Project title: Using Combined Difference image and unsupervised clustering algorithm for Image Change Detection

Abstract:

An easy and advantageous unsupervised based method using combined difference image and k-means clustering (CDI-K) is presented for synthetic aperture radar (SAR) image change detection task. To find change map of two time differed images, the subtraction operator and the log ratio operator is applied respectively which produces two change map image. Mean filter is used for making change map smooth and local area constant. Median filter is also used for preserving the edges of image. Median and Mean filters are applied to log-ratio and subtracted images respectively. Using median and mean filter, a combination framework is applied to change map images to find a better change map. At last, K-means clustering algorithm is applied to change map images to separate it into changed area and unchanged area. In this experiment sample change map images with ground truth is used to find the performance of this method. Then, this method is applied to SAR image. K-means clustering performance depends on random cluster point selection which is arbitrary in this method. It is one of the drawback of CDI-K method. A combination of fuzzy c means clustering method and K means clustering method is proposed to remove this problem. Image fusion procedure based on output performance is also a drawback of this method. As radar image may be deformed with speckle noise, the image can be de-noised using probabilistic patch based(PPB) algorithm. The performance between k-means clustering and combination of fuzzy c means clustering method and K means clustering method is compared with change detection image, false alarm, missed alarm, overall error and kappa index.

Technical Discussion:

Change Detection:

Change detection is a clustering process which separates the pixels of two image into two cluster region. This two clusters are changed area and unchanged area. For Synthetic aperture radar (SAR) image research, change detection technique is important technique. These change detection techniques can be used for agricultural survey, forest monitoring, natural disaster monitoring and urban change analysis.

Why unsupervised method?

Based on clustering process change detection can be classified into two groups which are supervised cluster and unsupervised cluster. On drawback of supervised technique is that this method requires some ground truth or label samples for training of supervised classifiers. Unsupervised method is the best option when ground truth or some kind of labeling is not available.

Here, unsupervised change detection algorithm is divided into three steps:

1. Image preprocessing
2. Obtaining the difference image
3. Analyzing the difference image and post processing.

General Procedure:

For SAR image, speckle noise is the primary disruption for change detection. As a result, probabilistic patch based filter is used to minimize speckle noise in preprocessing steps. Then, for obtaining difference image two methods are used: subtraction operator and the log ratio operator. After minimizing the initial noise, we obtained the de-noise image X_1 and X_2 from I_1 and I_2 image. Then using following subtraction operator and log ratio operation two difference image D_s and D_l is obtained,

$$D_s = \left| X_1 - X_2 \right| \qquad [1]$$

$$D_l = \left| log \frac{X_2 + 1}{X_1 + 1} \right| = \left| log(X_2 + 1) - log(X_1 + 1) \right| \qquad [2]$$

For log ratio operator X_i+1 where, i=1,2 is used instead of X_i to avoid zero pixels' value case. This two subtraction operator is normalized back into 0 to 255 scale. Next step mean filter is applied to the difference image which is obtained by subtraction operator to obtain locally smooth region which may have major local consistency. Mean filter can minimize isolated pixel in difference image and local area consistent. The window size of mean filter is 11*11 and output is denoted as D_s`. Median filter is applied on the difference image obtained by log ratio operator to preserve the edge information. As a result, it may reserve shape of edge information. Median filter window will be 3*3 and output difference image is $D_l$`. Then, a combination operation is used to integrate the difference image obtained by mean and median filter. For combination framework regularization parameter α is used and following framework is used for combination of two filtered image,

$$D = \alpha D_s^{'} + (1 - \alpha) D_l^{'} \qquad [3]$$

This parameter α varies between 0 to1. This value should be small for making smoother and large for edge preservation. Too large weight will more number of false alarm and too small value will cause more number of missed alarm output. At the end, K means clustering algorithm is used to partition the combination framework output into two clusters which are changed area and unchanged area.

K means Clustering:

K means clustering is one of easiest unsupervised clustering method which can solve the clustering problem easily. This algorithm follows a simple way to classify data into certain number of clusters. In data mining, k-means clustering partitions n observations into k number of cluster in which each observation belongs to cluster with nearest mean from cluster center. It defines k center one for each cluster. The center location plays a vital part in k means clustering algorithm because based on this centroid location clustering performance differs. Best way to find a good possible outcome is to place initial centroid away from each other as possible. Then it clusters data points into centroids based on minimum distance from centroid. Then this algorithm tries to find new centroid based on minimum distance. If no new centroid is possible to determine, it stops at that point. This algorithm tries to minimize square error function which is following,

$$J(V)=\sum_{i=1}^{c}\sum_{j=1}^{c_i}(\left\|x_i-v_j\right\|)^2 \qquad [4]$$

Where $\left\|x_i-v_j\right\|$ is the distance between x_i and v_j. c and c_i is the number of cluster center and number if data points in i^{th} cluster.

Fuzzy C-means Clustering:

Fuzzy clustering is the clustering process where data points belong to more than one group of cluster. In this process, items in same cluster are as similar as possible and items belongs to different classes as dissimilar as possible. This similarity measures may be distance, connectivity and intensity. Memberships are allocated to each data points. This membership indicates the degree each data points belong to each cluster. The fuzzy C-means clustering follows very similar algorithm like k means clustering. Chooses a number of clusters. Assigns each point coefficients randomly and repeat until the algorithm has converged. If point x has coefficients which gives the amount of degree in the kth cluster $w_k(x)$. For fuzzy c-means clustering the centroid of a cluster is C_k,

$$C_k=\frac{\sum_x w_k(x)^m x}{\sum_x w_k(x)^m} \qquad [5]$$

Performance Detection:

True Positive-number of times correctly identified when the given condition is fulfilled.

False Alarms/False Positive- it is also known as false alarm. When a result or outcome represents that given condition is fulfilled but in reality, it does not.

True Negative- number of negative number/conditions numbers which is correctly identified as negative numbers.

False Negative/Missed Alarms- number of positive number is incorrectly identified as negative numbers.

Overall Error: Overall error is the summation of false positive and false negative numbers.

True Positive rate (TPR) or sensitivity or hit rate: portions of the positive which is correctly identified as positive numbers or true events. It can be represented from following equations,

$$TPR = \frac{TP}{TP + FN} \qquad [6]$$

Where, TP and FP are numbers of true positive and false positive respectively.

False negative rate(FNR) or miss rate can be represented from following equations,

$$FNR = \frac{FN}{TP + FN} = 1 - TPR \qquad [7]$$

Kappa:

Kappa is the Cohen's kappa coefficient which evaluates inter-rater agreement for qualitative items in statistics. It is simple percent calculation that takes into account the chance of agreement occurring.

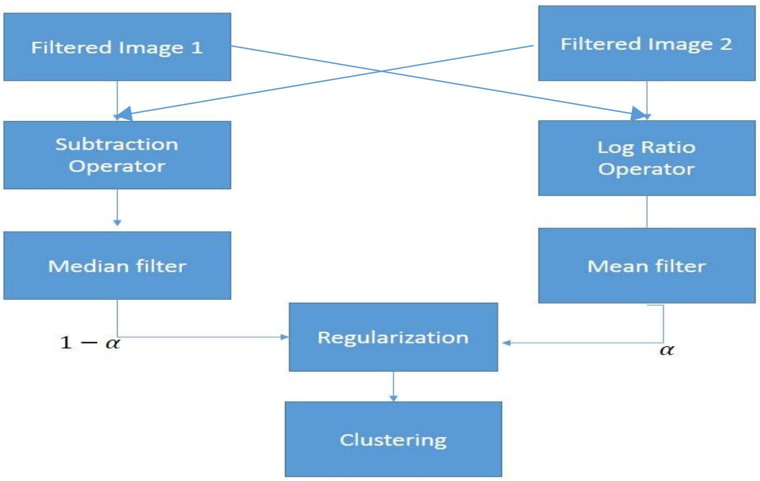
Cohen's kappa calculates the agreement between two raters who each classify N items into C categories which are mutually exclusive categories. The equation for kappa k is,

$$k = 1\frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_o} \qquad [8]$$

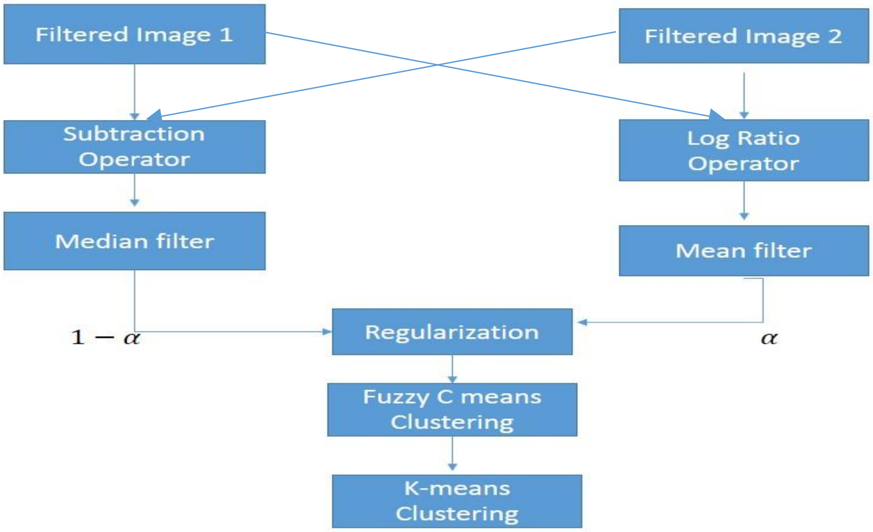
Where, P_o represents the relative agreements between raters and P_e represents hypothetical probability of chance agreement. When k=1 then the all raters are complete agreement with each other and k is less than or equal zero when no agreement between raters occur.

Flowchart of methods:

Methods 01- change detection using CDI-k



Method 02- Proposed Change Detection using fuzzy c means and k means clustering methods:



First diagram is the method described in corresponding paper and second diagram is method which is proposed to remove the drawbacks from first method. In proposed method, for improving the K-means clustering performance, combination of Fuzzy C means clustering and K means clustering is used.

Discussion of Results:

Table 01: False alarm, missed alarm, overall error and kappa index for Change detection using K-means clustering (CDI-K) and proposed method (combined fuzzy clustering and k-means clustering method.

Method	Alpha	False alarms	Missed Alarms	Overall Error	Kappa	Hit rate
K means Clustering	0	25	1961	1986	0.84	74.823
	0.1	135	1097	1232	0.908	85.916
	0.2	**163**	**1085**	**1248**	**0.907**	86.07
	0.3	172	1087	1259	0.906	86.044
	0.4	180	1091	1271	0.905	85.993
	0.5	185	1092	1277	0.9049	85.98
	0.6	186	1090	1276	0.905	86.005
	0.7	188	1091	1279	0.9047	85.993
	0.8	189	1092	1281	0.9046	85.98
	0.9	189	1092	1281	0.9046	85.98
	1.0	189	1094	1283	0.9044	85.954
	0	30	1920	1950	0.8455	75.349
	0.1	168	1034	1202	0.91072	86.725
	0.2	199	1014	1213	0.9101	86.981
	0.3	**208**	**1005**	**1213**	**0.9102**	87.097
	0.4	215	1002	1217	0.91	87.1356
	0.5	222	998	1220	0.909	87.1870

Fuzzy C means Clustering + K means Clustering	0.6	226	1000	1226	0.909	87.1613
	0.7	228	998	1226	0.909	87.1870
	0.8	228	999	1227	0.909	87.1742
	0.9	229	999	1228	0.909	87.1742
	1.0	230	1001	1231	0.909	87.1485

In table 1, CDI-k and proposed method perform is compared. For change detection method, based on their performance their corresponding change detection output is shown in figure 1.

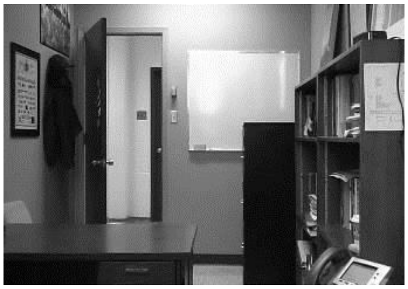


Figure 1.1: input image

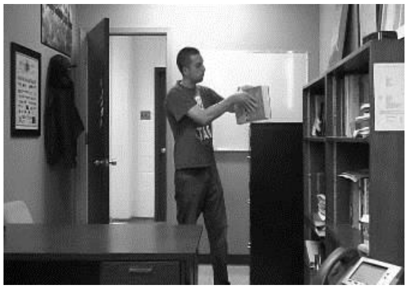


Figure 1.2: input image different time period

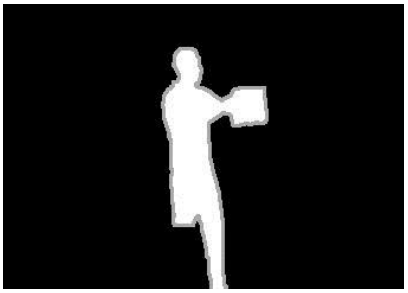


Figure 1.3: Ground Truth

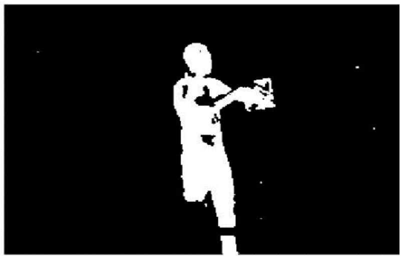


Figure 1.4: change detection using k means clustering (CDI-K) for alpha 0.0

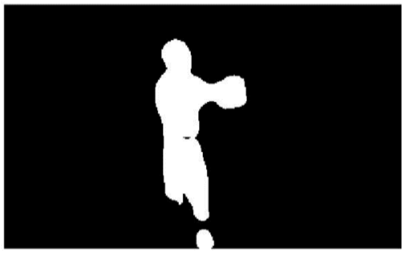


Figure 1.5: change detection using k means clustering (CDI-K) for alpha 0.3

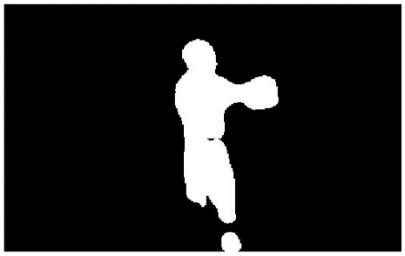


Figure 1.6: change detection using k means clustering (CDI-K) for alpha 0.5

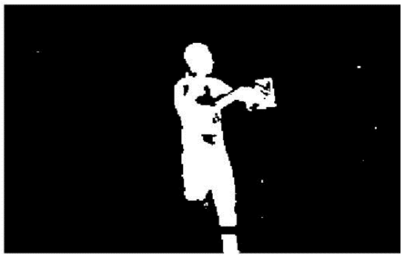


Figure 1.7: change detection using fuzzy c means and k means clustering for alpha 0.0

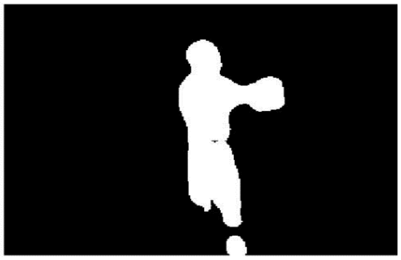


Figure 1.8: change detection using fuzzy c means and k means clustering for alpha 0.3

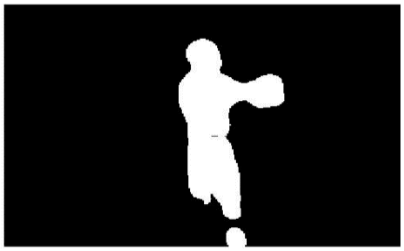


Figure 1.9: change detection using fuzzy c means and k means clustering for alpha 0.5

In the figure 1.1 to 1.9 we evaluated the performance of CDI-K method and proposed method. Figure 1.1 and 1.2 are two image at different time. Figure 1.3 is the ground truth image. Figure 1.4-1.6 shows the change detection using CDI-k method and Figure 1.7-1.9 shows the change detection using proposed method. It may be seen from the figure proposed method performance is better.

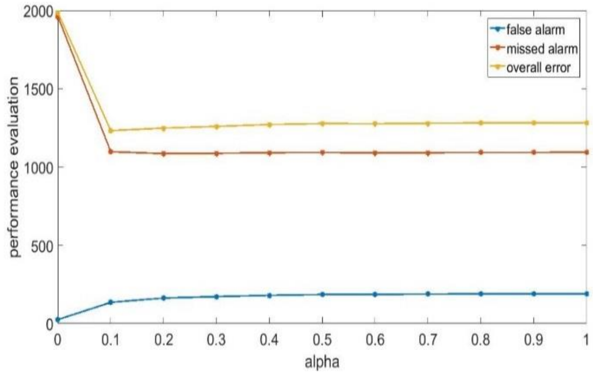


Figure 2.1: Performance Evaluation for CDI-k

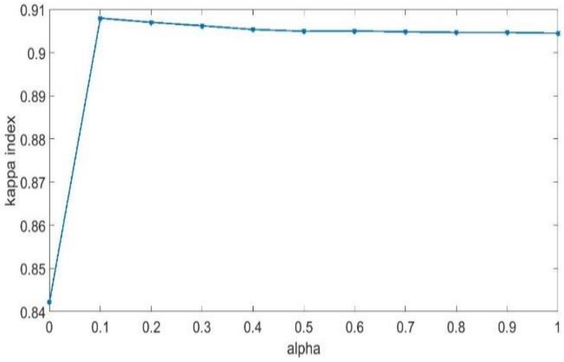


Figure 2.2: kappa value for CDI-k

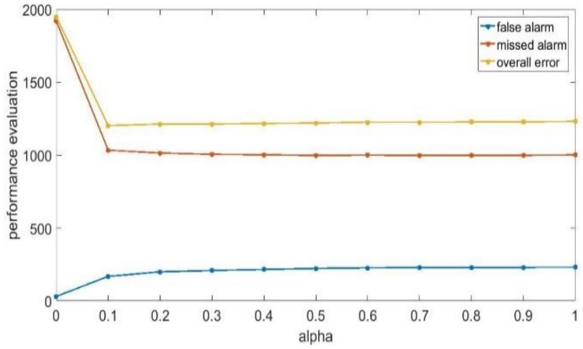


Figure 2.3: Performance Evaluation for Fuzzy C means and K means clustering (Proposed method)

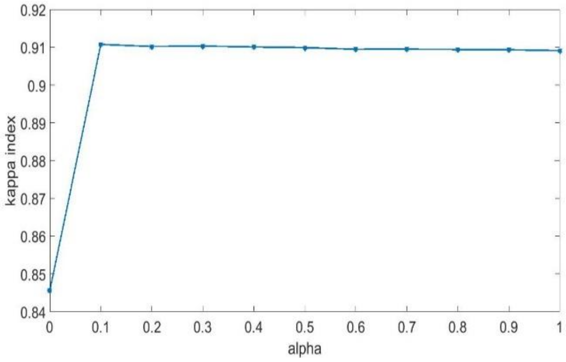


Figure 2.4: Kappa value for fuzzy C means and k means clustering (Proposed method)

The table show the values of regularization parameters. For desired performance of this method, less false alarm, miss alarm and overall error is required. In the other hand, Kappa index need to be close to 1 to produce a desired result for change detection. For CDI-K method, it is found from the table 1 that CDI- K method produce more missed alarm then proposed method. But in case of false alarm CDI-K produces less false alarm. But the overall error is better for proposed method. Figure 2.1 and 2.2 represents the curve for performance evaluation for CDI-k method. Where, Figure 2.3 and Figure 2.4 represents the curve for performance evaluation for proposed method. For proposed method, the regularization parameters are better for between 0.1 to 0.3 with respect to kappa, false alarm and missed alarm.

In, absence of ground truth, false alarm, missed alarm, overall error and kappa index calculation is not possible. As a result, change detection is produced only. In Figure 3.1 and Figure 3.2, Sharm-el-Sheikh, Egypt images are shown which are taken at 1998 and 2004. Then CDI-k and proposed method is applied to the images respectively.

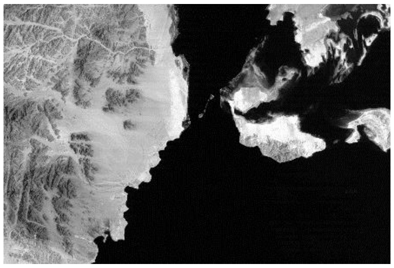


Figure 3.1: Sharm-el-Sheikh, Egypt image taken at 1998

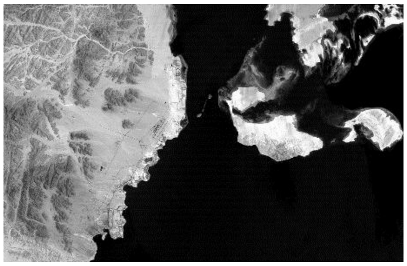


Figure 3.2: Sharm-el-Sheikh, Egypt image taken at 2004

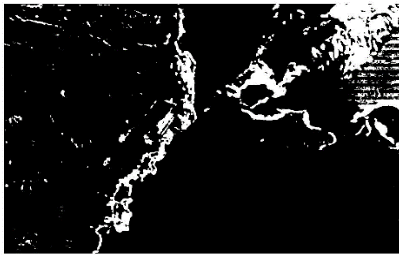


Figure 3.3: change detection using k means clustering for alpha 0.3

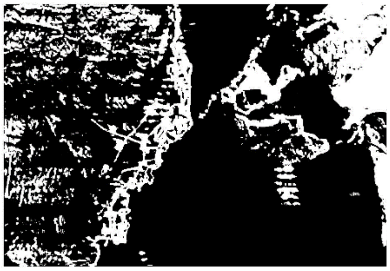


Figure 3.4: change detection using k means clustering (CDI-k) for alpha 0.5

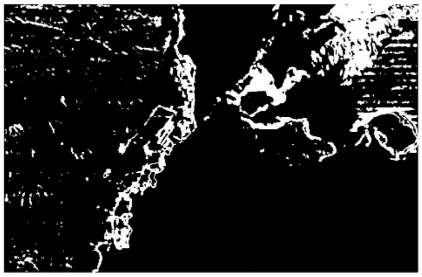


Figure 3.5: change detection using k means clustering (CDI-k) for alpha 0.8

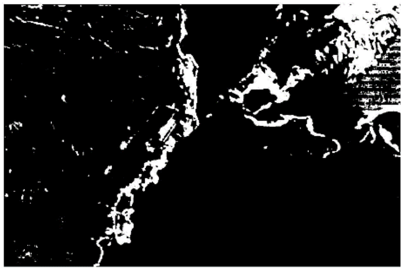


Figure 3.6: change detection using fuzzy c means and k means clustering for alpha 0.3

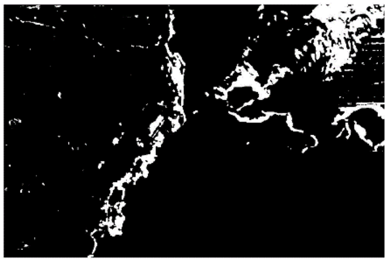


Figure 3.7: change detection using fuzzy c means and k means clustering for alpha 0.4

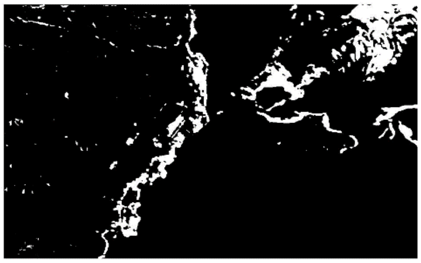


Figure 3.8: change detection using fuzzy c means and k means clustering for alpha 0.6

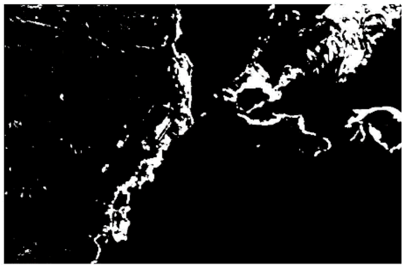


Figure 3.9: change detection using fuzzy c means and k means clustering for alpha 0.8

In Figure 3.3, 3.4 and 3.5 shows change detection using CDI-K method. In this Figures, more isolated pixel is detected as a change detection. Figure 3.6-3.9 is the change detection outputs due to proposed method. The performance using proposed method is better because of less isolated pixel. Comparing different output, for regularization parameter of 0.3 it produces desired change detection output.

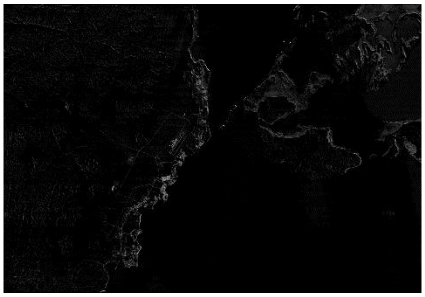


Figure 4.1: Difference image

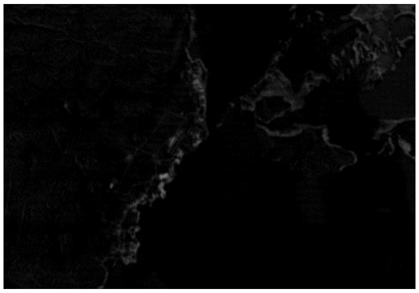


Figure 4.2: mean filtered image

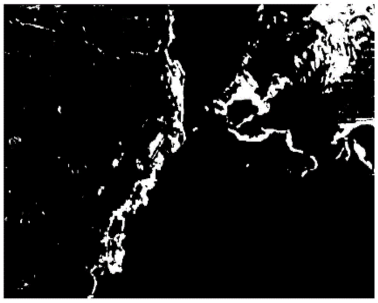


Figure 4.5: Change detection image using proposed method

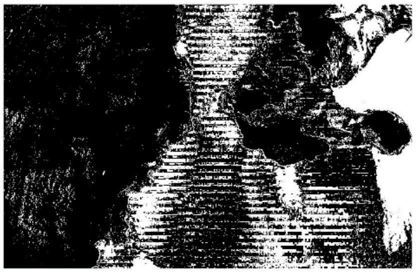


Figure 4.3: Log-ratio Image

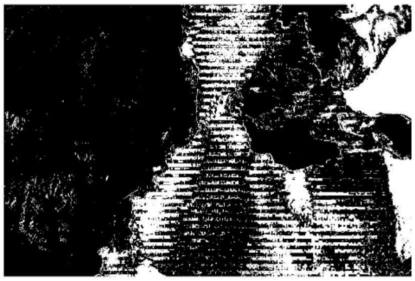


Figure 4.4: Median Filtered image

In figure 4.1-4.5 overall change detection procedure is shown. Difference image and log ratio image is determined from both of the time line image. Then mean filter and median filter is applied to both of this images. Mean filter shows the smooth region where median preserves the local edges of changes. Then, change detection is found from the combination framework of this filters in Figure 4.5.

In mean filter, if alpha value is selected as 0 then more number of missed alarms are produced because lot of change detection value is missed for detection. On the other hand, if alpha value is close to 1, it produces more false alarm. From the table, it is found that more missed alarm is occurred due to 0 regularization value and more false alarm is occurred due to 1 regularization value.

Conclusion:

In comparing both of this method, it is concluded that proposed method is better than CDI-k method based on change detection image, false alarm, missed alarm, overall error and kappa index. The methods worked better between regularization parameters 0.1 to 0.3. In future work, we should consider on image fusion method to find a better combination framework. For clustering, Gausssian mixture model should be considered for further research project. Speckle noise is not considered for this experiment because all of this test images did not have any kind of this noise. But SAR image may contain speckle noise which can be removed by probabilistic patch based filter. Due to absence of ground truth image, Sharm-el-Sheikh, Egypt images detection results performance is not evaluated.

Reference:

1. Yaoguo Zheng, Xiangrong Zhang, Biao Hou, and Ganchao Liu, “Using Combined Difference Image and k-Means Clustering for SAR Image Change Detection” IEEE geoscience and Remote Sensing Letters, Vol. 11, No. 3, March 2014
2. Nameirakpam Dhanachandra, Khumanthem Manglem and Yambem Jina Chanu “Image Segmentation using K-means Clustering Algorithm and Subtractive Clustering Algorithm” Eleventh International Multi-Conference on Information Processing-2015
3. Wikipedia Accessed December, 2016, https://en.wikipedia.org/wiki/Sensitivity_and_specificity
4. Wikipedia Accessed December, 2016, https://en.wikipedia.org/wiki/Cohen's_kappa
5. Accessed December, 2016, https://sites.google.com/site/dataclusteringalgorithms/k-means-clustering-algorithm
6. Wikipedia Accessed December, 2016, https://en.wikipedia.org/wiki/K-means_clustering

Appendix:

cdik_with_error_detection.m

```
clc;
close all;
clear all;
I1=imread('office1.jpg');
I2=imread('office000750.jpg');
i3=rgb2gray(I1);
i4=rgb2gray(I2);
i1=i3;
i2=i4;
%i1=i3(501:700,501:700);
%i2=i4(501:700,501:700);
[m,n]=size(i1);
ds=(i1-i2);
for i=1:m
    for j=1:n
        temp=abs(log(double(i2(i,j)+1))-log(double(i1(i,j)+1)));
        dl(i,j)=(temp);
    end
end
dlm = medfilt2(dl, [3 3]);
kernel = ones(11, 11) / (11*11);
J = conv2(ds, kernel, 'same');
J1=(J);
h = fspecial('average',[11 11]);
J2=conv2(ds,h,'same');
J3=uint8(J2);
%froundtruth
ground_trth=imread('officegt000750.png');
grnd_normalized=double(ground_trth/256);
r_max=10;
%initialization
a_true_positive=zeros(r_max,1);
a_true_negative=zeros(r_max,1);
a_false_positive=zeros(r_max,1);
a_false_negative=zeros(r_max,1);
hit_rate=zeros(r_max,1);
miss_rate_alarms=zeros(r_max,1);
false_alarms_rate=zeros(r_max,1);
overall_error=zeros(r_max,1);
k=zeros(r_max,1);
alpha3=zeros(11,1);
alpha3(1,1)=0;
for i=1:1:10
alpha2=0.1*i;
alpha3(i+1,1)=alpha2;
end
for index=1:1:11
alpha=alpha3(index,1);
D=(alpha*J1)+((1-alpha)*dlm);
D_re=reshape(D,m*n,1);
cdi=kmeans(D_re,2);
cdi2=reshape(cdi,m,n);
figure();imshow(cdi2,gray(2));
```

```
%floc = 'H:\final project2\output_pictures';
%saveas(gcf, fullfile(floc, num2str(alpha*10)), 'jpeg');

%figure;imshow(i1);
%figure;imshow(i2);
%cdi2=fcmkmns2;
[m,n]=size(cdi2);
%{
cdi=kmeans(D_re,2);
cdi2=reshape(cdi,m,n);
imshow(cdi2,gray(2));
figure;imshow(i1);
%}

true_positive=0;
true_negative=0;
false_positive=0;
false_negative=0;
for i=1:1:m
    for j=1:1:n

        t2=cdi2(i,j);
        t1=grnd_normalized(i,j);

        if t1==1 && t2==2
            true_positive= true_positive+1;
        elseif t1==0 && t2==1
            true_negative=true_negative+1;
        elseif t1 == 0 && t2==2
            false_positive=false_positive+1;
        elseif t1==1 && t2==1
            false_negative=false_negative+1;
        else
           disp('!!!error!!');
        end
    end
end

a_true_positive(index)=true_positive;
a_true_negative(index)=true_negative;
a_false_positive(index)=false_positive;
a_false_negative(index)=false_negative;
hit_rate(index)=true_positive*100/(true_positive+false_negative);
miss_rate_alarms(index)=false_negative*100/(false_negative+true_positive);
false_alarms_rate(index)=false_positive*100/(false_positive+true_positive);
overall_error(index)=false_negative+false_positive;

%cohen calculation
Po=(true_positive+true_negative)/(true_positive+true_negative+false_positive+
false_negative);
Aa=(true_positive+false_positive)*(true_positive+false_negative)/(true_positi
ve+true_negative+false_positive+false_negative);
Bb=(true_negative+false_positive)*(true_negative+false_negative)/(true_positi
ve+true_negative+false_positive+false_negative);
Pe=(Aa+Bb)/(true_positive+true_negative+false_positive+false_negative);
```

```
k(index)=(Po-Pe)/(1-Pe);

end

%plot(alpha3(1:11),a_false_positive(1:11));
figure();plot(alpha3(1:11),a_false_positive(1:11),alpha3(1:11),a_false_negati
ve(1:11),alpha3(1:11),overall_error(1:11));
ylabel('performance evaluation');xlabel('alpha');
legend('false alarm','missed alarm','overall error');
 figure();plot(alpha3(1:11),k(1:11));
ylabel('kappa index');xlabel('alpha');
```

cdik_sharmelshaikhdata.m

```
clc;
close all;
clear all;
I1=imread('SharmElSheik_1998.jpg');
I2=imread('SharmElSheik_2004.jpg');
i3=rgb2gray(I1);
i4=rgb2gray(I2);
B = imhistmatch(i3,i4);
i1=B;
i2=i4;
%i1=i3(1:800,1:800);
%i2=i4(1:800,1:800);
[m,n]=size(i1);
ds=abs(i1-i2);
for i=1:m
    for j=1:n
        temp=abs(log(double(i2(i,j)+1))-log(double(i1(i,j)+1)));
        dl(i,j)=uint8(temp);
    end
end
dlm = medfilt2(dl, [3 3]);
kernel = ones(11, 11) / (11*11);
J = conv2(ds, kernel, 'same');
J1=uint8(J);
h = fspecial('average',[11 11]);
J2=conv2(ds,h,'same');
J3=uint8(J2);
r_max=10;
%initialization
%{
a_true_positive=zeros(r_max,1);
a_true_negative=zeros(r_max,1);
a_false_positive=zeros(r_max,1);
a_false_negative=zeros(r_max,1);
hit_rate=zeros(r_max,1);
miss_rate_alarms=zeros(r_max,1);
false_alarms_rate=zeros(r_max,1);
overall_error=zeros(r_max,1);
k=zeros(r_max,1);
%}
alpha3=zeros(11,1);
```

```
alpha3(1,1)=0;
for i=1:1:10
alpha2=0.1*i;
alpha3(i+1,1)=alpha2;
end
for index=1:1:11
alpha=alpha3(index,1);
D=(alpha*J1)+((1-alpha)*dlm);
D_re=reshape(D,m*n,1);
cdi=kmeans(D_re,2);
cdi2=reshape(cdi,m,n);
figure();imshow(cdi2,gray(2));
%floc = 'H:\final project2\output_pictures';
%saveas(gcf, fullfile(floc, num2str(alpha*10)), 'jpeg');
end
```

fuzzycmenas_kmeans_with_errordetection.m

```
clc;
close all;
clear all;
I1=imread('office1.jpg');
I2=imread('office000750.jpg');
i3=rgb2gray(I1);
i4=rgb2gray(I2);
i1=i3;
i2=i4;
%i1=i3(501:700,501:700);
%i2=i4(501:700,501:700);

[m,n]=size(i1);
ds=(i1-i2);
for i=1:m
    for j=1:n
        temp=abs(log(double(i2(i,j)+1))-log(double(i1(i,j)+1)));
        dl(i,j)=(temp);
    end
end
dlm = medfilt2(dl, [3 3]);
kernel = ones(11, 11) / (11*11);
J = conv2(ds, kernel, 'same');
J1=(J);
h = fspecial('average',[11 11]);
J2=conv2(ds,h,'same');
J3=uint8(J2);
%froundtruth
ground_trth=imread('officegt000750.png');
grnd_normalized=double(ground_trth/256);
r_max=10;
%initialization
a_true_positive=zeros(r_max,1);
a_true_negative=zeros(r_max,1);
a_false_positive=zeros(r_max,1);
a_false_negative=zeros(r_max,1);
hit_rate=zeros(r_max,1);
miss_rate_alarms=zeros(r_max,1);
```

```
false_alarms_rate=zeros(r_max,1);
overall_error=zeros(r_max,1);
k=zeros(r_max,1);
alpha3=zeros(11,1);
alpha3(1,1)=0;
for i=1:1:10
alpha2=0.1*i;
alpha3(i+1,1)=alpha2;
end
for index=1:1:11
alpha=alpha3(index,1);
D=(alpha*J1)+((1-alpha)*dlm);
D_re=double(reshape(D,m*n,1));
[center,U,obj_fcn] = fcm(D_re,2);
center1=transpose(center);
U2=transpose(U);
U3=kmeans(U2,2);
fcmkmns2=reshape(U3,m,n);
figure();imshow(fcmkmns2,gray(2));
%floc = 'H:\final project2\output_pictures';
%saveas(gcf, fullfile(floc, num2str(alpha*10)), 'jpeg');
%figure;imshow(i1);
%figure;imshow(i2);
cdi2=fcmkmns2;
[m,n]=size(cdi2);
%{
cdi=kmeans(D_re,2);
cdi2=reshape(cdi,m,n);
imshow(cdi2,gray(2));
figure;imshow(i1);
%}
true_positive=0;
true_negative=0;
false_positive=0;
false_negative=0;
for i=1:1:m
    for j=1:1:n
        t2=cdi2(i,j);
        t1=grnd_normalized(i,j);
        if t1==1 && t2==2
            true_positive= true_positive+1;
        elseif t1==0 && t2==1
            true_negative=true_negative+1;
        elseif t1 == 0 && t2==2
            false_positive=false_positive+1;
        elseif t1==1 && t2==1
            false_negative=false_negative+1;
        else
           disp('!!!error!!');
        end
    end
end
a_true_positive(index)=true_positive;
a_true_negative(index)=true_negative;
a_false_positive(index)=false_positive;
a_false_negative(index)=false_negative;
hit_rate(index)=true_positive*100/(true_positive+false_negative);
```

```
miss_rate_alarms(index)=false_negative*100/(false_negative+true_positive);
false_alarms_rate(index)=false_positive*100/(false_positive+true_positive);
overall_error(index)=false_negative+false_positive;
%cohen calculation
Po=(true_positive+true_negative)/(true_positive+true_negative+false_positive+
false_negative);
Aa=(true_positive+false_positive)*(true_positive+false_negative)/(true_positi
ve+true_negative+false_positive+false_negative);
Bb=(true_negative+false_positive)*(true_negative+false_negative)/(true_positi
ve+true_negative+false_positive+false_negative);
Pe=(Aa+Bb)/(true_positive+true_negative+false_positive+false_negative);
k(index)=(Po-Pe)/(1-Pe);
end
%plot(alpha3(1:11),a_false_positive(1:11));
figure();plot(alpha3(1:11),a_false_positive(1:11),alpha3(1:11),a_false_negati
ve(1:11),alpha3(1:11),overall_error(1:11));
ylabel('performance evaluation');xlabel('alpha');
legend('false alarm','missed alarm','overall error');
figure();plot(alpha3(1:11),k(1:11));
ylabel('kappa index');xlabel('alpha');
```

fuzzycmeans_kmeans_sharmelshaikhdata.m

```
clc;
close all;
clear all;
I1=imread('SharmElSheik_1998.jpg');
I2=imread('SharmElSheik_2004.jpg');
i3=rgb2gray(I1);
i4=rgb2gray(I2);
B = imhistmatch(i3,i4);
i1=B;
i2=i4;
%i1=i3(1:800,1:800);
%i2=i4(1:800,1:800);

[m,n]=size(i1);
ds=abs(i1-i2);
for i=1:m
    for j=1:n
        temp=abs(log(double(i2(i,j)+1))-log(double(i1(i,j)+1)));
        dl(i,j)=uint8(temp);
    end
end

dlm = medfilt2(dl, [3 3]);

kernel = ones(11, 11) / (11*11);

J = conv2(ds, kernel, 'same');
J1=uint8(J);

h = fspecial('average',[11 11]);

J2=conv2(ds,h,'same');
J3=uint8(J2);
```

```
r_max=10;
%initialization
%{
a_true_positive=zeros(r_max,1);
a_true_negative=zeros(r_max,1);
a_false_positive=zeros(r_max,1);
a_false_negative=zeros(r_max,1);

hit_rate=zeros(r_max,1);
miss_rate_alarms=zeros(r_max,1);
false_alarms_rate=zeros(r_max,1);
overall_error=zeros(r_max,1);
k=zeros(r_max,1);
%}
alpha3=zeros(11,1);
alpha3(1,1)=0;
for i=1:1:10
alpha2=0.1*i;
alpha3(i+1,1)=alpha2;
end

for index=1:1:11

alpha=alpha3(index,1);
D=(alpha*J1)+((1-alpha)*dlm);
D_re=double(reshape(D,m*n,1));
[center,U,obj_fcn] = fcm(D_re,2);
center1=transpose(center);
U2=transpose(U);
U3=kmeans(U2,2);
fcmkmns2=reshape(U3,m,n);
figure();imshow(fcmkmns2,gray(2));
%floc = 'H:\final project2\output_pictures';
%saveas(gcf, fullfile(floc, num2str(alpha*10)), 'jpeg');
end
```