: squared Mahalanobis distance of object i from class j

X : feature vector of the object to be classified

μ_j : mean feature vector of the training samples of class j

S : is the covariance matrix of class j

The class which has the least Mahalanobis distance to the object i is the class of that object.

The four texture features, namely, coarseness, contrast, direction and roughness are Tamura texture features [4].

The user is provided with a UI which shows the range of the attributes for the images given as input. This is an editable form, which the user can edit to form a query. The objects which satisfy all the attribute range values provided by the user are displayed as a result, while all the other objects are omitted. If none of the objects satisfies the given query, a suitable message is displayed saying that the user needs to select some other range values.

b) Geometric Class Based Techniques:

This technique is based on the comparison of the size of an object in the two input images. The first step is Overlay. Both segmented images are overlaid to generate a combined segmented image. This step helps in capturing the objects where some shape change has happened. The new polygons thus created contain shape change information of both the

Figure 5: Generating overlay image from two segmented images

images.



Now, change is classified based on the change in the area of every object present in the overlay image obtained above. The two parameters High and Low taken as input from the user earlier are used here to quantify the change in the area of an object from initial time stamp to final time stamp. These parameters divide the values between 0 and 1 into three compartments, namely, low, moderate, and high. The technique used is as follows:

- $N \leftarrow$ number of objects in the overlay image I_o .
- for $i = 1$ to N
 - $j \leftarrow$ object of I_{t1} which i is a part of
 - $k \leftarrow$ object of I_{t2} which i is a part of
 - $r_{t1} \leftarrow A_i/A_j$ and $r_{t2} \leftarrow A_i/A_k$
 - Assign geometric class C to i depending on the change in ratio. (Refer Class Key figure below)

The figures given below demonstrate the different possible cases:

From	To	Color	Sample	Category
Low	Low	Yellow		Unknown
Low	Moderate	Cyan		Partial Appearance
Low	High	Green		Appearance
Moderate	Low	Light Blue		Partial Disappearance
Moderate	Moderate	Magenta		Unknown
Moderate	High	Red		Disintegrate
High	Low	Black		Disappearance
High	Moderate	Light Red		Merge
High	High	White		No Change

Figure 6: Class Key

Image Fusion Techniques

Object Based Image Analysis pipeline can be run on Multi-spectral data or on Pan-sharpened Multi-spectral data. Pan-sharpened Multi-spectral data would lead to better discriminating and delineating capabilities. The algorithm Implemented is "**Object Oriented Gaussian Linear Stretching Image Enhancement(OO-GLSI)**".

Object Oriented Gaussian Linear Stretching Image Enhancement(OO-GLSI) :

This algorithm [3] has been developed at the Lab for Spatial Informatics, IIIT Hyderabad. It basically maps the data distribution of multi-spectral bands in terms of data distribution of panchromatic band for a given object. It sharpens each multi-spectral band independently and thus preserves the spectral information of the multi-spectral data. Following are the steps of the algorithm:

- Choosing the appropriate window size (depends on the image to be sharpened)
- Compare the data distribution curves of respective windows
- Extract relevant data parts from high resolution band which generates the relevant data curves based on the number of objects that the distribution can clearly depict.
- Analysis of the data distribution curve of low resolution band. Depending on the outcome of the above step and the occurrence of one or more objects in high resolution band, the data is suitably modified based on its distribution to provide for suitable correspondence.
- *Object Matching*: This step is needed because an object can take higher (bright) values in one of the low resolution bands but have lower (dark) values in the other narrow band which places the Gaussian of the object at opposite sides.
- *Value Mapping*: Each value in high resolution band is linearly mapped to a value in low resolution band which generates a value for the fine resolution sharpened image pixel in the respective low resolution band.
- *Interpolation*: The unmapped points obtained due to pruning are filled by interpolation of the surrounding pixels, centered on the respective unmapped pixels or holes.

System Working Model

Change

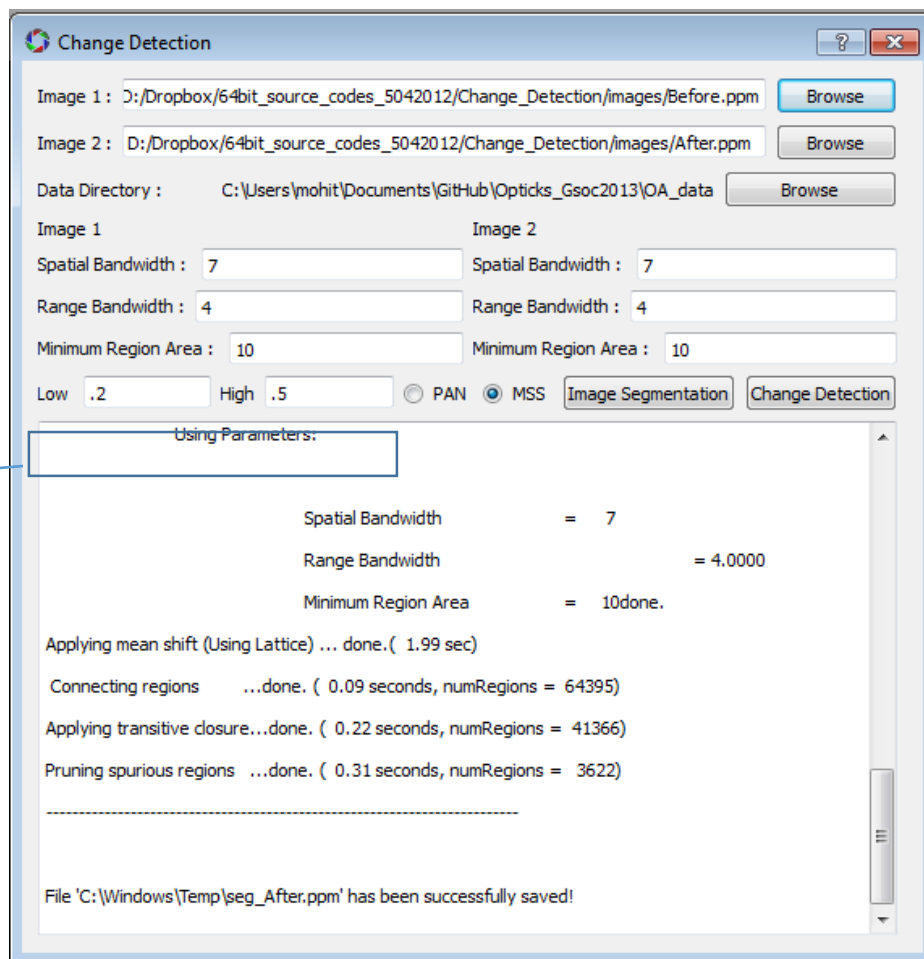
After the segmentation module, we proceed to change detection step. We start by providing the values of the two parameters High and Low and then executing the change detection process as shown in the figure below.

Both the images are displayed for the selection of training samples. Click on the image for selecting an object to be put in training set.

Detection Module

completion of

the change detection process as shown in the figure below.



Enter low and high Values

Click on "change detection", after filling values.

Figure 7: Setting up input for Change Detection step.

Classification, Enter the number of classes and click the 'Select Training Samples' button to select training samples (objects) to be used in classification. Both the images are displayed for the selection of training samples. Click on the image for selecting an object to be put in training set.

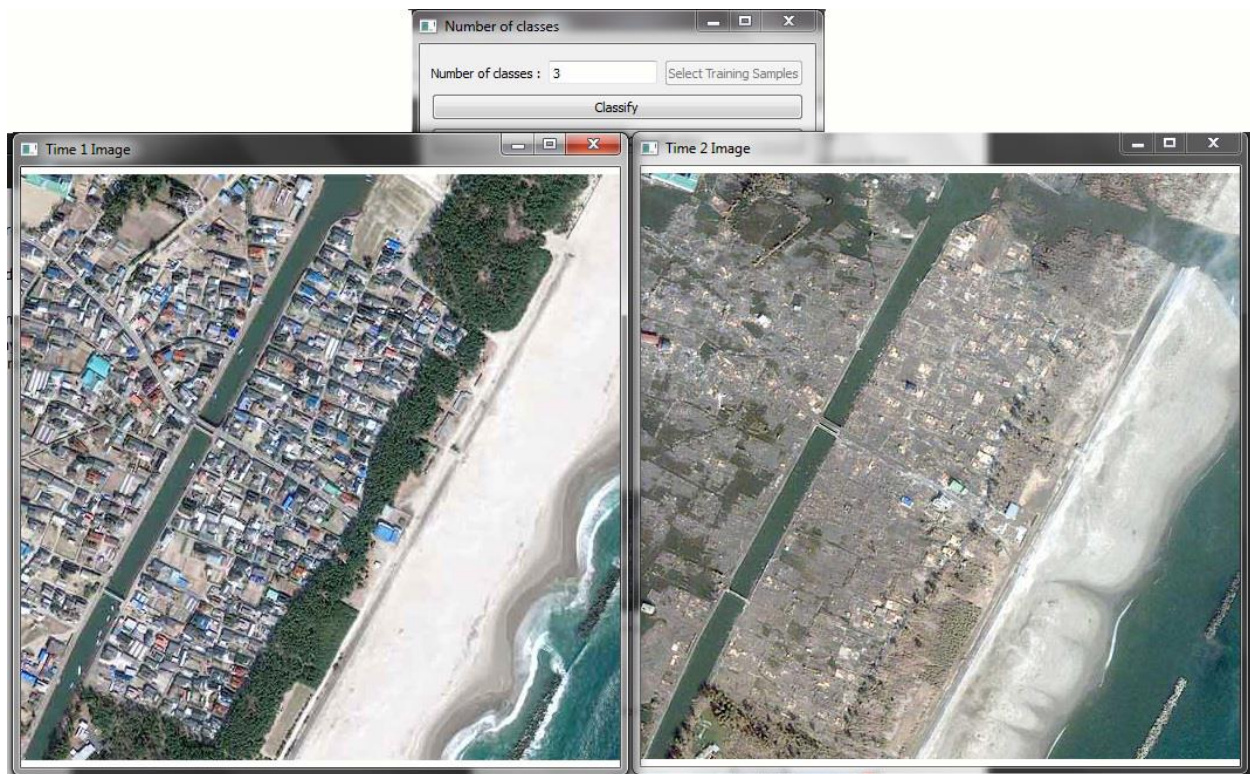


Figure 8: Selecting Training samples.

Clicking on an object in the image will generate a UI popup with a drop-down menu containing the classes. The user is required to put ample number of training samples in each class. Similarly, training samples have to be pushed from Time 2 Image also.

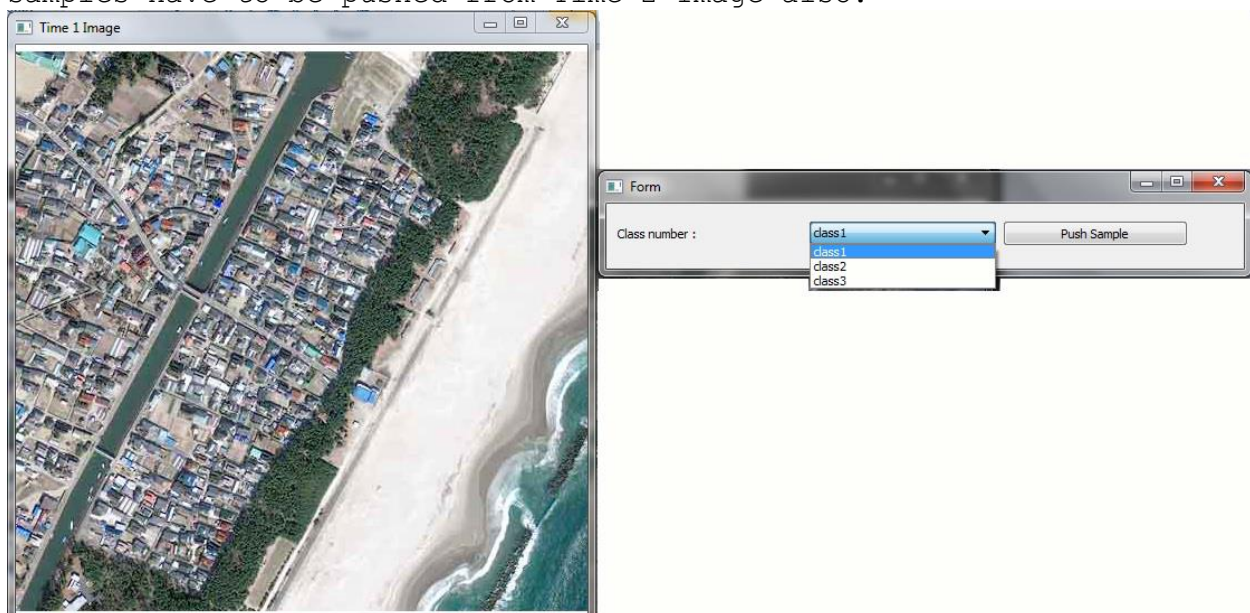


Figure 9 : Pushing Training Samples from Time 1 Image.

After the user is done pushing training samples, both the images and the sample push UI can be closed. Then push the 'Classify' button to perform classification. The UI also allows the user to proceed without classification. After this step a new window is generated which provides the user with option to choose from. The user may proceed either with **Geometric Class Based** or **Attribute Based Change Detection**.

- **Attribute Based Change Detection:**

Attribute Based Change Analysis Module gives you an UI which shows the value ranges of all the attributes. The values are editable. The screenshot does not contain the class attribute. If the user has performed classification, then class attribute will also be displayed here along with the other attributes.

The figure shows two screenshots of a software window titled 'Form'. The left screenshot shows a window with two buttons: 'Class Based' and 'Attribute Based'. The right screenshot shows the 'Attribute Based' window, which is titled 'Set Range :'. It contains a list of attributes with their current values and ranges, and a 'Reset' button for each attribute. The attributes and their values are: Area (10, 37797), Roundness (0.0303, 1.5514), Compactness (0.8062, 5.7645), Mean Red (5, 254), Mean Green (14, 254), Mean Blue (7, 254), Std Red (1.3008, 72.0069), Std Green (1.3008, 72.9353), Std Blue (1.3008, 72.7482), Perimeter (9, 2340), Contrast (0.266, 46.8702), Roughness (2.4695, 63.5565), Coarseness (2, 62.1824), and Direction (0, 7.5503). There is also a 'Reset All' button. At the bottom, there are fields for 'Minimum Value' and 'Maximum Value', radio buttons for 't1' and 't2', and a 'Generate Image' button.

Attribute	Value	Range	Reset
Area	10	37797	Reset
Roundness	0.0303	1.5514	Reset
Compactness	0.8062	5.7645	Reset
Mean Red	5	254	Reset
Mean Green	14	254	Reset
Mean Blue	7	254	Reset
Std Red	1.3008	72.0069	Reset
Std Green	1.3008	72.9353	Reset
Std Blue	1.3008	72.7482	Reset
Perimeter	9	2340	Reset
Contrast	0.266	46.8702	Reset
Roughness	2.4695	63.5565	Reset
Coarseness	2	62.1824	Reset
Direction	0	7.5503	Reset

Reset All

Minimum Value :

Maximum Value :

☐ t1 ☐ t2

Generate Image

Figure 9: Attribute based Change Analysis window

The UI also provides the option of resetting the individual values as well as all at once. After the user has entered the desired values, an image has to be chosen as reference as shown in the figure below.

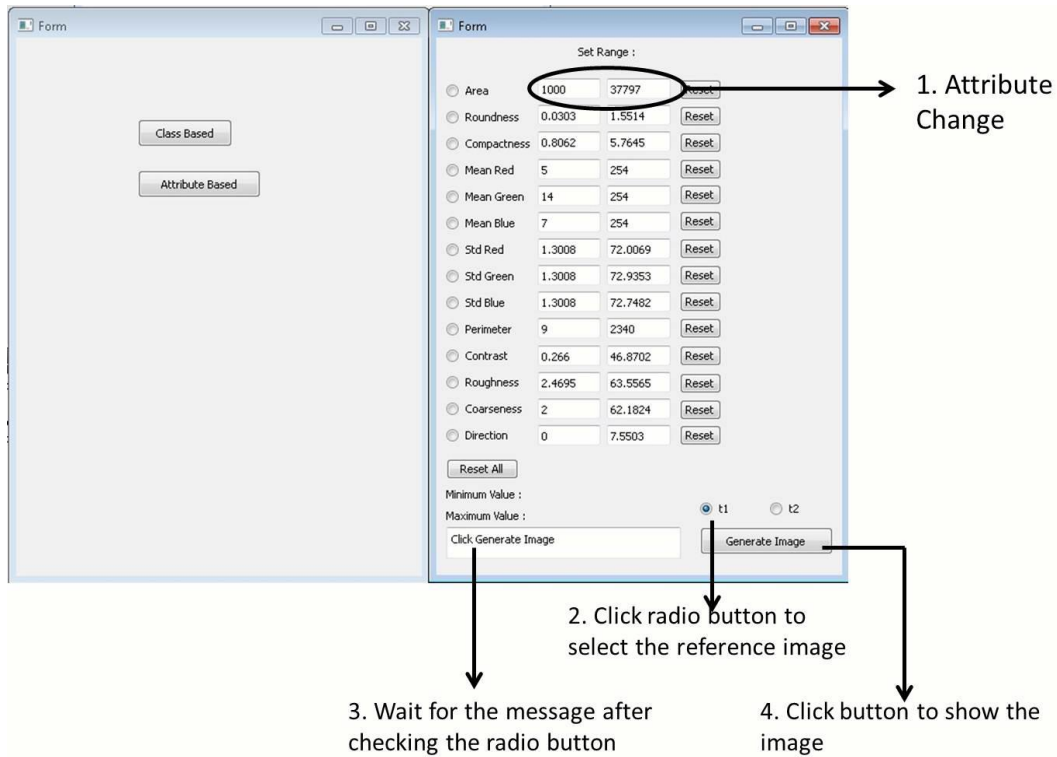


Figure 10: Setting up input for attribute based analysis step

Two images displayed with only those objects which satisfied the given query.

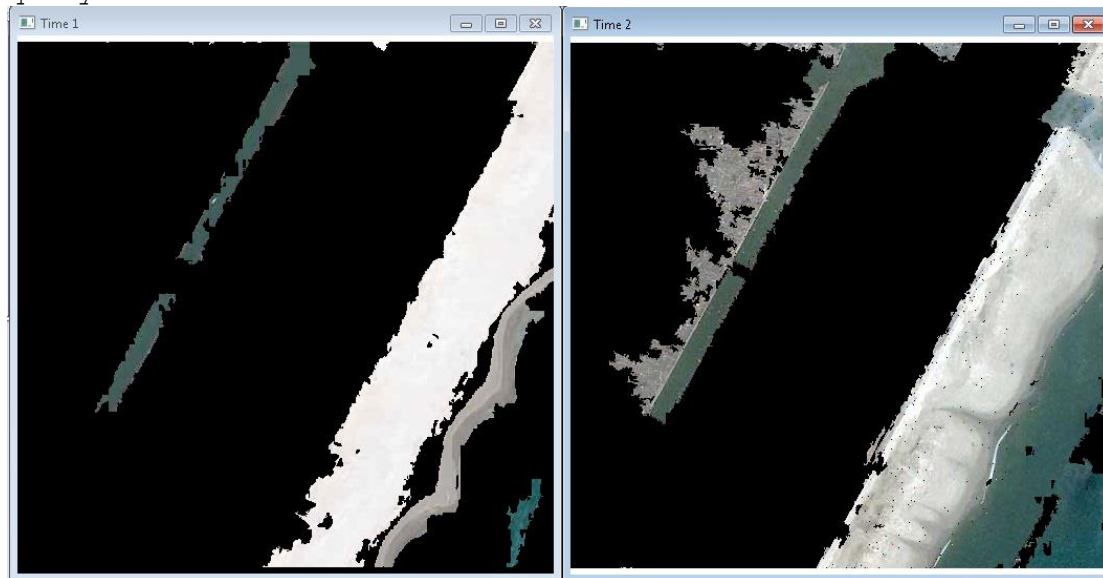


Figure 11: Image at time 1 and 2 satisfying the given query.

Attributes of any object can be seen by clicking on that object in any of the two images.

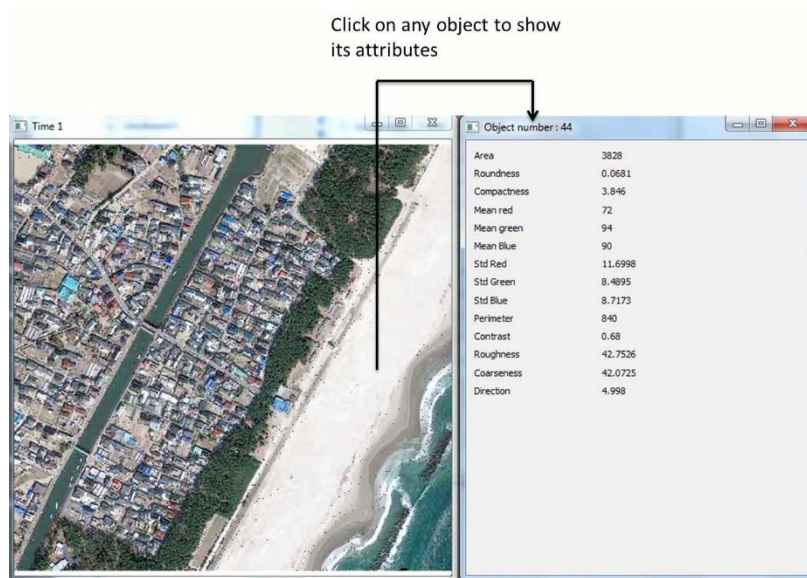


Figure 12: Attributes of an object seen in a new window.

- **Geometric Class Based Change Detection:**

Geometric Class Based Change Analysis Module gives you an UI which shows the classes.

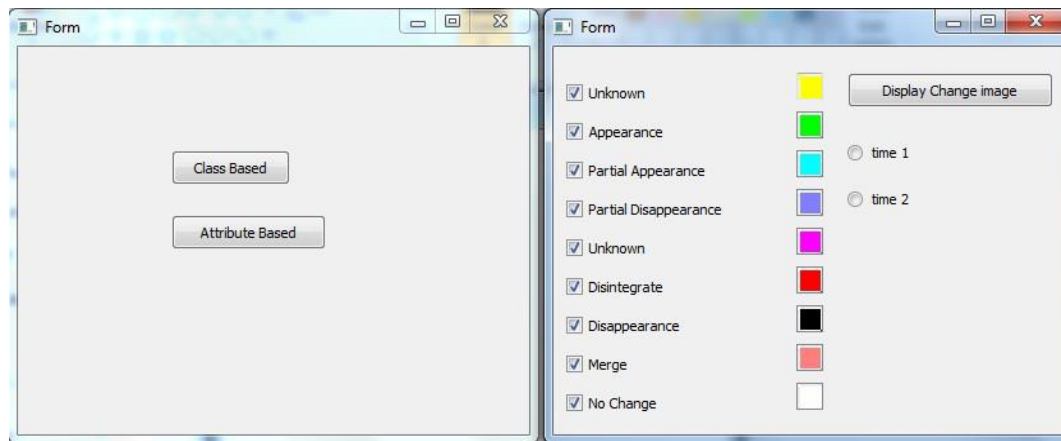


Figure 13: Geometric class based analysis Window.

The user can check the classes he wants to include in the change image and then click on the 'Display Change Image' button.

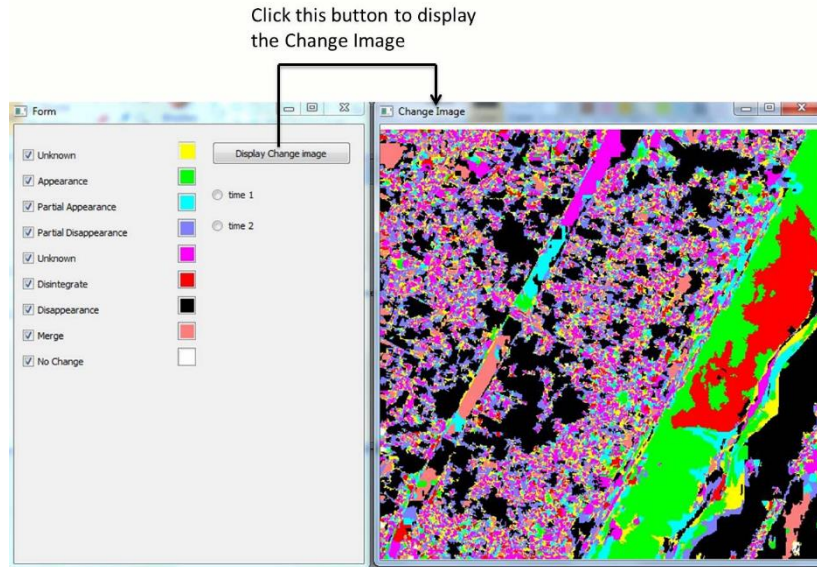


Figure 14: Change Image with the color coding as shown in the left window

The user can click on an object in the change image and then on either 'time 1' or 'time 2' in the parent window to display that particular object in that particular image.

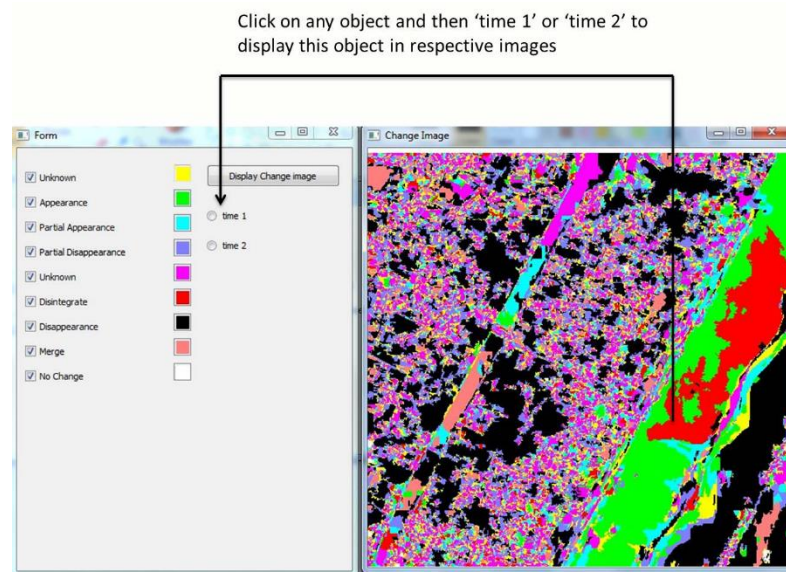


Figure 15: Displaying any object at time 1.

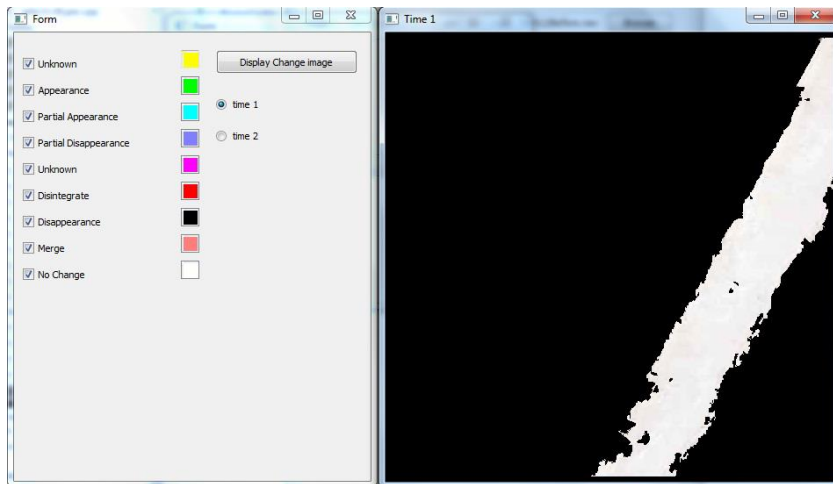


Figure 16: Selected object at time 1.

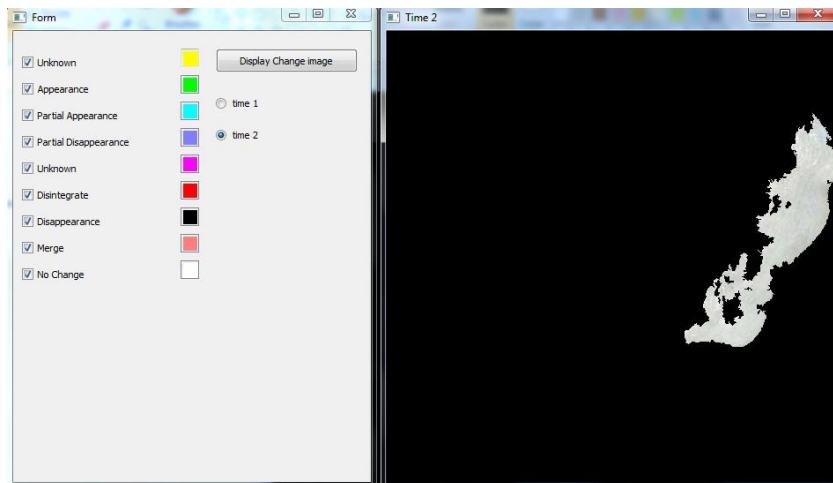


Figure 17: Selected object at time2.

Classes of interest can also be selected for analysis.

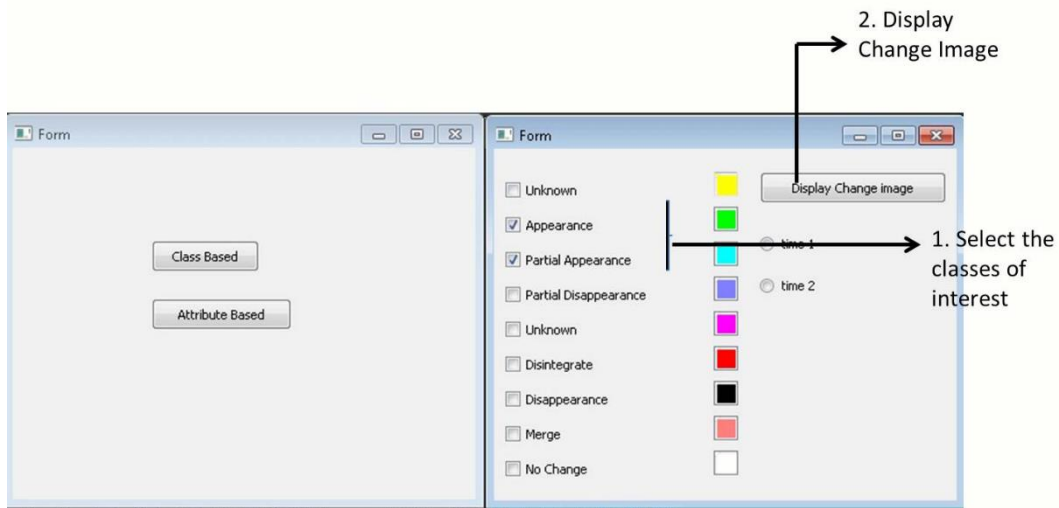
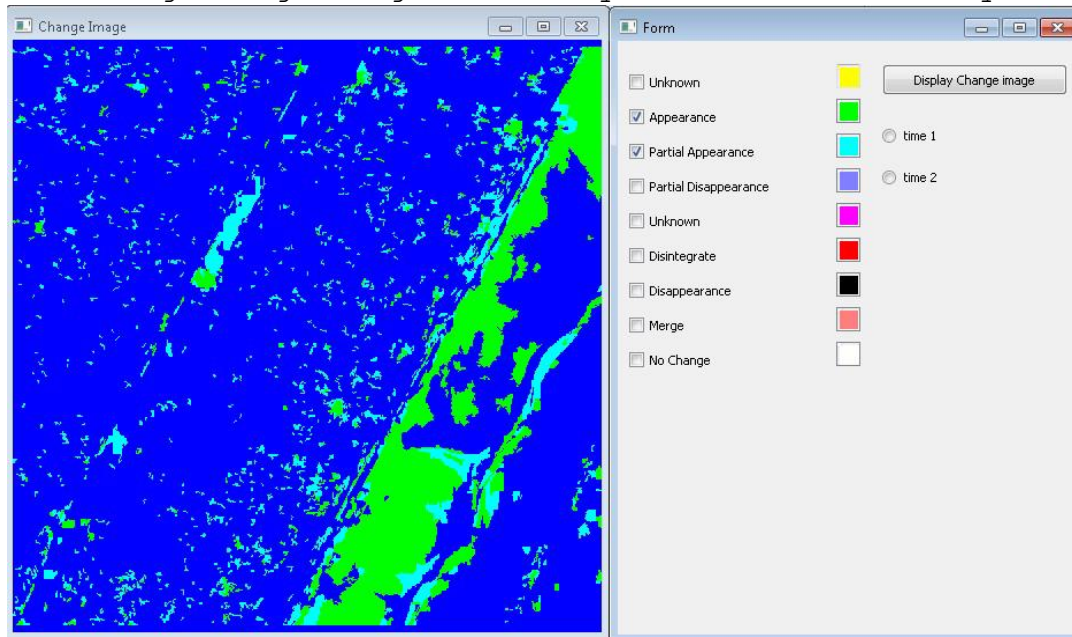


Figure 18: Selecting Class of interest.

Resulting Change Image with only classes selected by user.



BIBLIOGRAPHY

1. D. Comaniciu and P. Meer. **Mean shift: a robust approach toward feature space analysis**. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 24(5): 603 -619, may 2002. ISSN 0162-8828. doi: 10.1109/34.1000236.
2. Q. Meng, B. E. Borders, C. J. Cieszewski and M. Madden. **Closest Spectral Fit for Removing Clouds and Cloud Shadows** photogrammetric Engineering and Remote Sensing, 75(5):569 - 576, may 2009.
3. A. Khandelwal, K. S. Rajan. An Efficient Algorithm for Generating Pan Sharpened Quick Look Images.
4. H. Tamura, S. Mori, and T. Yamawaki. Texture Features Corresponding to Visual Perception Transactions on System, Man and Cybernetic, IEEE, 6:460 - 473, june 1978.
5. P. C. Mahalanobis. On the generalised distance in statistics Proceedings National Institute of Science, India, 2: 49 - 55, april 1936