A NEW CHANGE DETECTION METHOD USING DOUBLE SEGMENTATION AND ITS APPLICATION ON REMOTELY SENSED IMAGES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

EKIN GEDIK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

SEPTEMBER 2013

Approval of the thesis:

A NEW CHANGE DETECTION METHOD USING DOUBLE SEGMENTATION AND ITS APPLICATION ON REMOTELY SENSED IMAGES

submitted by **EKIN GEDIK** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** _______________

Prof. Dr. Adnan Yazıcı
Head of Department, **Computer Engineering** _______________

Prof. Dr. Fatoş Tünay Yarman Vural
Supervisor, **Computer Engineering Department, METU** _______________

Examining Committee Members:

Prof. Dr. Göktürk Üçoluk
Computer Engineering Department, METU _______________

Prof. Dr. Fatoş Tünay Yarman Vural
Computer Engineering Department, METU _______________

Prof. Dr. Yasemin Yardımcı Çetin
Informatics Institute, METU _______________

Assoc. Prof. Dr. Ahmet Coşar
Computer Engineering Department, METU _______________

Dr. Onur Tolga Şehitoğlu
Computer Engineering Department, METU _______________

Date: _______________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: EKIN GEDIK

Signature :

ABSTRACT

A NEW CHANGE DETECTION METHOD USING DOUBLE SEGMENTATION AND ITS APPLICATION ON REMOTELY SENSED IMAGES

Gedik, Ekin
M.S., Department of Computer Engineering
Supervisor : Prof. Dr. Fatoş Tünay Yarman Vural

Eylül 2013 , 108 pages

Change detection research, a branch of statistical data analysis, focuses on detecting changed samples between different observations of the same dataset. The proposed study presents a novel change detection procedure and its application as a complete framework which is designed to work on remotely sensed images. The scope of the study is defined as detecting man-made change objects between satellite images of the same region, acquired at different times. Proposed framework has three main steps as preprocessing, feature extraction-classification and postprocessing. Preprocessing step normalizes, registers and measures the similarity of image pairs. The main contribution of the proposed study lies at the feature extraction and classification step. With the help of newly proposed "double segmentation" paradigm, an object based approach can be utilized without any prior information or supervision. Well known features in the change detection literature are defined, combined and compared in the study. Apart from known classification methods such as K-Means Clustering and Expectation-Maximization, a novel heuristic thresholding method is also presented. A postprocessing procedure which helps to obtain more accurate and visually appealing results is also provided. Experiments conducted on artificial and real satellite images show that proposed framework is good at capturing the man-made change object characteristics in remotely sensed images with high accuracy.

Keywords: Remote Sensing, Change Detection, Unsupervised Classification

ÖZ

ÇİFT BÖLÜTLEME İLE DEĞİŞİKLİK ANALİZİ VE UZAKTAN ALGILANAN İMGELERE UYGULANMASI

Gedik, Ekin
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. Fatoş Tünay Yarman Vural

Eylül 2013 , 108 sayfa

İstatiksel veri analizi çalışmalarının bir dalı olan değişiklik algılama, aynı verinin iki farklı gözleminde değişiklik gösteren örnekleri bulmayı amaçlar. Bu çalışmada, yeni bir değişiklik analizi yöntemi sunulmuş ve bu yöntemin uydu görüntüleri üzerindeki uygulaması sistematik bir yapı içerisinde incelenmiştir. Uygulamanın kapsamı, farklı zamanlarda çekilmiş aynı alanı gösteren uydu görüntülerindeki insan yapımı farklılıkları algılamak olarak belirlenmiştir. Önerilen prosedür, ön işleme, öznitelik çıkarımı-sınıflandırma ve son işleme olmak üzere üç ana adım içermektedir. Ön işleme adımında girdi imgeler normalize edilmekte, çakıştırılmakta ve imgelerin benzerlikleri ölçülmektedir. Önerilen yöntemin ana katkısı "çift bölütleme" olarak adlandırılan yöntemin öznitelik çıkarımında uygulanmasıdır. Bu yöntem sayesinde herhangi bir ön bilgi yahut gözetim olmadan nesne bazlı bir uygulama geliştirilebilmektedir. Literatürdeki öznitelikler çalışmada kullanılmış, birleştirilmiş ve karşılaştırılmıştır. Sınıflandırma için, K-Ortalama Kümeleme ve EM gibi metodların yanı sıra, yeni bir bulgusal eşikleme yöntemi sunulmuştur. Görsel ve sayısal olarak daha iyi sonuçlar alınmasını sağlayan bir son işleme yöntemi ayrıca sunulmuştur. Yapay ve gerçek uydu imgeleri üzerinde yapılan deneyler, yöntemin uydu görüntülerinde insan yapımı değişiklik nesnelerinin özelliklerinin yüksek kesinlik ile yakalanmasında başarılı olduğunu göstermiştir.

Anahtar Kelimeler: Uzaktan Algılama, Değişiklik Tespiti, Gözetimsiz Sınıflandırma

To my beloved mother...

ACKNOWLEDGMENTS

First of all, I would like to express my sincere gratitudes to my supervisor Prof. Dr. Fatoş Tünay Yarman Vural for her consistent support, patient guidance and enthusiastic encouragement throughout this work. Without her guidance and persistent help, this dissertation document would not have been possible. In addition to academic assistance, I am also grateful to her for inspiring me in all aspects of life.

I also would like to express my very great appreciation to Prof. Dr. Yasemin Yardımcı Çetin for introducing me to this research area. Her invaluable guidance, assistance and support is much appreciated.

My special thanks are extended to my colleagues, Dr. Ersin Karaman, Umut Çinar and Okan Bilge Özdemir. Their invaluable support and encouragement made this research much easier.

I also would like to thank the staff of ASELSAN and HAVELSAN for their help on implementation specific issues.

Special thanks should be given to my friends, who are always with me physically and mentally throughout my research.

Finally, I wish to thank my parents, Dilek Mariye Gedik and Emin Ahmet Gedik, who made me the person I am today, with their endless love, gratuitous support and infinite trust.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

SIFT	Scale Invariant Feature Transform
SURF	Speeded Up Robust Features
CVA	Change Vector Analysis
PCA	Principal Component Analysis
PCC	Post Classification Comparison
KT	Tasseled Cap
EM	Expectation Maximization
NN	Neural Networks
GIS	Geographic Information Systems
MRF	Markov Random Field
RPE	Reduced Parzen Estimation
MAP	Maximum a Posteriori Probability
IR-MAD	Iteratively Re-weighted Multivariate Alteration Detection
DMC	Direct Multidata Classification
OBIA	Object Based Image Analysis
CNT	Classical Non-Automatic Thresholding
SCV	Spectral Change Vector
SVDD	Support Vector Domain Description
SAR	Synthetic Aperture Radar
KI	Kittler-Illingworth
FCM	Fuzzy C-Means
GKC	Gustafson Kessel Clustering
GA	Genetic Algorithms
SA	Simulated Annealing
AICD	Aerial
PCNN	Pulse-Coupled Neural Networks
CXM	Multilayer Conditional Mixed Markov Model
MPP	Marked Point Processes
OCI	Object Correlation Images
NCI	Neighborhood Correlation Images
MLE	Maximum Likelihood Estimation
D	Intensity Difference Values (Feature)

R	Ratio Difference Values (Feature)
F	Range Filter Difference Values (Feature)
M	Saliency Map Difference Values (Feature)
DB	Davies-Bouldin Validity Index
CH	Calinski-Harabasz Validity Index
KL	Krzanowski-Lai Validity Index
Wint	Weighted Inter-Intra Ratio Validity Index
RANSAC	Random Sample Consensus

CHAPTER 1

INTRODUCTION

1.1 Motivation

Statistical data analysis is a widely studied topic in computer science literature that has strong connections with many other study areas. It is strongly utilized in many applications for achieving different purposes. Change analysis procedures can be defined as a branch of this wide research area. Instead of analyzing the samples obtained from a single observation to detect predefined patterns, this branch aims to compare two different observations to detect possible changes. Similar to its root paradigm, data analysis, change analysis is applicable to different data types and utilized for different application areas. This procedure can be generalized as detection of changed samples in two similar datasets. Usage of different data types creates application specific initialization, normalization, feature extraction and classification steps. It is nearly impossible to propose a complete change detection procedure applicable to all data types. However, such a definition becomes possible when the scope of the study is narrowed.

Image and video analysis is a widely studied branch of statistical data analysis. This research area is one of the fields that heavily utilize change detection procedures. Change detection procedures are applied on image pairs or sets for purposes such as disaster monitoring, surveillance, medical diagnosis, region-infrastructure planning and etc. [51]. Methods in the literature defined for image change analysis can be also grouped with regard to input image types. One of these sub-research areas is remote sensing, which focuses on images acquired by satellites. These images generally have different characteristics than normal images, therefore need specifically designed procedures. Satellite imagery is becoming an important part of our life with recent developments. The impact of such information on every day life is gradually growing with the increasing availability of such images. With the help of specifically designed applications, a person can use remotely sensed images for every day purposes, such as navigation. However, the academic study on remotely sensed images has a much longer past. Many algorithms have been developed for more than four decades to use this information in many productive ways. This huge knowledge also contains studies that focus on change analysis. Change analysis in the remote sensing area is widely used for military and civil applications and these studies are not finalized. Among many proposed change detection algorithms, a method that outperforms the others does not exist yet. Most algorithms are designed for specific application areas, hence do not produce competitive results on all data types. With the increasing resolution of satellite imagery, new research questions also started to emerge. A hot research topic is the object based change analysis in high resolution images. A change detection algorithm that combines the powerful aspects of the available methods into a complete framework is needed. The proposed change detection framework should include all preprocessing and postprocessing steps necessary to create a robust change detection result independent of the image type.

1.2 Scope and Goal

Proposed study intends to present a complete and robust change detection algorithm that focuses on man-made objects for remotely sensed images. Developed method should be invariant to spectral differences caused by illumination and type. Spatial differences between image pairs should be eliminated, in order to obtain a robust change result. Significantly different image pairs need to be detected and discarded. Most importantly, proposed algorithm should distinguish man-made objects such as buildings from other type of changes and provide a geometrically correct, rigid and visually appealing change mask. Change classification step should be tunable and answer different needs of the application domain. These needs may be high precision compared to relatively low recall or vice versa. The main purpose of this study is to provide all aforementioned steps with high success rates and combine them into a complete and robust change detection framework.

1.3 Contribution

The main contribution of the study is the newly proposed feature extraction and classification paradigm named as "Double Segmentation". This procedure is explained in detail in the following chapters. With this procedure, an object-based approach can be defined without making any prior assumptions about change objects and more rigid and visually appealing results can be created. Another contribution is the well defined preprocessing steps. With the help of image normalization, registration and similarity measurement metrics defined in the study, many different data types can be used as inputs to change analysis. In addition to preprocessing, different features are defined and combined for detecting changes more precisely. This way, contributions of well known features to the resulting change maps can be examined in detail. Three different methods for classification are presented, each fulfilling different needs. This way, users can select between these classification methods depending on their priorities such as higher precision or higher recall scores. Final contribution is the post-processing step that is built on the double segmentation procedure. With this step, some regions mislabeled as change can be eliminated from the change map to produce results with higher precision scores. A segment reconstruction procedure is also included in the post-processing. Using this procedure, visually appealing change maps that closely resemble change objects can be obtained. All these contributions are major components in creating a robust change detection procedure that is easily usable with different type of images, adjustable to user needs and most importantly, provide competitive and balanced results. When developing this framework, the existing literature is carefully surveyed, examined and discussed, making this dissertation document also a review of current literature.

1.4 Outline

This dissertation document that explains the developed change detection framework is divided into 5 chapters:

1. Introduction
2. An Overview For The Change Detection and Its Preprocessing Methods
3. Change Detection Framework with Double Segmentation
4. Experimental Analysis of the Double Segmentation Change Detection Framework
5. Conclusion

First chapter makes an entrance to the specified research area and defines the proposed study. Literature survey for image change analysis is presented in the second chapter. This chapter also includes explanations for the theoretic background of the utilized methods in the framework. Third chapter explains the steps of the proposed algorithm in detail. In this chapter, preprocessing, feature extraction, classification and postprocessing steps are introduced. Fourth chapter defines the experiments conducted on different datasets and discusses the obtained results. This chapter also includes comparisons of the suggested change detection method with the existing algorithms in the literature. Lastly, study is concluded in the fifth chapter by wrapping up the key themes and presenting the possible future work.

CHAPTER 2

AN OVERVIEW FOR THE CHANGE DETECTION AND ITS PREPROCESSING METHODS

The background of the study is explained in this chapter. In the first section, the literature on change detection is investigated. Second section focuses on preprocessing techniques such as image normalization and registration. Third section elaborates methods for measuring similarity between image pairs. Following section describes image segmentation. In the fifth section, commonly used features for change analysis is presented. Last section elaborates saliency analysis and its possible applications to change detection.

2.1 An Overview on the Change Detection Methods Employed in Change Analysis

With the introduction of devices capable of capturing high quality image sequences, studies on change analysis become more significant in computer vision literature. Despite different sensors, data types and applications, the motive of these studies are always same: to identify the changed regions in image pairs correctly. Research efforts in this topic have taken pace when satellite imagery made available for commercial and scientific usage in the eighties. For over three decades, change detection studies are being conducted in many areas of computer vision, including remote sensing, surveillance, land cover detection, disaster assessment, military usage, motion detection, object tracking, civil infrastructure and medical diagnosis. Radke et. al. reviewed change detection methods and applications in these different disciplines in a detailed survey which is published in 2004 [51]. Geometric correction and intensity adjustment methods are explained in preprocessing step. Detection methods are grouped in two general categories. Image rationing, thresholding and Change Vector Analysis (CVA) are covered in simple differencing category. Advanced methods such as significance and likelihood ratio tests, mixture and predictive models are examined in the second category. Application specific background subtraction and shading modeling are, also, covered. Even though the aim of all change detection applications are similar, the steps and methods used are specific to the application domain. Considering this fact, most of the existing work surveyed in this section are in the area of remote sensing. Change detection research in remote sensing area is started in the first quarter of seventies, with the initiation of LANDSAT program. One of the first studies conducted is published by Howarth and Wickwate in 1981 [35] and focuses on band rationing and post classification change detection methods. In 1989, current up to date methods are reviewed by Singh[55]. Thresholding, image differencing, regression and rationing, vegetation index differencing, Principal Component Analysis (PCA), post-classification comparison (PCC), CVA and background subtraction are given as the main procedures utilized. It is also stated that the performance of all techniques varies heavily upon the geometric registration of image pair, variations in atmospheric conditions and illumination and sensor calibration. A more recent study by Lu et. al. published in 2004 groups current methods in literature into 7 categories according to their theory and present advantages and disadvantages [42]. This study summarizes change detection research in three main steps:

1. Image Preprocessing (Geometrical Rectification and Image Registration, Radiometric and Atmospheric Correction)
2. Selection of a Suitable Classification Technique
3. Quality Assessment

The aforementioned categories of techniques are:

- **Algebra:** This category includes relatively simple methods which aim to determine a threshold to identify changes. Methods like image differencing, image regression, image rationing, vegetation index differencing, background subtraction and CVA fall into this category. Selecting suitable bands or indices and finding the convenient threshold for detection is the critical points for these group of methods [42].
- **Transformation:** Transformations that reduce data redundancy between spectral bands such as PCA, Tasseled Cap (KT), Gramm-Schmidt (GS) and Chi-square is examined under this category. Similar to methods under algebra category, these methods generally require a threshold for detection [42].
- **Classification:** Methods that work on previously classified images are included in this category. PCC, spectral-temporal combined analysis, expectation-maximization (EM), unsupervised classification based change detection and Neural Networks are examples in the category. The main drawback of such methods is the fact that quality of change detection is highly dependent on the first classification results [42].
- **Advanced Models:** Methods which convert reflectance values to physically based parameters or fractions are grouped under this category. Examples in this group are given as Li-Strahler reflectance model, spectral mixture models and biophysical parameter estimation models [42].
- **Geographical Information System Approaches:** This category includes techniques which utilize GIS applications to combine different sources of information. Similar

to classification methods, the results are highly sensitive to the previous information incorporated [42].

- **Visual Analysis:** Methods, which include human intervention, are defined as visual analysis. Instead of using computer processing, some applications consult human analysts. There exist such works in the literature but it is out of scope in current study. [42]

- **Others** This category encapsulated methods which cannot be attributed to one of the categories either because of limited data type or rare usage. Methods present in this category are generally too specific for wider usage and out of scope of the current study. [42]

Methods for change detection also evolve with the development of new satellites. New satellites can provide imagery with higher spatial and spectral resolution. This progress helps researchers to overcome some problems present in current methods, but creates new challenges. Another recent survey addresses this challenges by examining state of the art techniques presented in the last decade. [46]. Approaches are grouped into two categories as unsupervised and supervised. In addition to many aforementioned methods such as differencing, rationing, CVA, PCA and EM, unsupervised approaches in this survey also consists methods such as Markov Random Fields (MRF), Reduced Parzen Estimation (RPE), Maximum a Posteriori Probability decision (MAP) and Iteratively Re-weighted Multivariate Alteration Detection (IR-MAD). It is stated in this study that the main limitation of unsupervised methods is difficulties on identifying the type of change (apart from CVA). Radiometric, athmospheric and illumination corrections are also needed in unsupervised cases. Supervised methods are better at classifying change types and generally do not require pre-correction methods but require large amounts of training data. PCC, Direct Multidata Classification (DMC), NN and Support Vector Machines (SVM) are given as examples of widely used supervised methods.

Blaschke also examined a recent paradigm called Object Based Image Analysis (OBIA) for remote sensing and discussed change detection studies built on this paradigm [6]. Object based analysis deals with "objects" that is made of pixels, instead of general pixel-based or sub-pixel based attitudes. With the increasing spatial resolution provided by new satellites, it becomes possible to define objects in remote sensing context. Many studies shows that object-oriented paradigm enhance the quantitative analysis of traditional pixel-based methods.

This section explains the studies using aforementioned methods for change detection. Similar to [46], existing works are grouped into two categories as unsupervised and supervised ones.

2.1.1 Unsupervised Methods for Change Analysis

Two different automatic change detection methods are proposed by Bruzzone and Prieto [11]. Both methods work on difference image and aim to find a threshold for distinguishing changed regions. First method treats every pixel independent whereas second one considers the spatial-contextual information included in the neighborhood of each pixel using MRF. Both methods assume that prior knowledge of statistical distributions of change and unchanged pixels are known. In order to estimate these distributions, an iterative procedure based on EM is used. The conditional density functions of two classes are modeled as Gaussian distributions. After generating a probability map using EM, the probabilistic change map is thresholded to form a binary change map. First method uses Bayes rule for minimum error to find the threshold value which maximizes the posterior conditional probability for each pixel. Second method utilizes the presumption that a change pixel is likely to be surrounded by another change pixels. A second order spatial neighborhood system is used when forming the MRF model. Similar to the first method, the aim is to label pixels such that the posterior probability is maximized. This is achieved by minimizing the corresponding energy function. Results show that EM algorithm estimates the statistical distributions accurately. Second method enhances first one by dismissing possible noises.

Bruzzone and Prieto enhance their work [11] by utilizing RPE procedure [12]. Instead of directly modeling prior distributions using EM, non-parametric estimates of probability density functions are derived using RPE. Then, these estimates are improved iteratively using EM. The procedure does not presume any a-priori model on the input data distribution and adaptive to any type of data. A non-parametric model is converted to a more suitable semi-parametric model using EM, which describes difference image characteristics better. In the view of these information, the proposed method is named "adaptive semi-parametric". This method also considers spatial-contextual information included in the neighborhood using MRF. Obtained results are compared with classical non-automatic thresholding (CNT) approach and provided less noisy and more accurate results.

A change detection approach based on transformation and clustering is studied by Çelik [14]. Method starts by computing the difference image. Then, *hxh* non-overlapping blocks are generated from the difference image. Method continues with PCA to create eigenvector space on *hxh* non-overlapping blocks. A feature vector space is created by projecting the blocks around each pixel onto eigenvector space. This feature vector is clustered using k-means algorithm where k=2. These two clusters correspond to changed and unchanged regions. Then, the change map is generated by assigning each pixel to one of the clusters depending on minimum Euclidean distance between its feature vector and the mean feature vector of the clusters. Results obtained from a limited dataset are compared with EM and MRF based solutions presented in [11]. Provided visual and quantitative results are slightly better than approaches in [11].

Change Vector Analysis is a widely used approach in the literature of change detection. Bovolo and Bruzzone enhanced CVA approach [44] by proposing a framework in polar domain [8]. It is stated that, even though change vectors possess information regarding the direction of change, most applications only exploit the magnitude, resulting in a suboptimal analysis. Proposed framework is developed to address this issue. Unchanged pixels are expected to have a magnitude close to zero whereas changed pixels tend to have a magnitude far from zero.

Based on this assumption, unchanged pixels are defined as a circle and changed pixels are defined as an annulus in the polar domain. Since it is harder to label pixels closer to the decision threshold, another annulus is created for representing uncertain pixels in the framework. Then the statistical models suitable for representing class distributions are explained. In order to obtain a more effective technique, simplifying assumptions are made in this step which resulted in different conditional distributions for changed and unchanged pixels. In order to represent unchanged pixel magnitudes Rayleigh distribution and for its direction Uniform distribution is used. Changed pixels are represented by Rice and Non-Uniform distributions for magnitude and direction, respectively. The experiments show that the unchanged pixels tend to gather around origin as a cluster whereas changed ones are far from origin. Quantitative results show that using the aforementioned distributions for representing change and unchanged classes yields much better results.

Semi-supervised methods are also suggested in the literature. In an article by Bovolo et al. [9], a semi-supervised approach which uses change vectors to derive a pseudo-training set is used with a classifier. The proposed approach starts with calculating change vectors using CVA. Using Bayesian decision theory, the initial distributions of changed and unchanged pixels are estimated and a decision threshold is found. Contrary to former studies [11, 12], this threshold is not directly used to generate a change map. Pixels with a magnitude closer to the threshold value exhibits uncertainty. In order to overcome this limitation, an uncertainty zone is defined around the threshold value. Using the magnitudes fetched from the certain zone, a pseudo-training set is formed. Then, a Semi-supervised Support Vector Machine, S^3VM [5], is initialized using this pseudo-training set. By regarding unlabeled patterns in the uncertainty zone, S^3VM performs change detection in original feature space and the final decision boundary is determined. When initializing S^3VM, model selection is required. Since no labeled test data exists for such selection, a procedure for model selection is also proposed. It is assumed that different sets of parameters could result in proper change maps. Among all parameter set values, ones that acquire a higher accuracy on the pseudo-training set than a predefined threshold are chosen. For every chosen solution the ratio of detected changed and unchanged pixels are compared with the same quantity computed on the pseudo training set. The solution is discarded if the value differs significantly. Next, a similarity measure is defined between existing solutions. The solution with the highest similarity measure is chosen as the correct parameter set for initilization. The proposed approach generates better accuracies than the CVA+EM method [12], but requires much more computational time. A similar version of this approach [9] is proposed by Bovolo,Vals & Bruzzone in 2010 [10]. This method uses support vector domain description (SVDD) one-class classifier instead of a S^3VM and produces similar results.

Some studies in literature focus on a specific data type. In such study [2], a change detection method for synthetic aperture radar (SAR) images is proposed. Algorithm starts with a preprocessing step that aims to minimize speckle noise present in SAR images. It continues with image rationing to create a ratio image which is represented in logarithmic scale. The ratio image is then analyzed to create a change map. This step is handled with automatic thresholding using Kittler-Illingworth (KI) threshold selection criterion which is modified to suit the problem domain. The optimal number of iterations for despeckling filter is also determined automatically using KI. When modeling the class-conditional distributions for obtaining a threshold, Gaussian and Generalized Gaussian distributions are used. Generalized Gaussian assumption performed far better in the experiments. It is shown that algorithms developed for different data types tend to differ by preprocessing and modeling steps.

Different thresholding methods for change detection are also analyzed in the literature. In a study suggested by Rosin, 4 different methods are described and compared [52]. According to the study, for determining an optimal threshold, signal, noise, intensity and spatial properties can be modeled. For modeling image noise, using zero mean Normal distribution is recommended. If the edge maps are differenced instead of intensity images, Rayleigh distribution could be used, which can be approximated by a Normal distribution. In order to estimate noise, least median squares of difference image histogram is used. Spatial distribution of the noise is expected to be random which could be modeled by Poisson distribution. In order to model intensity distribution of the signal, a surrounding window approach is proposed. Non parametric methods such as Kolmogorov-Smirnov and Cramer-von Mises test are used for comparing two cumulative distributions of the intensities in image pairs. It is expected that changed regions will remain stable over a range of threshold values instead of a noise. This property is used to model the spatial distribution of the signal using the Euler number. Experiments performed on normal camera images showed that spatial distribution assumptions provide the most reliable and accurate results.

Clustering algorithms are also used in change detection. Fuzzy clustering algorithms are utilized in the study of Ghosh, Mishra & Ghosh [29], where they employ fuzzy c-means (FCM) and Gustafson-Kessel Clustering (GKC) methods for detecting changes. In order to avoid the converge to the local minima, the clustering algorithms are combined with Genetic Algorithms (GA) and Simulated Annealing (SA). The suggested approach starts with computing the difference image using CVA. Then, two dimensional feature vectors for each pixel are created. The first feature is the intensity of the difference pixel and the second feature is computed as the average intensity of the 8 neighboring pixels. Using the feature vector, image is clustered into two components, namely changed and unchanged clusters. Different clustering algorithms like hard c-means clustering, FCM, GKC, FCM+GA and GKC+SA are used and their performances are evaluated. In order to validate the results, Xie-Beni index is used. GKC enhanced with simulated annealing tends to outperform other approaches in experiments since it can extract clusters with different shapes. Suggested method also outperforms EM+MRF based methods in both accuracy and time complexity.

Genetic Algorithms are also utilized for change detection. Çelik used a genetic algorithm approach for change detection in his article dated 2010 [15]. It is stated that most of the existing unsupervised methods either require a parameter optimization step or a priori assumptions to model the difference image data which make them undesirable when dealing with different type of sensors. Proposed method does not require any parameter optimization or prior knowledge. Method starts with computing the difference image. For SAR images, log-ratio of image pairs are preferable. Change detection mask is computed by partitioning the difference image into two clusters, changed and unchanged. Partition is done by a genetic algorithm. For each region, Mean Square Error (MSE) between its difference image values and the average of its difference values are calculated. The weighted sum of the MSE of the changed and unchanged regions is used as a cost value and GA founds the change mask that minimizes that cost function. Experiments are conducted on both optical and SAR images. The proposed algorithm outperforms other approaches such as EM, MRF, PCA and multiscale based ones.

Another study uses Pulse-Coupled Neural Networks (PCNN) for change detection in very high resolution images [47]. PCNN are defined as a relatively new technique which is an unsupervised and context sensitive version of the Multi-Layer Perceptrons. Its application to change detection is done by comparing the PCNN outputs of image pairs by measuring their similarities. An average correlation value for each pixel is computed using the PCNN signal values. When the PCNN output is plotted, signals of unchanged regions are highly correlated both on the waveform and time dependence. If the input area is changed, these properties tend to get rather different and correlation value decreases. In the experiments, where a very high resolution panchromatic image pair is used, 100x100 windows with 50 pixel overlap are used and mean correlation value is used to discriminate changed patches from unchanged ones. No automatic thresholding method is proposed, since in the experiments changed regions tend to have mean correlation value close to zero and unchanged ones close to 1. Proposed algorithm achieves high accuracy scores according to the quantitative results presented.

2.1.2 Supervised Methods for Change Analysis

Neural Networks are widely used in change detection research. One of these studies that utilize Artificial Neural Networks for multitemporal change analysis is published in 1999 [20]. This study differs from previous studies, since it does not only detect changes but also classifies them. Using a supervised method allows to define the nature of changes. This study names such applications as categorical change extraction and distinguish them from traditional change mask development. The constructed Neural Network land cover change detection system takes image pairs as input and outputs change classes via direct encoding or binary encoding depending on the number of change classes. The network is trained using backpropagation algorithm. In the study, a four-layer network is utilized to analyze large change combinations. In the classification part, 4 classes are defined as forest, agriculture/bare/urban, cypress/wet deciduous scrub/marsh and water resulting in 16 possible change transitions. Using a training set, the network weights are adjusted during the backpropagation procedure. Proposed approach produces high accuracy values. Results are compared with post-classification protocol and proposed approach outperformed it. For post classification, both images are classified using maximum likelihood beforehand using the same training data.

Optical flow is also applicable to change detection problem when an image sequence is present. Bourdis, Marraud & Sahbi proposed an optical flow based approach for detecting changes in aerial images [7]. The proposed method is remarked as resilient to viewpoint differences and parallax effects that could be encountered because of camera motion. Method starts with registering image pairs using Speeded Up Robust Features (SURF) and Random Sample Consensus (RANSAC). In order to estimate residual parallax vectors, optical flow is used. Parallax vectors can be sources of false alarms. Hence, they are restricted to be collinear to the epipole direction. Changed pixels are found using a likelihood ratio test that aims to minimize optical flow matching error. The null and alternative hypothesizes are generated using the established ground truth. Experiments are conducted on the provided artificial dataset named Aerial Imagery Change Detection (AICD). Results shows that constraining the parallax vectors results in better performance.

Markov models other than MRF are also used in change detection methods. Benedek and Sziranyi utilized a multilayer conditional mixed Markov (CXM) model for this purpose [4]. CXM model is defined as a combination of mixed Markov model and a conditionally independent random field of signals. CXM can integrate global intensity statistics and local correlation to provide locally correct and a smooth change map. Algorithm starts by computing 4 different change maps each based on different features and constraints. When performing the classification, changed and unchanged pixels are assumed to be generated by random processes with different distributions and MLE method is used with training data. The CXM model is then initialized with these four different segmentation masks produced by global intensity statistics, local correlation, contrast-based feature selection and per pixel integration of former three masks. By stochastically optimizing an energy function which encapsulates the spatial smoothness, optimal local features and observation-consistent classification constraints, CXM model creates the final change mask. When minimizing the energy function, iterative Modified Metropolis relaxation algorithm is used. Proposed algorithm is compared with PCA, Hopfield type NN, RPE+MRF and MLP+MRF methods and outperformed them in overall error percentage. Visual results are also more compact and relatively less noisy.

Some studies on the literature focus on classification and change detection of objects, instead of the traditional pixel based methods. The objects in question could be pixels grouped with regard to their spectral and/or spatial properties with segmentation paradigms or they could be specifically extracted by classification methods. One example study conducted in 2009 [3] aims to classify the buildings in the scene and performs change detection. This research aims to merge the advantages of both low level pixel and high level object based approaches by utilizing Marked Point Processes (MPP). The output of the proposed approach consists of both the size, orientation and position of each building and flags such as newly built, modified, demolished and unchanged. Low level features for building identification, similarity measurement and object based features are defined for building detection. These features include local gradient orientation density, roof color filtering, shadow, roof homogeneity and more. Change detection part is handled by transforming the task into an energy minimization problem. This problem is realized within the MPP framework. In this framework, whole population is characterized instead of a single object which gives the power of exploiting entity interactions. The relational model between the entities is characterized by the neighborhood relation which is valid if the bounding boxes of the objects intersect. Building energies are calculated regarding soft constraints and change classes are assigned to each entity. Proposed method is compared to PCC approach and qualitative and quantitative results are shown to be better.

Another object based research is conducted in 2008 [36]. This study facilitates object/neighborhood correlation image analysis and image segmentation. Correlation analysis is based on the assumption that if there is no change in a region, intensity values of the image pairs should be highly correlated and vice versa. Five different change detection approaches are compared. These approaches are object-based change classification that combine object correlation images (OCI), object-based classification that combine neighborhood correlation images (NCI), object-based classification without contextual features, per pixel classification incorporating NCIs and traditional per pixel classification. For the decision part, decision trees and Nearest Neighbor methods are used. OCIs and NCIs are extracted from the input images and consist three features; correlation, slope and intercept. The difference is that OCIs are computed on the object level whereas NCIs are computed on per pixel basis. Objects are defined as segments with spectral and spatial homogeneity and extracted using a commercial software. 8 different classes are defined five being unchanged and 3 being changed. According to experiments, object based classification outperforms pixel based classification in all cases. Two different decision algorithms performs similar and both OCI and NCI provide useful change information and should be considered.

Studies that facilitate GIS software and databases are also frequently used in the current literature. Walter proposed an object-based classification method for change detection which utilizes GIS data in 2004 [58]. The object based approach derives objects from an existing GIS database and combines the pixels representing the object. The classification step uses Maximum Likelihood and the combined pixels are classified together, instead of classifying them together. Then, it compares the results with the present GIS data to detect changed objects. Five different classes are defined as water, forest, settlement, greenland and roads. Different features are defined for classification. The first feature is the mean gray value of each channel, vegetation index and texture for all objects. Mean variance of the all input values are also taken as a feature. For analyzing the texture of objects, a texture operator which measures the contrast by co-occurrence matrix is chosen. Normalised Difference Vegetation Index (NDVI) is selected as the vegetation index. Apart from mentioned input channels that are generally used in pixel based classification, land cover classification percentage of the pixels in an object is taken as a feature. This percentage is calculated by classifying the image per pixel basis and counting the different classes present in an object. Proposed approach is tested on two test areas which include many pre-classified objects. Proposed method produces acceptable results. The main flaw of this approach is explained as the possibility of missed changes in an object with a large area. In order to overcome this flaw, integration of a pixel based classification to the current method is proposed.

2.2 Preprocessing

Preprocessing is a very crucial step for change detection applications. In order to obtain a correct change map, image pairs should show similar spectral characteristics in unchanged regions. Differences in spectral characteristics can be caused by different factors such as, different types of satellites, atmospheric effects, acquisition time and seasonal variations. Besides that, two images should be co-registered to develop an efficient change analysis method. Not all change analysis methods make comparison on pixel basis, so pixel by pixel correlation is not always necessary, but it becomes impossible to compare two images when the spatial difference gets larger. Also, global and sudden local illumination differences are widely encountered which can emerge as false alarms. In this section, preprocessing methods are discussed. Firstly, radiometric and atmospheric correction methods are inspected and an easy way to correlate spectral characteristics of two images is presented. Then, image registration is explained and the methods used in this study for registration is introduced. In the last part, methods for eliminating illumination differences is discussed.

2.2.1 Image Normalization and Illumination Invariance

Even though the input image pair represents the same area which is spatially equivalent, the spectral characteristics of two images may differ significantly. These variations may be due to the different factors such as radiometric distortions, atmospheric effects, local and global illumination changes. Radiometric and atmospheric differences can be caused by different factors such as scattering, absorbance and refracting of the light by the satellite, solar irradiance and zenith angles. These effects should be eliminated before using the images for any purpose. Methods for such corrections are present in the literature and widely used in the remote sensing community [16]. However, they are not related to the context of the current study. In this study, it is assumed that the both images are radiometrically corrected and atmospheric effects are eliminated. Even though such effects are minimized, illumination changes greatly effect the reflectance values of the image. Also, different satellites provide different wavelengths for colors, again resulting in images with different spectral characteristics. Illumination variance is not specific to remote sensing. It is a widely studied topic in whole computer vision area. It is nearly impossible to achieve a good change detection result with different intensity characteristics for an image pair when the intensity values are main features in classification. A preprocessing step that eliminates illumination effects and global intensity differences should be introduced. Image normalization, being one of the most basic yet effective approaches, normalizes the pixel intensity values to have the same mean and variance as the other images in the database [51]:

$$I_2(x) = \frac{\sigma_1}{\sigma_2} I_2(x) - \mu_2 + \mu_1. \tag{2.1}$$

Normalization can be also localized by considering blocks instead of the whole image. Also, rather than transforming one images statistical properties to other, both images can be normalized to have zero mean and variance.

Homomorphic filtering is also used for image normalization, especially in computer vision application where Lambertian surfaces are present[51]. Each pixels intensity value could be divided in two components, $I_i(x)$, illumination from the light source and $I_o(x)$, reflectance of the corresponding object. Since the illumination is assumed to be constant across the image, the only part relevant to change is the reflectance value. Then, the *shading model* can be formulated as [50]:

$$I(x) = I_i(x)I_o(x). \tag{2.2}$$

By taking the logarithm of (2.2) and applying a high-pass filter $F(.)$, the reflectance part can be estimated as:

$$I_o(x) = \exp\{F(lnI(x))\}. \tag{2.3}$$

Instead of using the raw intensity values, reflectance part could be used in change detection application. There also exist methods that explicitly define the illumination model in the scene but they are not completely applicable to remote sensing area.

Linear transformations are, also, used both for preprocessing and classifying the image content. PCA can be applied to all samples (both bands and images) or the difference image to detect changes. In the first case, principal component images that correspond to low valued eigenvalues are expected to show changed regions. In contrast, first few principal components indicate changed areas [51]. Rather than directly using transformations for classification, they can be used to create more robust features for further steps. Tasseled Cap Transformation proposed for LANDSAT images in 1976 [39], transforms the raw bands into ones with semantic meanings such as "soil brightness", "green vegetation", etc. Another study produced more meaningful results by using $(R/(R+G+B), G/(R+G+B), B/(R+G+B)$ values instead of RGB [24].

2.2.2 Image Registration

Image registration can be defined as the process of matching pixels in image pairs, obtained at different times, by different sensors and/or with viewing angles and positions [60, 23]. Registration of images obtained by similar sensors is named unimodal registration whereas registration of images acquired by similar sensors is called multimodal registration [34]. Registration methods can be put in two distinct categories. Direct correlation based methods evaluate the correlation between two images and maximize it. Other methods rely on extracting distinctive features from image pairs and matching them. Feature extraction based methods are generally more robust and faster, therefore more widespread. One way to find potential feature points is the utilization of Harris corner detector or its variants [32]. More recent feature extraction methods like Scale Invariant Feature Transform (SIFT) [41] and Speeded Up Robust Features [1] are also getting widespread. For feature matching, cross-correlation or normalized sum-of-squared differences can be used [25]. Another approach to feature matching is extracting a feature and finding its pair on the second image by gradient based optical flow methods, instead of extracting features independently [45]. Robustness problem can be minimized by utilizing an outlier detection algorithm [54] like RANSAC [26]. After matching feature points, one image is taken as the reference and the other one is transformed using homography [34].

This study assumes the input image pair is co-registered and the algorithm is developed on this assumption. However, an optional registration step is also proposed, since it is hard to find co-registered image sets. Proposed registration procedure starts with extracting features from image pairs using SIFT, detects and eliminates outliers with RANSAC, creates a 2D homography matrix and transforms the image using it. In the following subsections, SIFT and RANSAC are explained in detail.

2.2.2.1 Scale Invariant Feature Transform (SIFT)

SIFT is a feature extraction technique presented by Lowe in 2004 [41]. Extracted features are invariant to scale and rotation. SIFT is proven to be handy when there exists viewpoint differences, noise, illumination changes and affine distortions. SIFT algorithm has 4 main steps:

1. **Scale-space Extrema Detection** In this step, whole image is searched for possible features in different scales. Only possible scale space kernel, Gaussian, is used. Scale space of an image which is produced from the convolution of a variable-scale Gaussian, $G(x, y, \sigma)$ defined with the input image $I(x, y)$ is represented by the function:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y), \tag{2.4}$$

   where $*$ is the convolution operator and G is

$$(G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \tag{2.5}$$

   Difference of two scale space functions are then found as

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma), \tag{2.6}$$

   where k is a constant multiplicative factor. It is stated that Difference-of-Gaussian function is a close approximation to the scale-normalized Laplacian of the Gaussian. Local maxima and minima of $D(x, y, \sigma)$ are detected by comparing every sample point to its eight neighbors in the current image and nine neighbours in the scale below and above. Sample point is taken as a candidate if its value is larger or smaller than all of them. Frequencies of sampling in scale and spatial domain are found experimentally.

2. **Keypoint Localization** After a candidate keypoint is found, it is checked for two possible conditions that can generate false alarms which are low contrast and edge responses. In order to detect and reject keypoints with low contrast, shifted version of Taylor expansion of the scale-space function is used where the origin is at the sample point. If

the function value at the extremum is smaller than a predefined threshold, sample point is rejected. Taylor expansion is approximated up to the second term as

$$D(\mathbf{x}) = D + \frac{\partial D^{\mathbf{T}}}{\partial \mathbf{x}}\mathbf{x} + \frac{1}{2}\mathbf{x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x}, \tag{2.7}$$

where $\mathbf{x} = (x, y, \sigma)^T$ is the offset from the sample point. Extremum is found by taking the derivative of this function and setting it to zero:

$$\hat{\mathbf{x}} = -(\frac{\partial^2 D}{\partial \mathbf{x}^2})^{-1}\frac{\partial D}{\partial \mathbf{x}}. \tag{2.8}$$

In order to obtain the function value at maximum, equation (2.5) is substituted in to equation (2.4), yielding: s

$$\hat{\mathbf{x}} = D + \frac{1}{2}\frac{\partial D^{\mathbf{T}}}{\partial \mathbf{x}}\hat{\mathbf{x}}. \tag{2.9}$$

The minimum extremum value for a decent keypoint is chosen as 0.03 in the study. In order to eliminate false keypoints caused by edge responses, ratio of eigenvalues of the Hessian matrix is used. A poor keypoint caused by edge response is exptected to have a strong response at the edge direction but the response in perpendicular direction should be small. The principal curvatures of the point is computed from a $2x2$ Hessian matrix, $\mathbf{H}$:

$$\mathbf{H} = \begin{Bmatrix} D_{xx} & D_{xy} \\ D_{yx} & D_{yy}. \end{Bmatrix} \tag{2.10}$$

Rather than directly computing the eigenvalues of this vector, their ratios are found using the Trace and Determinant of this matrix:

$$Tr(\mathbf{H}) = D_{xx} + D_{yy} = \alpha + \beta, \tag{2.11}$$

$$Det(\mathbf{H}) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta \tag{2.12}$$

where α is the largest and β is the smallest eigenvalues. Let $\alpha = r\beta$. In order to check the ratio of principal curvatures, the following formula can be used:

$$\frac{Tr(\mathbf{H}^2)}{Det(\mathbf{H})} < \frac{(r+1)^2}{r} \tag{2.13}$$

The study suggest $r = 10$ to detect edge responses correctly.

3. **Orientation Assignment** After eliminating false keypoints, each keypoint is assigned an orientation. Using pixel differences, each keypoints gradient magnitude $m(x, y)$ and orientation $\theta(x, y)$ at the corresponding scale are computed as:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \tag{2.14}$$

$$\theta(x, y) = \tan^{-1}((L(x, y+1) - L(x, y-1))/L(x+1, y) - L(x-1, y)) \tag{2.15}$$

An orientation histogram with 36 bins is then formed using the sample points around the candidate keypoint and the highest peak in the histogram is selected as the dominant orientation.

4. **Keypoint Descriptor** With the former 3 steps, position, scale and orientation are found for each keypoint. At this step, keypoint descriptor is created using this information. Keypoint descriptor construction is handled in 4 steps:

    (a) Orientations and gradients around the keypoints are sampled. Coordinates and orientation of the descriptor is rotated relative to keypoint orientation.

    (b) Using a Gaussian weighing function, magnitude of each sample point is assigned.

    (c) Samples are accumulated into orientation histograms.

    (d) Descriptor is strengthened against illumination and sudden gradient changes by normalizing to unit length.

After the keypoint descriptors are computed for image pairs, a procedure for matching them is needed. A possible matching is accomplished between the candidate keypoint and its nearest neighbor. Nearest neighbor is defined as the keypoint with smallest distance. As a distance measure, Euclidean is chosen in the aforementioned study. Authors uses Best-Bin-First algorithm for effectively finding the nearest neighbor (match) of a keypoint. For matching the occluded objects, that have many more outlier matchings than inliers, usage of Hough transform is suggested [41].

2.2.2.2 Random Sample Consensus

Even though SIFT is quite robust at extracting and matching features, experiments shows that there can still be outliers. These outliers can ruin a registration process and must be eliminated. In this study, outlier detection algorithm RANSAC [26] is chosen for this purpose. RANSAC is introduced in 1981 and still widely used for detecting outliers in image analysis. Instead of using the whole data to obtain a good fit to detect outliers, RANSAC starts with an initial set and adds consistent sample points into the initial set. When there are sufficient points, RANSAC initiates a smoothing algorithm to find the parameters of the distribution. Three parameters are unspecified in RANSAC algorithm, that can be changed according to application needs which are the error tolerance, number of subsets for trial and number of sufficient points for a good estimation, t. The steps of RANSAC can be summarized as follows:

1. Randomly select a subset S of n data points from P to instantiate the model M, where n is the minimum number of data points for the model and P is the whole set of data points which satisfies $(P) \geq n$

2. Determine the consensus set S^* by selecting points from P that are inside the error tolerance of M.

3. Use S^* to create a new model M^* if $\#(S^*)$ is greater than t

4. Randomly select a new subset and repeat the process if S^* is less than t.

5. If no consensus set with t or more points is found after a predefined number of iterations, solve the model or return failure.

Some useful suggestions about error tolerance, threshold and number of subsets are given in the article. Error tolerance can be set by perturbing the data and calculating sample deviation. Two different solutions for determining the maximum number of trials are given. When the probability of any selected data point being inside the error tolerance region is defined as w and $b = w^n$, expected value of the number of trials k is found as:

$$E(k) = 1/b = w^{-n}. \tag{2.16}$$

If one wants to ensure with probability z that at least one random selection is error free, k can be found with the equation:

$$k = [log(1 - z)/log(1 - b)]. \tag{2.17}$$

2.3 Metrics for Measuring Similarity

This study focuses on detecting small changes between image pairs like construction or destruction of man made objects. Two significantly changed or completely different images should not be given as an input and the proposed method should detect such cases. Therefore, a method that can robustly measure the similarity between images is needed. There exist different similarity and dissimilarity measures in the literature for comparing two datasets. Some of these approaches are applicable to a large variety of data and some of them are specifically designed for comparing images. S can only be called a similarity metric if it satisfies following 4 conditions where S_0 is the largest value that it can produce [56]:

1. $S(x, y) \leq S_0$ for an arbitrarily large number S_0
2. $S(x, y) = S_0$ only if x=y
3. $S(x, y) = S(y, x)$
4. $S(x, y)S(y, z) \leq [Z(x, y)S(y, z)]S(x, z)$

Similarly, for a dissimilarity metric D, following conditions must hold:

1. $D(x, y) \geq 0$
2. $D(x, y) = 0$ only if x=y
3. $D(x, y) = D(y, x)$
4. $D(x, y) + D(y, z) \geq D(x, z)$

A basic approach to the problem is computing the Euclidean distance for every pixel pair in I_1 and I_2 and using the result as a dissimilarity metric. However, this method is very vulnerable to possible spectral norms and spatial changes and do not produce robust results in many cases. A similar method is mean squared error, which can be computed by averaging the squared differences between intensity values of corresponding pixels. Mahalanobis distance and chord distance can also be used for measuring dissimilarity [17]. Computing the color histogram of images globally or locally and comparing them with a distance metric is an another approach. In order to use the aforementioned methods efficiently, normalization can be utilized. Normalized color histograms and normalized distance values are examples. Another approach is to compute the normalized dot product of input images which can be used as a similarity metric since the value will be 1 if the two input images are same. Correlation based methods are also widely used for this purpose. Pearson correlation coefficient and Spearmen rank coefficient are two examples that based on computing the correlation value between images [17]. The main problem with aforementioned methods are their vulnerability to spatial differences. Even a misregistration of 1 pixel results in huge difference in the metric value. Local spatial changes can be handled with a window approach. However, the global changes still persist and effect the final outcome of the metric. A method which employs the spatial relationships can model the difference relatively better. Such an approach is Structural Similarity Index Measure (SSIM) which is introduced in 2004 [59]. SSIM is explained in detail at the following subsection.

2.3.1 Structural Similarity Index Measure (SSIM)

Structural Similarity Index Measure is a full reference metric which is defined for measuring similarity between images. Full reference metrics are used for measuring the image quality of a compressed image by comparing it to the raw and compression free original version of the image. Thus, experiments show that, SSIM is also a good metric for measuring similarities in change detection applications. Some other full reference metric examples are mean squared error (MSE) and peak signal-to-noise-ratio. The main difference of SSIM from these metrics is that SSIM measures the *perceived differences in structural information* not the *numerical errors*. With this paradigm, it is easier to detect changes that are perceivable by human visual system which we also desire. In the definition of SSIM, the similarity measurement task is reduced to comparison of three components that are luminance, contrast and structure. This way the illumination invariance is also achieved for better structure observation. For each image signal in direction x in a spatial patch, the luminance component is defined as a function of the mean intensity value:

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i. \tag{2.18}$$

The contrast of the image is defined as a function of the standard deviation:

$$\sigma_x = \left[\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2\right]^{1/2} \tag{2.19}$$

The structural similarity comparison is then made on the signals normalized with its own standard deviation, $(x-\mu_x)/\sigma_x$. After defining the values for each component, the comparison functions in the SSIM are formulated as $l(x, y)$ which is luminance, $c(x, y)$ which is contrast and $s(x, y)$ which is the structural similarity. The luminance comparison function $l(x, y)$ is defined as:

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \tag{2.20}$$

where constant C_1 is defined as $(K_1L)^2$. K_1 is a small predefined constant and L is the dynamic range of the image intensity values in this definition. Contrast comparison function $c(x, y)$ is then defined as where $C_2 = (K_2L)^2$;

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \tag{2.21}$$

The structural comparison function $s(x, y)$ is obtained with the correlation of normalized signals as;

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}, \tag{2.22}$$

where σ_{xy} is defined as;

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y) \tag{2.23}$$

and $C_3 = C_2/2$. Combining these three comparison functions yields the final formulation of the SSIM:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \tag{2.24}$$

In the original article [59], constants are given as $K_1 = 0.01$ and $K_2 = 0.03$ by default. SSIM works locally on image patches and computes the similarity between patches. For globalization of the method, authors proposes the usage of the mean SSIM value for the whole image.

2.4 Segmentation Prior to Change Detection

The problem of image segmentation has been a widely studied research topic for over 30 years in computer vision literature [48, 43, 27]. Over hundred different segmentation techniques are proposed in the literature for gray-level and color images, each having their own advantages and disadvantages. Many segmentation methods are based on discontinuity and similarity of pixels regarding their neighborhood [27]. The survey published by Lucchese and Mitra in 2001 [43] groups segmentation techniques into 3 categories as feature space based techniques (clustering, histogram thresholding, ...), image domain based techniques (split-and-merge, region growing, edge based,neural-network) and physics based techniques. Different color spaces like RGB, HSI, HSV, $L^*u^*v^*$ and $L^*a^*b^*$ are utilized in segmentation procedures. In this study, mean shift segmentation algorithm proposed by Comaniciu and Meer [19] is used.

2.4.1 Mean Shift Segmentation

Proposed by Fukunaga and Hostler in 1975, mean shift is a non parametric feature space analysis algorithm [28]. The formalization and additions to the proposed method is done by Cheng [18]. Mean shift theory is then adapted into an image segmentation algorithm by proving the convergence property for discrete data [19]. Mean shift aims to find the local maxima points in data by estimating a density function. This density function assumes that the data points are sampled from the function itself. For estimating the density function, kernel functions which identify the weights of nearby points for next iteration are utilized. The kernel estimation method is chosen as the Parzen window technique [49]. The formulation of the d-variate kernel function $K(x)$ and the dxd bandwidth matrix $\mathbf{H}$ for n data points $x_i = 1, 2....., n$ in the d-dimensional space R^d are

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} K_H(x - x_i) \tag{2.25}$$

$$K_H(x) = |H|^{-1/2} K(H^{-1/2}x) \tag{2.26}$$

Density estimator is obtained by considering the constraints defined in [19] and radial symmetry. It is formulated as,

$$(\hat{f})_{h,K}(x) = \frac{c_{k,d}}{nh^d}\sum_{i=1}^{n} k(||(\frac{x - x_i}{h})||^2), \tag{2.27}$$

where c_k is a constant. Analysis of the feature space starts with the detection of modes. Modes are expected to be at $\triangledown f(x) = 0$. By using the linearity of (2.27), the density gradient estimator is formulated as:

$$\hat{\triangledown} f_{h,K}(x) = \frac{2c_{k,d}}{nh^{d+2}}\sum_{i=1}^{n}(x_i - x)g(||\frac{x - x_i}{h}||^2), \tag{2.28}$$

where $g(x)$ is

$$g(x) = -k^{'}(x). \tag{2.29}$$

Then, with defining kernel $G(x) = c_{d,d}g(||x||^2)$, *mean-shift vector* is formulated as:

$$\mathbf{m}_{h,G}(x) = \frac{1}{2}h^2 c \frac{\hat{\triangledown} f_{h,K}(x)}{\hat{f}_{h,G}(x)}. \tag{2.30}$$

One interesting property of mean shift vector is that it always points to the direction of maximum density increase. The estimated density, then, can be found by following the path shown by mean shift. General procedure of the mean shift is iteratively computing the mean shift vector and redefining the position of the kernel window according to it. The proposed feature space analysis technique could be used for different applications. In [19], two approaches, smoothing and segmentation, for image analysis are discussed. For representing the image in the context, a joint spatial-range domain is defined. Spatial domain represents the location of the pixels. Range domain holds the actual values and its dimension is dependent on number of image channels (1 for grey, 3 for RGB, more for multispectral). Joint domain with a dimension of $d = p + 2$ is formed by concatenating the range and spatial vectors. With respect to the joint domain, multivariate kernel is defined as:

$$K_{h_s h_r}(x) = \frac{C}{{h_s}^2 {h_r}^2} k(\|\frac{x^s}{h_s}\|^2), \tag{2.31}$$

where C is the normalization constant, x_s and x_r are the spatial and range parts of the feature vector and h_s and h_r are the kernel bandwidths.

One of the proposed applications of the mean shift space analysis is the Mean Shift Filtering which is a smoothing that preserves discontinuity technique, also utilized in segmentation procedure. Smoothing can be defined as the processes of changing the magnitude of a pixel in the center of the window with the average magnitude of the pixels inside the window. Methods that preserve discontinuity reduce the amount of smoothing near sudden changes. Steps of mean shift filtering are:

1. Initiliaze $j = 1$ and $\mathbf{y}_{i,1} = x_i$.
2. Until $\mathbf{y} = \mathbf{y}_{i,c}$, compute $\mathbf{y}_{i,j+1}$.
3. Assign $z_i = (\mathbf{x}_i^s, \mathbf{y}_{i,c}^r)$.

where x_i is the input and z_i is the filtered output pixels in a d-dimensional domain.

The mean-shift based segmentation process that utilizes filtering associates each pixel with a significant mode of the joint domain density in its neighborhood. The steps of the segmentation are:

1. Execute mean shift filtering and keep all the information at the convergence point $z_i = y_{i,c}$.
2. Group all z_i that are closer than h_s in the spatial domain and h_r in the range domain into clusters.
3. For each pixel in the image, assign a label L with regard to cluster belonging.
4. Optionally, disregard spatial regions with an area below a predefined threshold.

When using mean-shift segmentation algorithm in the context of our current study, $L^*u^*v^*$ color space is chosen for representing the images. $L^*u^*v^*$ is stated as one of the best choices for approximating the perceptually uniform color spaces. The optional elimination step based on pixel numbers is also facilitated.

2.5 Commonly Used Features and Procedures for Change Detection

Image pairs can be analyzed for changes with different approaches. Instead of directly using image intensity values for change classification, many methods derive features or compute new images using the information from both images. As mentioned before, one of the widely used techniques is simple image differencing. This method aims to compute a difference image where each pixel corresponds to the change in intensity values. Image differencing is formulated as $D(x) = I_2(x) - I_1(x)$ [51].

Instead of directly using intensity values for differencing, other features such as edge and texture responses can be used to compute difference image. There are techniques such as image rationing and CVA which are derived from simple differencing. Rather than taking the difference of corresponding pixels in image pairs, image rationing method takes the ratio of them: $D(x) = I_2(x)/I_1(x)$ [55].

Change Vector Analysis is a more advanced version of the image differencing that can also store the direction of change in addition to magnitude [44]. Even though CVA can be applied to original bands of the image (it only changes the dimension of the change direction vector), original study which aims to detect forest changes proposes the usage of Tasseled Cap Transformation [39]. After performing the Tasseled Cap Transformation on images, brightness and greenness variables of the transformation are chosen as vectors. Then, the magnitude of the change vectors can be defined for multiple spectral bands as;

$$M_{CV} = \sqrt{(I^1_{b_1} - I^2_{b_1})^2 + (I^1_{b_2} - I^2_{b_2})^2 + ... + (I^1_{b_n} - I^2_{b_n})^2}. \tag{2.32}$$

After computing the change vectors for each pixel, magnitudes can be used to distinguish changed and unchanged regions. Directions of change vectors are used to classify the type of change.

After computing the features for change detection (change magnitude, ratio, etc.), various methods can be employed to distinguish changed regions from unchanged ones. One of the most basic procedures for classification is thresholding. With respect to a threshold value t, computed features are labeled as changed or unchanged as follows;

$$L(x) = \begin{cases} 1, & \text{if} |F(x)| > t \\ 0, & \text{else} \end{cases} \tag{2.33}$$

where L represents the label and F stands for the computed feature for the respected pixel. Finding the correct threshold t which minimizes the error is a tricky issue that is still being studied in the literature. Hard thresholding and adaptive methods based on the data are some of it. However, most of the advanced methods treat the problem as a statistical hypothesis test by assuming the pixels are derived from a statistical distribution. Then, the problem is transformed into deciding if a pixel $\mathbf{x}$ belongs to one of the two hypotheses, H_0(*unchanged*) or H_1(*changed*). If change is non-existent, the difference between images are most likely to be caused by noise. Using this information, *null hypothesis*, H_0 can be modeled. In order to check the validity of the model, a significance test could be performed on the features(*difference, ratio image, CVA, etc.*) [51]. The test is formulated as with a predefined threshold τ, which represents the false alarm rate,

$$S(x) = p(D(x)|H_0) \lessgtr^{H_0}_{H_1} \tau. \tag{2.34}$$

If it is assumed that both conditional probability distributions are known, the *likelihood ratio* can be formulated as;

$$L(x) = \frac{p(F(x)|H_1)}{p(F(x)|H_0)}. \tag{2.35}$$

Then, a threshold τ can be defined, where $P(x)$, C_{10} and C_{01} correspond to the prior probability, risk factor of labeling a changed region as changed and vice versa, respectively:

$$\tau = \frac{P(H_0)(C_{10} - C_{00})}{P(H_1)(C_{01} - C_{11})}. \tag{2.36}$$

A decision in favor of H_0 or H_1 is then made, according to the likelihood ratio at x. If the value is greater than τ, the pixel is marked as changed, otherwise it is marked as unchanged. At the end of the procedure, the decision with the minimum *Bayes risk* is made by choosing the hypothesis with the greater posterior probability. Since it is hard to describe the observations with parametric distributions and they are not known *a priori*, this procedure is challenging. In order to overcome these difficulties, attempts are made to estimate the parameters of the distributions beforehand using methods such as EM [12]. Expectation - Maximization algorithm iteratively finds the maximum likelihood estimates for incomplete data [22]. Contrary to its denotation, incomplete data refers to the parameters in many applications. The estimation is then made for these unobserved latent variables that defines the statistical models. Mentioned algorithm iteratively works on two alternative steps which are expectation (E) and maximization (M). In the expectation step, a log-likelihood expectation function is defined using the current estimate for the parameters. Maximization step finds the parameter values that maximize the expected log-likelihood for this function. Using EM, distributions of the changed and unchanged pixels can be estimated without prior knowledge, under the assumption of a certain distribution type.

Changed and unchanged pixels can be discriminated by using a clustering algorithm. K-Means and its variants are used in the current literature for change analysis [14, 29]. K-Means clustering is an iterative hard clustering algorithm that partitions the data into k clusters [33]. It works in two alternating steps. In the first step, each data point is assigned to the cluster with the nearest mean. Following this assignment, the means of the clusters are updated and the new centroids for each cluster are computed. Algorithm finishes when there are no changes of cluster assignment for all data points. K-Means algorithm is very sensitive to the initialization of clusters and do not guarantee convergence with the global optimum. Also, the choice of k greatly affects the results. In theory, there should be two clusters in the scope of change analysis, which define the changed and unchanged regions. However, in application, such an assumption generally produces unwanted results. A diagnostic step that determines the characteristics of the observations to assign a plausible cluster number is generally utilized. Another approach is the utilization of validity indexes. Samples can be clustered with a range of different k values and the quality of the clusterings can be compared to find the proper number of clusters. Four different validity indexes are employed in this study; Davies-Bouldin (DB), Calinski-Harabasz (CH), Krzanowski-Lai (KL) and Weighted Inter-Intra Ratio (Wint). All four measures the clustering quality by using the features inside the current dataset and fall inside the internal validity indexes for this reason. For each validity index, different distance metrics can be used. In this study, Euclidean distance is chosen as the metric and the formulations for the indexes are given with respect to it. DB [21] evaluates the quality by measuring the scatter within cluster and the separation of distinct clusters. For each cluster i, the scatter within cluster S_i is defined as;

$$S_i = \sqrt{\frac{1}{n_i}\sum_{j=1}^{n_i}(X_j - C_i)^2}, \tag{2.37}$$

where C is the centroid of the current cluster, n is the total number of samples and X are the samples inside the cluster. The separation between two clusters are then defined using their centroids C_i and C_j and defined as;

$$M_{i,j} = \sqrt{(C_{i_1} - C_{j_1})^2 + (C_{i_2} - C_{j_2})^2}. \tag{2.38}$$

Then, combining Equation 2.38 and 2.39, the quality measure $R_{i,j}$ for cluster numbers i and j is found as;

$$Ri, j = \frac{S_i + S_j}{M_{i,j}}. \tag{2.39}$$

Generalizing Equation 2.40 to a clustering result with k clusters, DB Index is then formulated as;

$$DB(k) = \frac{1}{k}\sum_{i=1}^{k}\max_{j:i\neq j} Ri, j. \tag{2.40}$$

Since DB index measures the ratio of within cluster scatter and cluster separation, a low value is desired for a high quality clustering. Second index, CH [13], measures the clustering efficiency by computing the between cluster and within cluster distances of the samples. Within cluster (WC) and between cluster (BC) distance measurements for a clustering with k clusters are given as;

$$WC(k) = \frac{1}{2}\sum_{i=1}^{k}\sum_{i,j\in C_i}\sqrt{(X_{i_1} - X_{j_1})^2 + (X_{i_2} - X_{j_2})^2}, \tag{2.41}$$

$$BC(k) = \frac{1}{2}\sum_{i=1}^{k}\sum_{i,j\notin C_i}\sqrt{(X_{i_1} - X_{j_1})^2 + (X_{i_2} - X_{j_2})^2}, \tag{2.42}$$

respectively where C_i is the current cluster. Combining Equation 2.42 and 2.43, the CH Index is given as

$$CH(k) = \frac{BC(k)/(k-1)}{WC(k)/(n-k)}, \tag{2.43}$$

where n is the total number of samples. A higher CH value indicates a better clustering. KL index uses within cluster distances as a basis for the quality measurement [40]. This index finds the number of k where the within cluster distance difference with the clustering $k-1$ is maximized. For this purpose, a difference value D between following k values are defined as;

$$D(k) = (k-1)^{2/p}WC(k-1) - k^{2/p}WC(k), \tag{2.44}$$

where p equals the dimension of the feature space. Using D, KL index is defined as:

$$KL(k) = \left| \frac{D(k)}{D(k+1)} \right|. \tag{2.45}$$

The k value which maximizes the function $KL(k)$ is chosen as the best clustering result. The last validity index, Weighted Inter-Intra Ratio (Wint), finds the cluster number k which maximizes the intra-cluster and minimizes the inter-cluster similarity. The intra and inter cluster similarities are defined as;

$$Intra(i) = \frac{2}{(n_i - 1) * n_i} \sum_{i,j \in C_i, j > i} \sqrt{(X_{i_1} - X_{j_1})^2 + (X_{i_2} - X_{j_2})^2}, \tag{2.46}$$

$$Inter(i, j) = \frac{1}{n_i * n_j} \sum_{i \in C_i, j \in C_j} \sqrt{(X_{i_1} - X_{j_1})^2 + (X_{i_2} - X_{j_2})^2}, \tag{2.47}$$

where n_i is the number of samples belonging to cluster C_i. With the formulations in Equation 2.47 and 2.48, the Wint index for k clusters is defined as:

$$Wint(k) = 1 - \frac{\sum_{i=1}^{k} \frac{n_i}{n - n_i} \sum_{j \notin C_i} n_j * Inter(i, j)}{\sum_{i=1}^{k} n_i * Intra(i)}. \tag{2.48}$$

Wint index reaches high values when the within cluster similarities are high and between cluster similarities are low, which is a desired property in good clusterings. These indexes can be used to determine the correct number of clusters in a set of observations. When classifying the changed regions with clustering, a number of trials can be made with different number of clusters and the clustering result with the desired index value can be chosen as the actual result.

2.6 Salient Region Analysis

A salient part of an image is defined as a region that is highly distinctive and creates a visual arousal in the early stages of human visual system immediately [38]. Different models for human visual saliency have been proposed in the literature. Most of them are theoretical and are not applicable to real life problems. Defining the salient regions in an image can improve the performance of change detection applications in both at classification and postprocessing stages. In the former case, the obtained saliency map can be used as a feature for detecting changes. It can be also used to eliminate false changes after classification step as the second case suggests. Since the aim of this study is to detect salient changes as man made objects, the application of saliency algorithms seems favorable. The first attempts for defining visual saliency can be associated with the detection of edges in an image. This approach defines edges as more attractive regions in the whole image. Generally, the gradient magnitude is used for discriminating edges, but there are also other saliency measures defined for edges as lifetime, wigglines, width and clutter, local intensity and contrast, projection onto edge-subspace and phase congruency [53]. Several of these measures are applicable for any salient region and they are defined as:

1. **Lifetime**: Salient regions should persist over different scales. This can be achieved by blurring the image and tracking the regions across scale.

2. **Local Intensity and Contrast**: Local intensity and contrast differences are great measures for saliency. Before globally applying edge thresholding, locally normalizing image gradients could produce better results.

It should be noted that most measures applicable to edge saliency are also valid for detecting any salient region since edges are a great indicator for salient regions.

Apart from geometric features, rarity is also defined as an important indicator for saliency [30]. Even though it is used in literature for saliency measurement, the problems regarding this approach is stated in the literature. When the rarity increases, the salient regions decreases, so the rarity is defined by the measure it uses. Local complexity is then presented as an another measure. By computing the Shannon Entropy of local attributes, a measure for saliency can be defined. Regions with higher signal complexity are more likely to be salient regions [38].

A saliency measurement method that benefits from the human visual system is proposed in 2000 [37]. The proposed approach is known as Itti-Koch model in the literature. This model simulates the visual system for visual attention by combining three modalities, orientation, intensity and color. Algorithm starts by extracting these features at several spatial scales by utilizing Gaussian pyramids. For each feature, a center-surround structure is created and the feature is computed according to it, resulting in 42 dimensional feature maps. For combining these maps, an approach that inspires from the connections in early visual areas is used. The interactions between the maps are modeled by a two-dimensional Difference-of-Gaussians. After computing the three *conspicuity maps* for orientation, intensity and color from 42 feature maps, these three are combined into a unique saliency map by linear combinations. The proposed approach performed decently in both synthetic images and real world complex scenes.

Another biologically plausible visual saliency model is proposed by Harel et. al. in 2006 [31]. Proposed method is bottom-up and utilizes graph algorithms to detect visually salient regions in images. The proposed model is named as Graph Based Visual Saliency (GBVS). Markovian chains are used in the calculation of saliency values. Over different image maps, Markov chains are defined and the activation values at equilibrium distribution over map locations are treated as saliency values. The proposed approach is summarized by three steps, which are feature extraction, forming of an activation map based on the features and the normalization of the activation map. Computation of feature maps are handled by linear filtering, which is succeeded by elementary nonlinearity. Then a Markov chain is defined on the feature map M, by creating a fully connected directed graph G_A, where all nodes of the lattice M are connected to each other. The weights of the edges are defined proportional to the dissimilarity of the nodes and to nearness in the domain M. This way, probable salient nodes, which show dissimilarity to neighboring nods, will have higher activation values. Then, computed activation map is normalized to eliminate uniformity, which may cause less informativeness. The normalization step is again handled with the utilization of a Markovian procedure. The resulting normalized activation map then defines the salient regions in the image.

CHAPTER 3

CHANGE DETECTION FRAMEWORK WITH DOUBLE SEGMENTATION

In this chapter, the algorithm that is proposed for detecting changes in image pairs is explained in detail. This chapter is divided into 6 sections, each elaborating a crucial part of the conducted research. The proposed approach is coded using MATLAB and C++ and tested in various datasets. Images from artificial or real datasets are used for showing the results of each step. In the first section, an overview of the proposed framework is given. Some common problems encountered in change detection applications are also mentioned in this section. Second section, preprocessing, explains and demonstrates methods used for image normalization, similarity measurement and registration. Third section explains the double segmentation process on image pairs. Parameter benchmarking for mean-shift algorithm on the artificial dataset is also mentioned in this section. Fourth section discusses different feature selections and their combinations for achieving a better change analysis process. Features mentioned in this section are intensity, ratio, range filter (texture) and saliency map difference values. Fifth section focuses on classifying the changed regions. Different methods for discriminating changed regions are discussed in this section. Last section discusses post-processing methods for obtaining a better change map. For eliminating false changes and obtaining a better visual change map, segment elimination and reconstruction based on segment inspection is proposed. Saliency analysis is also discussed for detecting false changes in the last section.

3.1 Introduction

This study aims to develop a complete change detection algorithm for significantly similar images. The application area of the study is chosen as remote sensing and the steps of the algorithm are organized according to this domain. Theoretically, proposed approach can work in any type of images and datasets, but utilization of different methods and features can provide better results in such cases. Proposed approach is established on double segmentation of image pairs. Each image is segmented and labeled separately and different labellings are used in both images, resulting in 2 different labellings for each. Feature extraction and classification are then made using these labellings and obtained results are combined to produce final decision. The major aim of the application on remote sensing is to detect human-made changes. Environmental changes such as tide of waters, forestation and natural disasters are also in the scope. However, small changes in the environment like waves on water, vegetation differences, etc. are not considered. The proposed approach is unsupervised and does not use any training set apart from parameter optimization. The main purpose of this study is to develop a change analysis method which is adaptable to different datasets by parameter change. Algorithm starts by likening the spectral characteristics of the input images, by employing image normalization methods. It continues by spatially registering the input images. The algorithm is designed to work on datasets with small changes, therefore it should detect two significantly different images. In order to detect such cases, a similarity measurement metric is included in the proposed method. After preparing the images at preprocessing step, both images are segmented and their labellings are obtained. The segmentation procedure for I_1 and I_2 produces two different labellings, L_1 and L_2. Feature extraction is done with respect to these two different labellings, resulting in two different feature vectors. These features and their combinations are used for classifying the changes in the next step. Classification step creates two different change maps and they are combined to provide the initial change results. Initial results are then improved using segment analysis and visual saliency. General flowchart of the proposed method can be seen at Figure 3.1.

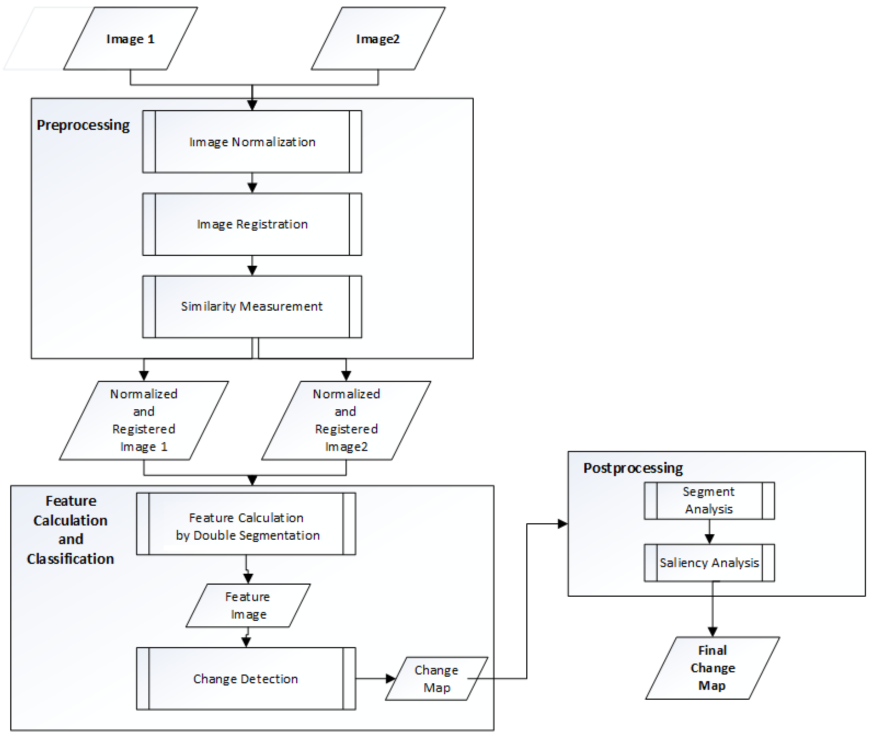


Figure 3.1: Flowchart of the proposed method

Features used for change classification are generally derived from the intensity differences of input images. When designing a change detection method, an assumption about the intensity difference values of changed and unchanged regions is made. Changed regions are expected to have high intensity differences between images. Conversely, unchanged regions are assumed to have zero or negligible intensity differences. However, this assumption is not satisfied for all cases. In order to illustrate different cases that are not compatible with this assumption, images from an artificial dataset are used. Varying illumination between image pairs can cause local shadows which are not suppressible with image normalization. A visual example of the mentioned case is given in Figure 3.2.

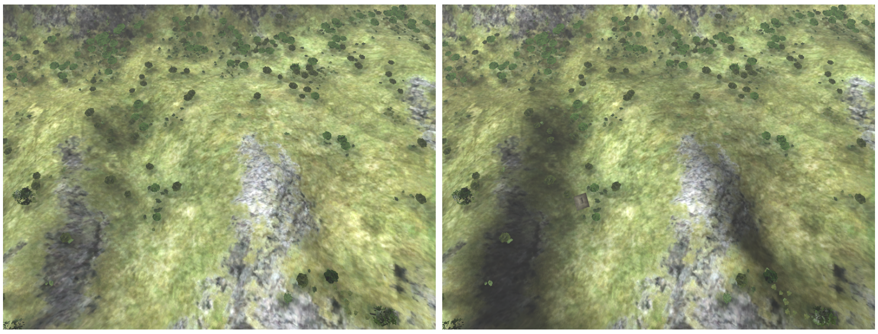


Figure 3.2: Local illumination changes-Shadows

Shaded regions that are visible in Figure 3.2 are not regarded as changes in the scope of the study, but have high intensity difference values between images. Such regions effect the change detection process and cause change maps with false positives. In a case where changed regions have significantly lower intensity differences than the shaded regions, change detection process may fail to detect actual changes. Shades can also occur on man made objects, causing excessive intensity difference between images. Such objects can be completely unchanged, however their different spectral characteristics make them candidates for change. An example of such a case is given at Figure 3.3.

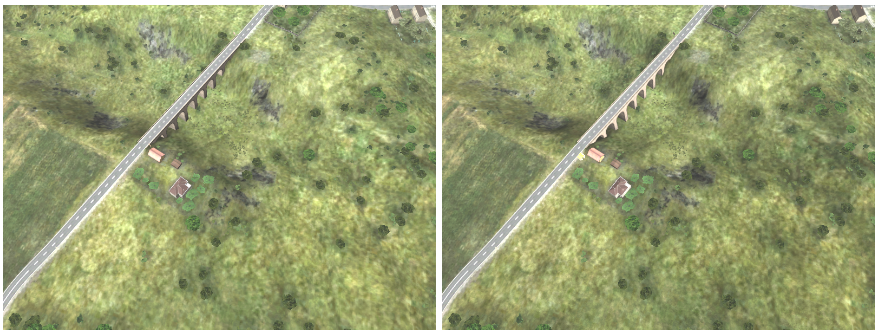


Figure 3.3: Local illumination changes-Objects

The legs of the bridge in image pair shown in Figure 3.3 are structurally same. However, the intensity values are highly different because of the shadow. The intensity difference on the bridge legs is even higher than the actual change object. Another difficulty arises when the actual change regions show similar spectral characteristics in image pairs. Change detection process becomes even more difficult when unchanged regions with different spectral characteristics are present in the scene. Such an example can be seen at Figure 3.4. A small hut is built in the second image but it shows similar spectral characteristics with the region that is built on. However, the buildings at the top of the both images have different intensity values that are caused by varying illumination. Even though there are no changes in that region, these regions are more likely to be marked as changes. Such regions can, also, effect the decision criteria in classification, causing actual changed regions to be marked as unchanged.

Figure 3.4: Actual change with small intensity differences

These problems are encountered in many change detection applications. They can be solved with the help of prior knowledge such as acquisition time of the images, sun azimuth and angle, objects in the scene and the surface model of the image. However, such prior information is not available in general. In the scope of the current study, formerly mentioned issues effect nearly all proposed steps. Local illumination differences cause the image normalization to be less effective, shadows and spectral changes can result in erroneous segmentation results and the classification step can be affected by the results of the former steps. The proposed method aims to address these issues by normalizing, selecting different features and utilizing effective decision algorithms but the aforementioned effects can not be canceled entirely.

3.2 Preprocessing for Change Detection

Most of the change detection methods utilize preprocessing steps to achieve better results. The proposed method have three major preprocessing phases. First, the spectral characteristics of input images are likened using image normalization. This method is applied first because different spectral characteristics also effect latter preprocessing steps. Two images should be spatially registered before any change analysis, since two pixels in image pairs should represent the same area. Image registration is done at the second step of the preprocessing. Finally, in order to detect completely different image pairs, a similarity measurement metric is utilized. If the similarity of two images is below a predefined threshold, change detection process is terminated.

3.2.1 Image Normalization

This study uses the intensity values of image pixels as main features for change detection. Hence, both images should show similar spectral characteristics to obtain a meaningful change map. Also, latter preprocessing steps assume both images have similar intensity values. When extracting and matching keypoints for registration, spectral values are used as main features. Proposed similarity measurement metric detects structural similarities, but it still works on intensity, contrast and variance. Spectral characteristics of image pairs acquired from the same area may still differ because of seasonal changes, atmospheric variations, different kind of satellites and illumination changes. Therefore, a preprocessing step which associates intensity values of image pairs is needed. For this purpose, image normalization is applied.

The proposed method normalizes the pixel intensity values of one image to have the same mean and variance as the other [51]. For an image with ixj dimensions and k spectral bands, the procedure is defined as,

$$I_2(i,j,k) = \frac{\sqrt{\sigma_1(k,k)}}{\sqrt{\sigma_2(k,k)}}(I_2(i,j,k) - \mu_2(k)) + \mu_1(k). \tag{3.1}$$

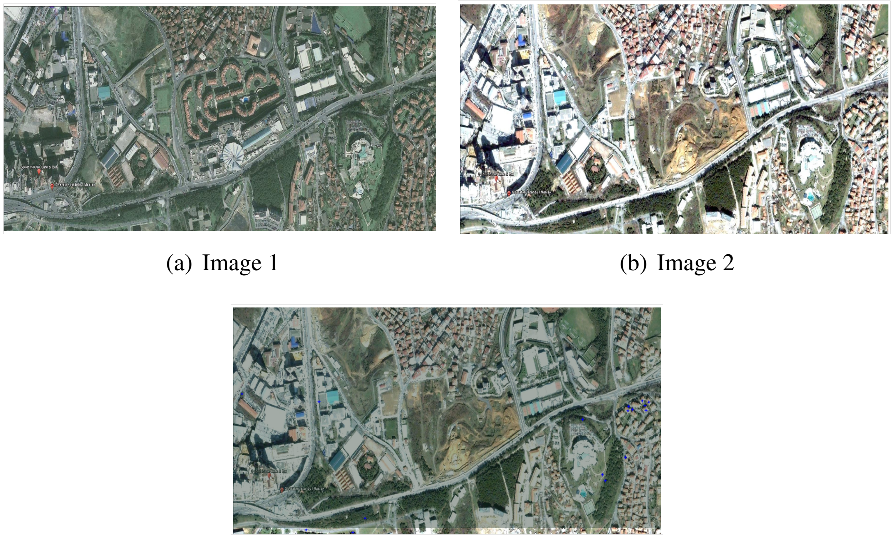
Using the Equation 3.1, each spectral band of I_2 is transformed to have the same mean and variance as the corresponding spectral band of I_1. The results are illustrated with two examples. The first example, which is shown in Figure 3.5, deals with different type of satellites. Same scene is acquired at different times using different type of satellites, Quickbird(a) and Ikonos(b). Different satellites generally produce images with different spectral characteristics, which makes change analysis procedure harder. Result of the proposed image normalization can be seen at Figure 3.5 (c).

(a) Quickbird Image (b) Ikonos Image (c) Normalized Ikonos Image

Figure 3.5: Image Normalization with Different Satellite Types

The second example illustrates spectral differences caused by seasonal and atmospheric variations. Both images are acquired with the same type of satellite and obtained from Google Earth. The effects of the normalization process are visualized in Figure 3.6.

These two examples (and other experiments) show that the proposed method for normalization is successful in associating image spectral characteristics.



(a) Image 1 (b) Image 2

(c) Normalized Image 2

Figure 3.6: Image Normalization for Different Atmospheric Conditions

3.2.2 Registration

All change detection algorithms assume co-registered image pairs. Matching pixels in image pair should represent the same area. Therefore, both images must be overlaid before executing a change detection algorithm. This requirement is resolved by utilizing an image registration process. A four step registration process is utilized in the proposed solution:

1. SIFT is used to detect and match keypoints.
2. Matchings are than analyzed using RANSAC to detect and eliminate possible outliers.
3. A homography matrix that defines an affine transformation is computed by using keypoint matches.
4. Second image is transformed using the homography matrix.

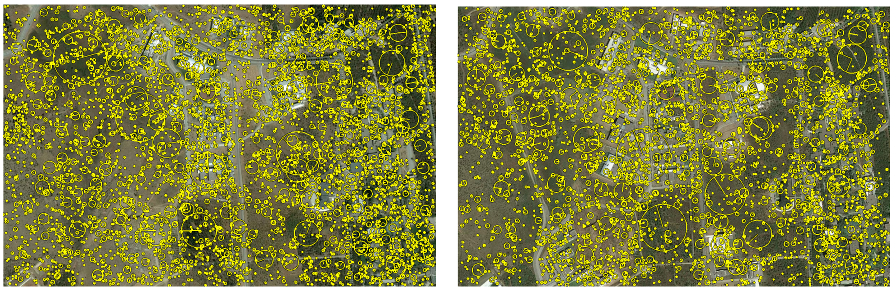
Also, the matchings are tuned at subpixel level using normalized cross-correlation to achieve better results. There is no guarantee that two images intersect entirely. So after the registration is completed, non-intersecting parts of the both images are discarded. For the implementation of registration process, VLFeat library is used [57]. In order to illustrate the registration process, an image pair acquired from Google Earth is used. Both images cover the METU Technopolis area and are chosen because many newly constructed regions are present in the second image. Images are acquired in 2003 and 2009, respectively, and can be seen in Figure 3.7.

SIFT procedure works in grayscale images, so both images are converted to grayscale before proceeding. Then, the procedure that extracts SIFT features are called with default parame-

(a) Image-2003 (b) Image-2009

Figure 3.7: Image pair used in registration process

ters. Most important features are the peak and edge thresholds. Peaks in the Difference of Gaussians scale space with a smaller value than peak threshold are eliminated. The default value of this parameter is 0. Therefore, in this example, no peaks are eliminated. Similar to peak threshold, edge threshold parameter eliminates the peaks with a curvature value smaller than it. Increasing this threshold results in more descriptors and the default value is set as 10. Extracted descriptors using default parameters are shown in Figure 3.8.



(a) Image-2003 (b) Image-2009

Figure 3.8: Extracted SIFT Descriptors

Using these parameters, 4706 and 3984 keypoint descriptors are extracted for images. The circles show the scale of the descriptor, where as the radii presents the orientation. These parameters can be optimized according to input images. After obtaining the descriptors, the matchings are found by a basic matching algorithm which finds the closest descriptor according to the $L2$ norm. The descriptor matchings are visualized in Figure 3.9.

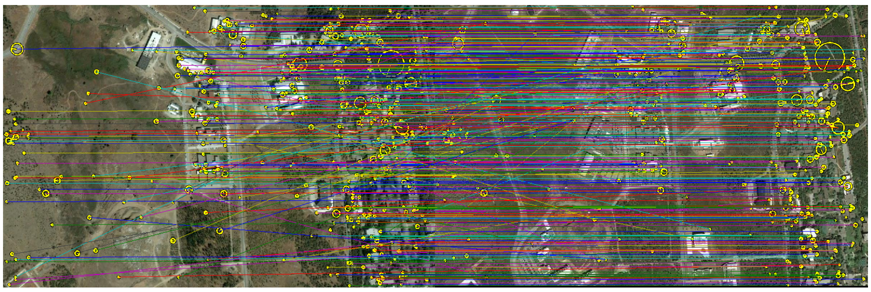


Figure 3.9: Matched SIFT Descriptors

386 descriptors are matched in this example. If the matchings are carefully examined, some false matchings can be seen. Even one false matching can completely change the homography matrix that will define the affine transformation between images. In order to detect such cases and eliminate them, RANSAC is utilized. Figure 3.10 shows the descriptor matchings after the elimination of outliers.

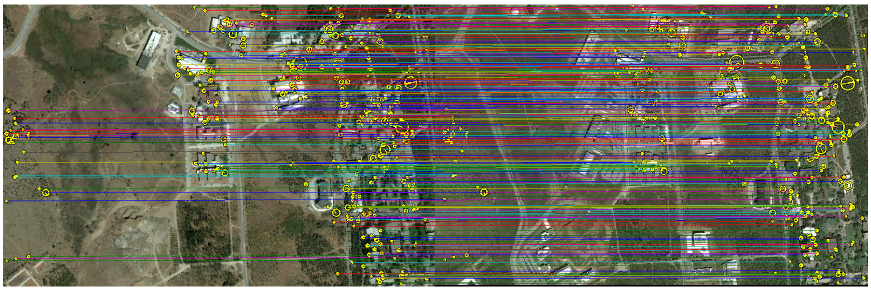


Figure 3.10: Matched SIFT Descriptors after RANSAC

From 386 matchings, 76 matchings are eliminated as outliers. Even in a non-complex scene such as the one in Figure 3.7, false matchings can be produced and should be eliminated. Result in Figure 3.10 shows the necessity of an outlier detection algorithm. These matchings are, also, tuned using normalized cross-correlation. The homography matrix that defines the affine transformation is estimated in the RANSAC process. Using the homography matrix, second image is transformed onto the reference image. In Figure 3.11, the registered second image is overlaid onto the first image with a transparency coefficient to create a mosaic image.

Figure 3.11: Mosaic image after registration

After the second image is registered, both images are checked for intersecting regions. Non-intersecting regions are then eliminated from both images to produce an image pair fully co-registered. The proposed registration procedure have some weaknesses. First of all, 2D homography is calculated with planar assumption. Therefore, if the viewing angles of the images differ greatly, the registration process can fail. Even though the planar registration is valid, the objects may occur in different places. Images with such properties are not applicable for change detection either. Such problems can only be solved by additional information such as digital surface model, object information at the scene etc.. Hence, such cases are excluded from the current scope of the study. Registration of two images with great differences is another problem. In such cases, the matchings can concentrate on a specific region of the images, resulting in a valid registration at those regions and insufficient at others. This possibility grows when the size of the input images increase. Since the proposed changed detection algorithm aims to detect changes between similar images, proposed registration method is considered sufficient.

3.2.3 Similarity Measurement

The last step of the preprocessing procedure is to determine a similarity metric. Unlike the former two steps, no changes to input images are made in this step. This step aims to measure the similarity of input images. The proposed change detection algorithm intends to work on similar datasets to detect minor changes. It also needs to detect when two significantly different images are given as input. Since the changes that the algorithm focuses on generate structural differences, Structural Similarity Index Measure (SSIM) is chosen as the similarity metric. SSIM is a full reference metric which aims to measure the similarity of an original image and its compressed version. In order to make sure that this metric can be used for measuring similarity in the scope of this application, some experiments are made. In the experiments, the default values for K, $0.01, 0.03$, are used. The dynamic range of the images are chosen as 8 bits, with the maximum intensity value of 255. SSIM works on a window basis and the window used in the experiments is a $11x11$ square Gaussian low-pass filter with a standard deviation of 1.5. These values are default values in the related article [59] and found empirically. A co-registered set of image pairs are acquired from an artificial dataset is used in the experiments. As expected, if same images are provided to SSIM, the similarity value is computed as 1. The similarity of an image pair with a small difference caused by a newly built house (and illumination differences) is then calculated. The images are shown in Figure 3.12.

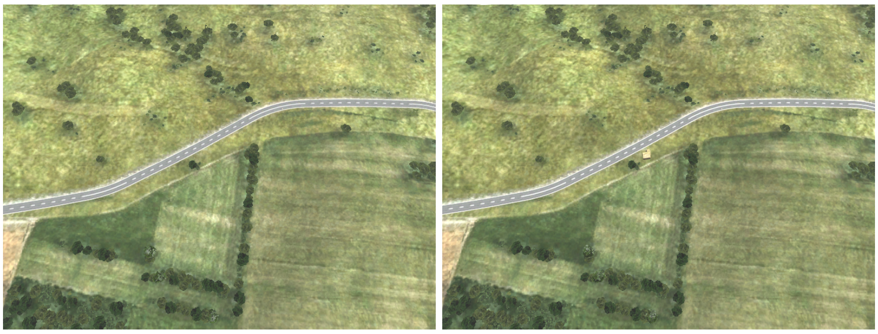


Figure 3.12: Image pair used in SSIM calculation

The mean SSIM value is then found as 0.9914. This proves that a small change that covers less than the 5 percent of the image effects the measurement slightly. In order to fully understand the behavior of SSIM, artificial changes are made on the same image. Black boxes are iteratively added to the first image and the SSIM values are calculated. Artificially modified ten images are illustrated in Figure 3.13.

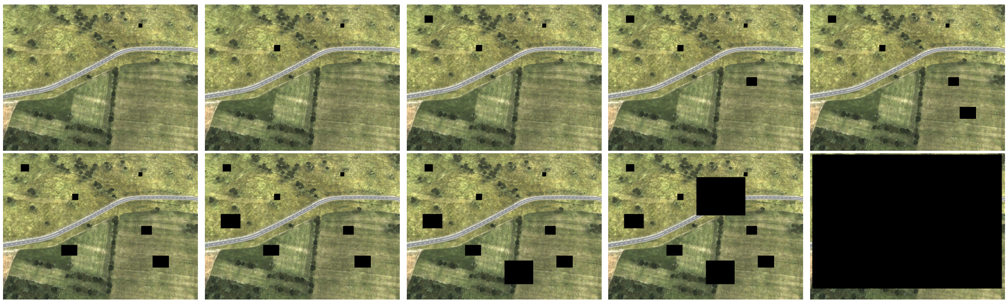


Figure 3.13: Modified images for SSIM calculation

For each modified image, SSIM values are calculated with respect to the unmodified one. Calculated values are plotted in Figure 3.14.

As expected, small changes do not greatly effect the SSIM value, but it gradually reduces with every included change. With small local changes, SSIM values are dropped to 0.8 at maximum. However, a change effecting the whole image results in a SSIM measure less than 0.1. These results obtained with the artificial dataset show that SSIM satisfies the requirements and can be used as a similarity indicator for the proposed approach. On the other hand, results obtained with real images are a bit discouraging. SSIM value is, also, calculated on image pairs that are depicted at Figure 3.6 and 3.7 with and without image normalization and registration. The image pair in Figure 3.7 with only normalization, produces a SSIM value of 0.1792. If the input images are registered using the aforementioned method, value is increased to 0.5062. Images from Figure 3.6 generate the SSIM value 0.0413 if only normalization applied. With the addition of registration process, this value is ascended to 0.2688.

The registration procedure does not guarantee a completely correct result. Since there are no ground truths for assessing registration performance, visual evaluation is performed. Visual inspection shows some differences between the registration of input pairs from Figure 3.6 and 3.7. Some misregistrated regions can be easily identified by visual inspection at images shown at Figure 3.6. Even though it is hard to detect misregistered regions from Figure 3.7, a basic image differencing between pair reveals some registration errors. Visual inspection and image differencing shows that registration of image pair Figure 3.7 is much better than Figure 3.6, but still far from perfect. The experiments with SSIM using these pairs and the artificial dataset show that SSIM is highly vulnerable to registration errors. Each image pair from artificial dataset is co-registered, which gives expected values of SSIM. However, the experiments with unregistered real world images provide different results. This shows that, the threshold for deciding if an image pair is highly similar is highly dependent on the image pair and its registration quality. It is hard to detect a hard threshold for measuring similarity that is applicable to all inputs. Therefore, for the manually registered image pairs, thresholds should be defined accordingly.

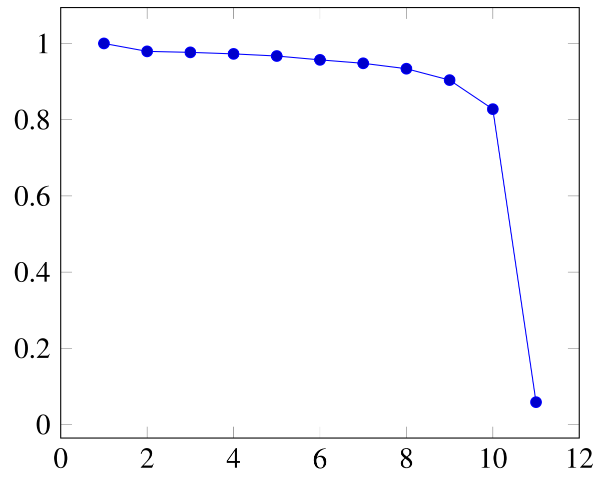


Figure 3.14: SSIM values obtained from modified images

3.3 Double Segmentation

Most of the change detection algorithms work on pixel basis. This approach generally produces relatively noisy results. Since our goal in this thesis is to detect changes of man-made objects, a higher level classification approach is more subtle than using pixels. Some object based approaches use prior knowledge on objects or use a classification step formerly [55, 3]. Such an approach is not preferred in the proposed method, since prior classification errors greatly effect the outcome of change detection methods. Also, the proposed method aims at achieving a generalized approach which is independent of any prior information. Instead of classification or utilization of prior knowledge, proposed method groups the pixels with respect to their intensity values. These segments are used in the feature extraction and classification process. It should be noted that this step does not directly aim at classifying objects, but groups the pixels belonging to an object that might be subjected to change between image pairs.

Before explaining the Double Segmentation approach, definitions of some abbreviations and notions are given. These notions are frequently used in the document for defining the process.

- I_1 - Original Image : First image acquired earlier in the timeline.
- I_2 - Changed Image : Second image which includes man made additions to I_1.
- L_1 : Segmentation result of I_1 with parameter set 1.
- L_2 : Segmentation result of I_2 with parameter set 2.
- L_3 : Segmentation result of I_1 with parameter set 2.
- F_1 : Feature vector created using I_1, I_2 and L_1.
- F_2 : Feature vector created using I_1, I_2 and L_2.

- C_1 : Change map obtained by classifying F_1.
- C_2 : Change map obtained by classifying F_2.

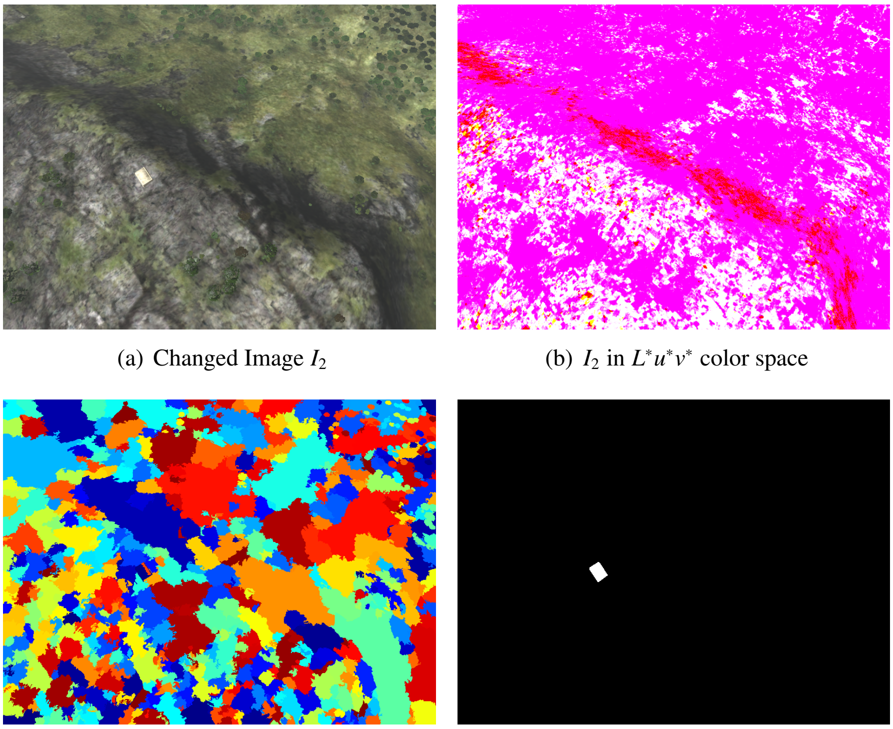
The steps of Double Segmentation are then given as,

1. Apply segmentation on I_1 and I_2 to obtain labellings L_1, L_2 and L_3.
2. Crate feature images (difference, ratio difference, etc.) using I_1 and I_2.
3. For every segment $l \in L_n$ and $n = 1$ or 2;
   - Compute mean feature values for segment l using the corresponding pixel values in feature images.
4. Using the mean feature values for L_1 and L_2 computed in the former step, create feature vectors F_1 and F_2.
5. Classify F_1 and F_2 separately and obtain change maps C_1 and C_2.
6. Obtain final change map by intersecting C_1 and C_2.
7. Apply segment reconstruction and elimination procedure on the final change map with respect to L_1 and L_3 (Explained in detail at Section 3.6).

Double segmentation method creates two different feature vectors with respect to segmentation results of original and changed images. These feature vectors can be multi dimensional, depending on the number of feature images. Average feature value for each segment is treated as a sample in the feature vector. 2 different parameter sets for segmentation are created. When segmenting I_2, changed image, it is aimed to represent the change objects with few segments as possible. A parameter set 1 is determined by benchmarking with this necessity at mind which yields L_2. If the segmentation of I_1, original image, is done with same parameter set, the changed region may exist in a relatively large segment. Therefore, a second parameter set which computes relatively smaller segments is also determined, resulting in L_1. For comparing two different segmentation results in the postprocessing step, the parameters used in segmentations should be same. Therefore, I_1 is also segmented with parameter set 1, resulting in L_3. L_3 is only used in the postprocessing step.

Mean shift segmentation algorithm is applied on images I_1 and I_2 [19]. Firstly, input images color space RGB is converted to $L^*u^*v^*$. An example changed image, I_2, its false color representation in $L^*u^*v^*$ color space, the segmentation labellings L_2 and the ground truth depicting the actual change between image pair can be seen at Figure 3.15.

As seen from Figure 3.15, the change region is divided into two separate segments in the segmentation, as expected. However, segments not belonging to change objects are also important at the feature extraction and classification steps. Like many other applications,



(a) Changed Image I_2

(b) I_2 in $L^*u^*v^*$ color space

(c) Segmentation result L_2

(d) Ground Truth

Figure 3.15: Segmentation Example

under-segmentation and over-segmentation cause difficulties for the proposed method. Under-segmented results generally do not effect the changed regions since each pixel is assumed to be labeled as changed in the classification step. However, under-segmented unchanged regions can cause noisy change detection results, probably due to local illumination changes and possible artifacts. Similarly, over-segmented labels can cause small actual changed regions to go unnoticed. In order to obtain the most effective change detection results, segmentation process should, also, produce reasonable results.

Implementation of the mean shift segmentation algorithm has three parameters; *spatial bandwidth*, *range bandwidth* and *minimum region area*, all of which effect the segmentation results highly. Using the artificial dataset, a benchmark for these parameters is done by defining a similarity metric, given in the following subsection.

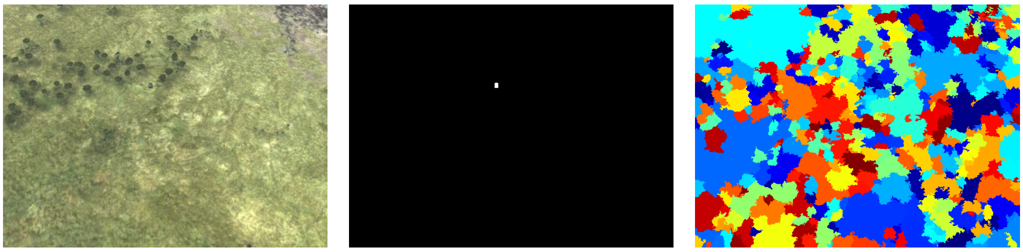
3.3.1 Benchmarking For Parameter Optimization

The purpose of this step is to find a parameter set that will segment change objects successfully. A basic metric is defined for assessing the quality of the segmentation process. This metric measures the degree of over-segmentation with respect to change objects ground truth mask. For each segment S_n intersecting the ground truth of the object GT, two subsegments are defined as $S_{i_n} = S_n \cap GT$ and $S_{o_n} = S_n - S_{i_n}$. The first subsegment defines the number of pixels inside the intersection mask of the ground truth and the extracted segment. Second one defines the number of pixels inside the difference mask of the extracted segment and the ground truth. Using these definitions, the metric M for each segment S_n that intersects the ground truth mask is defined as:

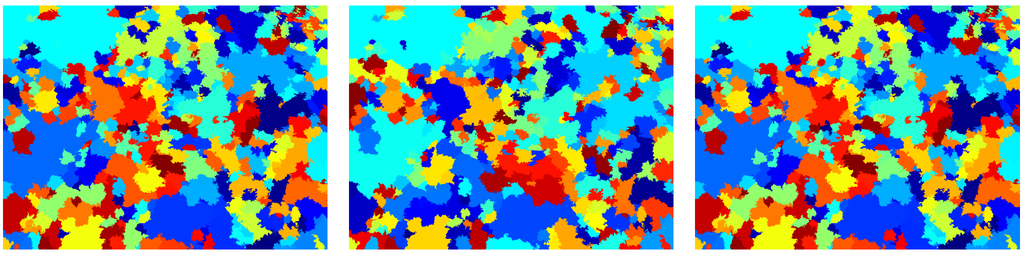
$$M = \frac{\sum_{i=1}^{n} |S_{i_n}|}{\sum_{i=1}^{n} [|S_{i_n}| + |S_{o_n}|]} \tag{3.2}$$

This metric approaches to one, as the extracted segments suits the ground truth mask. As it is seen from Equation (3.2), a small value of M shows over-segmentation. Proposed metric is a fast and easy way to assess the quality of segmentation based on the change objects. Segments with an intersection ratio ($|S_i|/|GT|$) smaller than 0.15 are discarded in the measurement. This way, a segment which intersects the change object slightly is not used in the computation. Such segments can effect the computation and may cause a lower metric value in a desired segmentation result.

From the artificial dataset of AICD [7], 15 changed images, I_2, with different sized change objects are chosen for benchmarking. For each parameter benchmark, other parameters are taken as constants. Then, M value is computed for 10 different values of the current parameter. The parameter set with the highest average M value is chosen. Similar experiments are, also, conducted on the original image, I_1. Since there is no ground truth information for real datasets, such an experiment can not be conducted. However, segmentation parameters can be selected by visual inspection on a few sample images. Figure 3.16 shows different segmentation results obtained on a sample image of the AICD with different values for *minimum region area* parameter:



(a) Changed Image I_2 (b) Object ground truth mask (c) Minimum region area = 25



(d) Minimum region area = 100 (e) Minimum region area = 150 (f) Minimum region area = 250

Figure 3.16: Effects of the minimum region area parameter on Mean Shift Segmentation results

From Figure 3.16, it can be seen that the object is extracted as a single segment with the *minimum region area* parameter values 25 and 100. Note that as the *minimum region area* parameter increases, the ground truth mask disappears in a large segment. This is a problem for the proposed application, since the change features of an object then can be blocked because of the unchanged regions in the segment. This is generally true for change objects with relatively small areas. In order to understand the effects of *minimum region area* parameter more clearly, M values are calculated for each of the 15 images with the parameter values $25, 50, 75, 100, 150, 175, 200, 225, 250$. Resulting measurements are plotted and can be seen at Figure 3.17. Some of the images that are not affected from the parameter change are excluded from the plot. X-axis shows the parameter values, Y-axis represents the calculated value of the measurement metric. Each separate image is shown with distinct lines and their ids are given as legends in the plot.

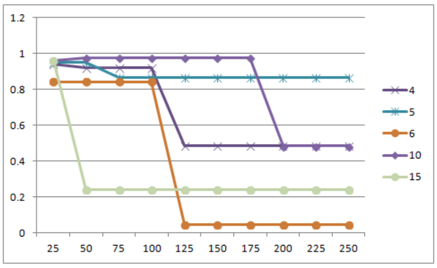
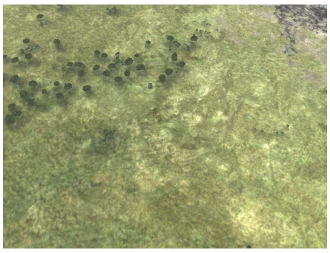


Figure 3.17: M values for various images in AICD as a function of minimum region area parameter

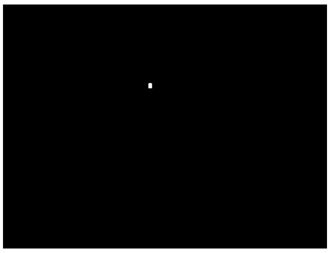
It can be seen from Figure 3.17, M values decrease with the increase of the parameter value. This is caused by over-segmentation, it is not possible to detect an object with an area A smaller than the *minimum region area* parameter. Therefore, minimum region area parameter should not be larger than the average area of the objects. When both the visual and quantitative results are inspected, it is seen that under-segmentation is not generally encountered. Most of the objects are formed by one or two segments and under-segmentation is not generally encountered. According to plot of M values at Figure 3.17, best results are obtained with the parameter value 25. However, according to qualitative results, such a small value can cause very small segments to appear in other regions and not desired. Also, it is seen that small changes in the parameter value generally do not effect the segmentation results. The best interval for the *minimum region area* parameter is then set as $50 - 100$. In the other benchmarking experiments, this parameter is taken as 75.

Similar experiments are, also, conducted for other parameters. The visual results for the *range bandwidth* parameter can be seen from Figure 3.18.

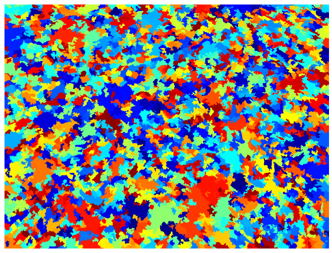
It is easily noticed that this parameter effects the segmentation results much more than the minimum region area parameter. Small changes in the parameter value can cause relatively different segmentation results. For example, Figure 3.18(c), with the *range bandwidth* parameter 1, is clearly under-segmented. With the increase of the parameter value, results tend to get over-segmented. This can be clearly seen from from Figure 3.18(c) and Figure 3.18(d). In Figure 3.18(c) the object is extracted as a single segment, however, other segments start to get too large. A parameter value which does not result in over or under-segmentation need to be



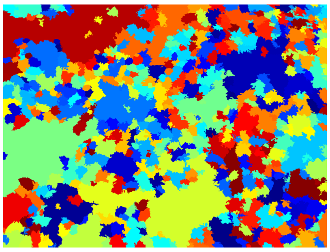
(a) Changed Image I_2



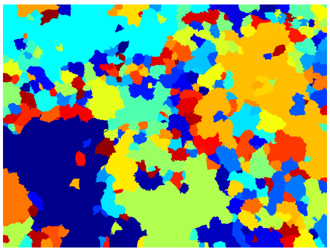
(b) Object ground truth mask



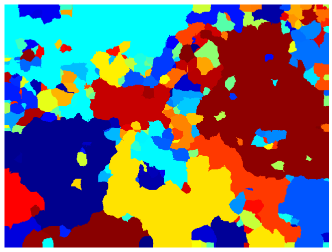
(c) Range Bandwidth = 1



(d) Range Bandwidth = 5



(e) Range Bandwidth = 10



(f) Range Bandwidth = 20

Figure 3.18: Range Bandwidth parameter effects on segmentation

detected. For this purpose, M values as a function of range bandwidth parameter are plotted. These values change more than it does with the *minimum region area* parameter. Therefore, instead of plotting the values for each image, average values for 15 changed images are plotted in Figure 3.19. X-axis depicts the measurement metric values whereas y-axis shows the corresponding parameter values.

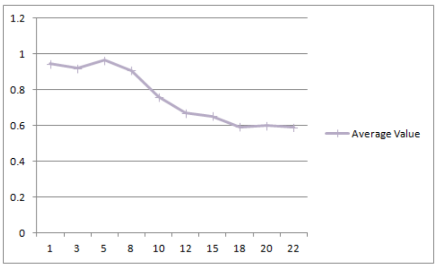
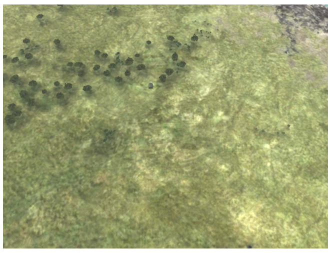


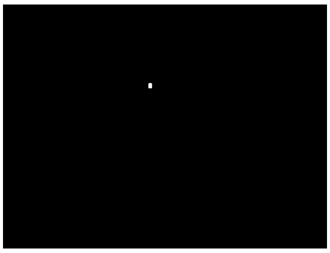
Figure 3.19: Average M values for 15 images in AICD as a function of range bandwidth parameter

According to plot of M values, best results are obtained with the range bandwidth parameter value 5. This value generally provides results neither over non under-segmented. Similar benchmarking process is made for the last parameter, *spatial bandwidth*. Visual results can be seen at Figure 3.19.

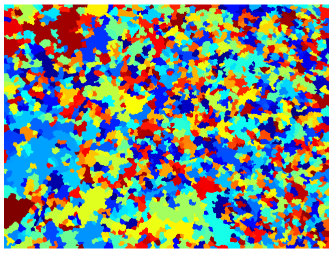
Similar to *range bandwidth* parameter, segmentation results are more vulnerable to *spatial bandwidth* than it is to *minimum region area* parameter. Low values cause under-segmentation, whereas high values create large segments that can cause over-segmentation both for the object and the other regions. In order to detect the best parameter value, average M values of 15 changed images from AICD obtained with different parameter values are plotted in Figure



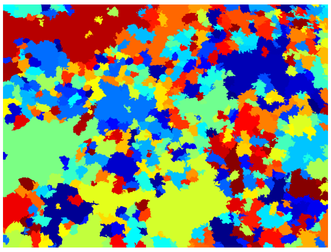
(a) Input image



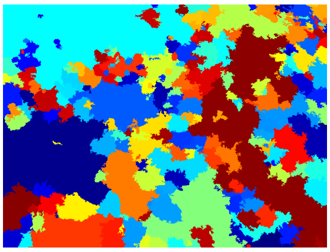
(b) Object ground truth mask



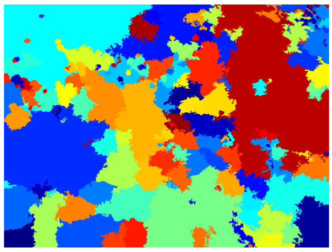
(c) Spatial Bandwidth = 5



(d) Spatial Bandwidth = 12



(e) Spatial Bandwidth = 20



(f) Spatial Bandwidth = 27

Figure 3.20: Spatial Bandwidth parameter effects on segmentation

3.21. The axes are same with the Figure 3.19.

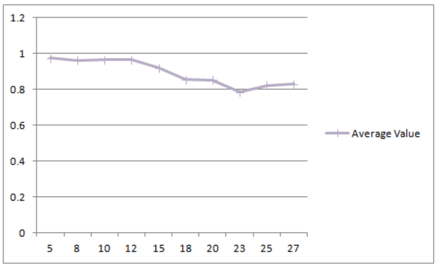


Figure 3.21: Average M values for 15 images in AICD as a function of spatial bandwidth parameter

Best M value is obtained when 12 is used as the *spatial bandwidth* parameter. At parameter value 12, objects are extracted better and divided into two segments at most. Using this information, value for *spatial bandwidth* parameter is chosen as 12.

Such benchmarking is needed when a different dataset is used for change detection. As it can be seen from the experiments, segmentation results are highly vulnerable to the parameters, especially to *spatial bandwidth* and *range bandwidth*. Therefore, a dataset with different spatial resolution or different sized change objects needs a parameter set of its own. Also, these parameters are found for the changed image I_2. These parameter values form the formerly mentioned parameter set 2 and results in labeling L_2. Segmentation parameters of the original image, parameter set 1, should be different. Experiments show that an under-segmentation is plausible for that image. In a case of over-segmentation, changed region can be inside a big segment which will cause the changed regions to go unnoticed in the feature extraction and classification step. However, using over-segmentation makes sure that the regions to be marked as change. False alarms caused by the over-segmentation will be suspended by L_2. Therefore, when segmenting the original image, it is wise to use lower *range bandwidth* and *spatial bandwidth* values.

3.4 Feature Selection

In many change detection applications, different features are calculated using the raw intensity values. Generally, these features are calculated and represented on pixel basis, creating an intermediate image. These images are then used in the classification step or for visualization. In this study, this image is named as *Feature Image*. Typically, usage of different features in the classification step is desired. However, not all type of features are representable with an image. In this section, features for change classification are discussed. Their calculation, resulting feature images or feature plots are given. Features that are not representable by images are either transformed into feature images or their distinction capabilities are shown using plots. In order to inspect possible features, same subset including 15 image pairs of the artificial dataset AICD that is used in the segmentation benchmark section is used.

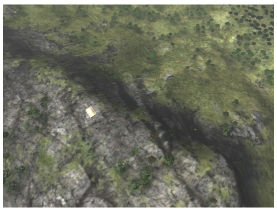
3.4.1 Difference Image

A basic and widely used feature for change classification is the difference image[51]. Pixel intensity values of the image pairs are subtracted to create a difference image. Remote sensing images generally include more than one bands, in this case each corresponding band value is used. Then, the difference image can be calculated by summing the difference values on the different bands. Normalization can be used in this step. For images I_1 and I_2 with n bands, the difference image D can be calculated with normalization as:

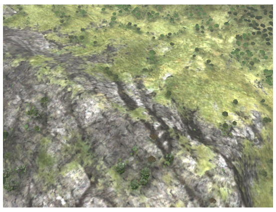
$$D = \frac{\sum_{i=1}^{n} \left[\left| I_1^n - I_2^n \right| \right]}{n} \tag{3.3}$$

Every pixel in the difference image represents the intensity difference magnitude of the pixel pair at this spatial location. Example difference images obtained with and without image normalization can be seen at Figure 3.22.

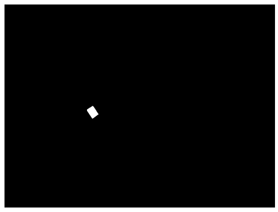
In this example, the difference image is normalized to have values between 0 and 1, for visualization. Pixels with high difference magnitudes are represented with bright colors. Similarly, when the difference magnitude approaches 0, colors representing such pixels get darker. Input images are co-registered. As it can be seen from the Figure 3.22 (a and d), image normalization is not sufficient for neutralizing the illumination variances. Even though the actual changed regions come forward in the normalized version, unchanged regions still persist as good change candidates. In order to illustrate the distribution of changed and unchanged pixels, difference magnitude values for 15 different image pairs are calculated. Values underneath the ground truth mask are treated as change pixels and vice versa. Image pairs are normalized, co-registered and each difference image is normalized to $0 - 1$ interval. It should be noted



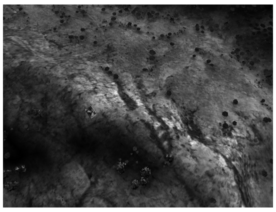
(a) Changed Image I_2



(b) Original Image I_1



(c) Ground Truth

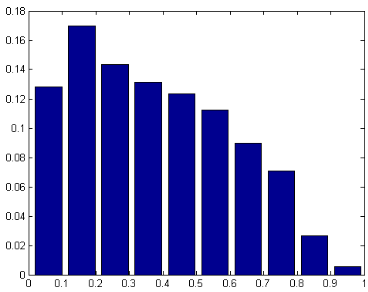


(d) Difference image without image normalization

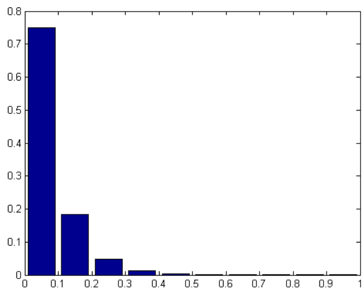
(e) Difference image with image normalization

Figure 3.22: Example Difference Images

that changed pixels constitute a smaller percent of the whole image. Therefore, number of the unchanged pixels are way greater than the changed ones, 7191989 – 8011, respectively. The bar plot of the changed and unchanged pixels can be seen at Figure 3.23.



(a) Bar plot for the changed pixels

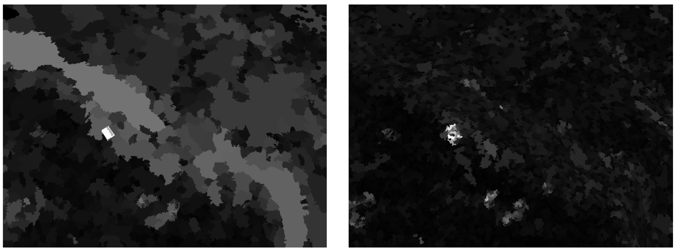


(b) Bar plot for the unchanged pixels

Figure 3.23: Pixel Difference Image Value Plots

According to the plots, unchanged pixels have generally lower values, whereas changed pixels tend to have larger ones. However, a distinction between them is still hard, because of the overlapping regions. If the high number of unchanged pixel samples is considered, many pixels in unchanged regions exhibit change characteristics. Also, most of the change pixels have low difference image values, in contrary to the assumption. Implications change positively when the segment based approach is used. In this study, instead of directly using the difference image, each input image is first segmented separately. It is assumed that the changed image, I_2 is known. Hence, changed image I_2 is segmented with the optimum parameters, parameter set 2, whereas original image I_1 is under-segmented using parameter set 1. For each segment, mean difference value is computed using the difference image. Two intensity difference segment images obtained with different segmentations are visualized in

Figure 3.24.

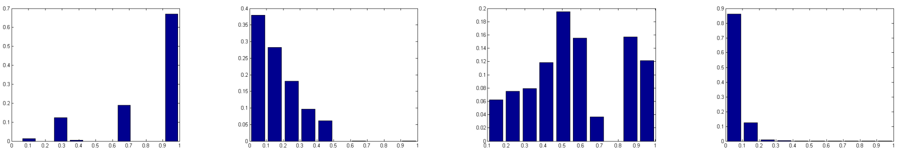


(a) L_2 Segmentation for the changed image

(b) L_1 Segmentation for the original image

Figure 3.24: Segment based difference images

First image is segmented with parameters to extract the change object. The change object represents the highest change value in Figure 3.24 (a). Some other regions, also, have high change values, but the distribution is definitely better than the simple change image. In the second image, most segments belonging to the change object have the highest change values. Most other segments do not show change characteristics. Usage of these two doubly segmented difference images for classification provides better results than the simple difference image. In order to observe the improvement caused by double segmentation method, the changed and unchanged pixel values with double segmentation are plotted and shown in Figure 3.25. L_2 corresponds to changed image segmentation and L_1 corresponds to original image segmentation. When the plot is compared to the Figure 3.23, it can be seen that the distinction of changed and unchanged pixels are accentuated in the segmented images.



(a) Changed pixels for L_2

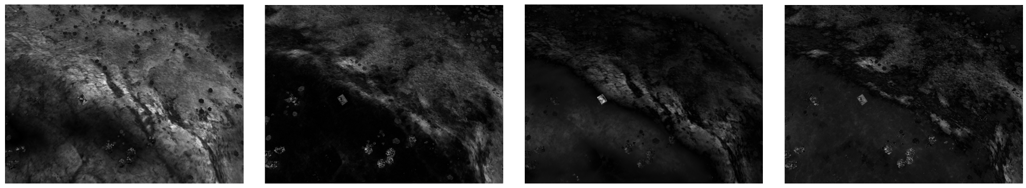
(b) Unchanged pixels for L_2

(c) Changed pixels for L_1

(d) Unchanged pixels for L_1

Figure 3.25: Changed-unchanged intensity difference value plots for segmentations

Instead of directly using the intensity values for calculating the difference image, some studies suggest the utilization of normalized values, such that for each band b_i, $b_i = \frac{b_i}{\sum_{i=1}^{n} b_i}$ [24]. This approach is also experimented and provided no qualitative and quantitative improvement. Results of such normalization are visualized and compared to the simple difference images in Figure 3.26. It is observed that, normalization process does not improve image pair for change detection.



(a) Difference image without image normalization (b) Difference image without image and with spectral normalization (c) Difference image with image normalization (d) Difference image with image and spectral normalization

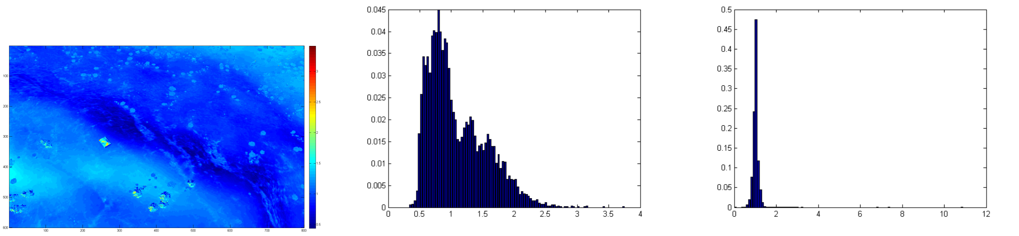
Figure 3.26: Example Difference Images with Normalization

3.4.2 Image Rationing

Another approach for computing a feature vector is the image rationing which is derived from image differencing. Intensity values for pixels are divided rather than subtracting, providing a ratio for each band of the image. Then, these band ratios can be used as separate features or can be summed. In first case, the features can be treated separately or as a feature vector in the classification step. Usage of a feature vector is more plausible since it intrinsically conceives the connections between band values. In the second case, a feature image can be calculated similar to difference image by summing and then normalizing. In such an image, values close to 1 represent unchanged regions. Pixel values that are lower or higher than 1 are likely to represent the changed regions. For images I_1 and I_2 with n bands, the ratio image R is, then, computed with normalization as:

$$R = \frac{\sum_{i=1}^{n} \frac{I_1^n}{I_2^n}}{n}. \tag{3.4}$$

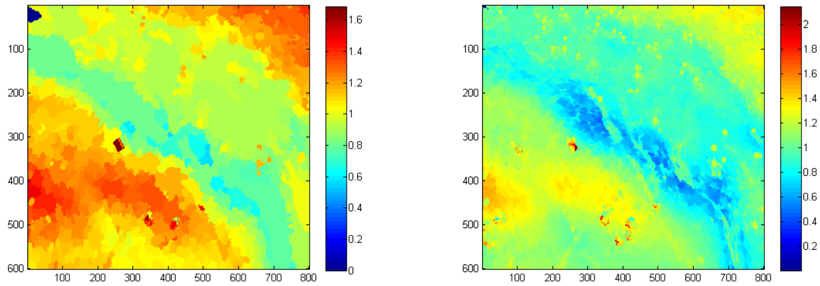
An example ratio image and the plots for pixel values are shown at Figure 3.27:



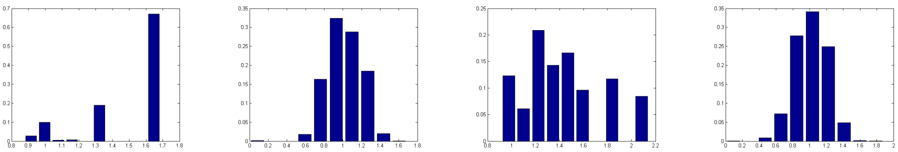
(a) Changed pixel values plot (b) Unchanged pixel values plot

Figure 3.27: Example Ratio Image and Plots

Ratio image is shown at Figure 3.27 (a). Color map is also provided. Pixel values close to 1, which are colored light blue, represent the unchanged regions. Pixel values smaller or larger than one (colored dark blue and greenish blue, green, yellow and red with respect to increasing value) are expected to be the changed regions. When both the ratio image and the plots are examined, similarities with the intensity difference values can be noticed. The distinction is still hard since changed regions, generally, do not have greater or smaller values and many unchanged pixels, also, show change characteristics. In order to examine the effects of the double segmentation method, mean ratio difference values for segments are calculated. Each image is segmented (with different parameters) and for each segment, mean ratio value is taken as the main feature. Resulting segmentation images L_2 and L_1 which depict pixel ratio values and new plots for changed-unchanged pixel ratio values are shown in Figure 3.28.



(a) L_2, Segmentation for the changed image (b) L_1, Segmentation for the original image



(c) Changed pixels for the L_2 image (d) Unchanged pixels for the L_2 image (e) Changed pixels for the L_1 image (f) Unchanged pixels for the L_1 image

Figure 3.28: Segmentation images and plots for ratio difference values

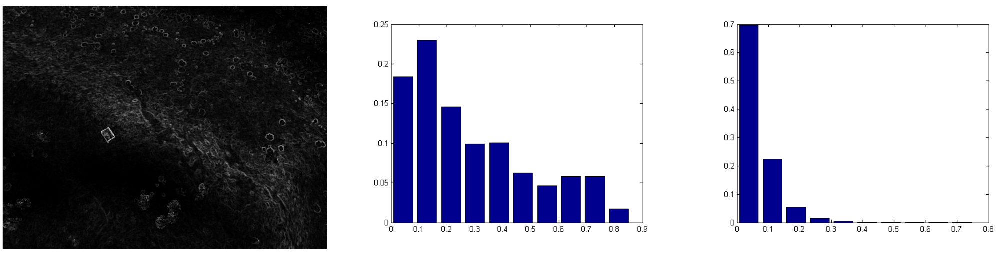
Similar to the difference image case, double segmentation method improved the discriminative power of changed regions according to the Figure 3.28. For the first segmented image, ratio difference values of changed pixels are peaked and unchanged pixels have closer values to 1 than the simple ratio differencing method. For the second segmentation, change pixels have a similar distribution with the simple ratio differencing method, but it increased the detection possibility of pixels inhibiting high change characteristics. When the ratio difference image is generally considered and evaluated as a feature with this evidence, no improvement and decline over the intensity difference image can be seen. Like other features explained in this section, its contribution to the change analysis process will be further analyzed in the following chapter.

3.4.3 Change Vector

Another widely used feature for change detection is the change vector. CVA is utilized when it is desired to classify the type of the change. Change vectors are generally defined with 2 properties, their magnitude and direction. Direction of the change vector designates the type of the change. In the scope of this study, it is not possible to define the type of changes with direction. A newly built or destructed man made object can create a change vector with any direction, depending on the spectral values of the region its built or destructed on. When the magnitude of the changed vector is used as the only feature, the feature is basically reduced to the intensity difference image. Hence, change vectors are not utilized in this study.

3.4.4 Texture

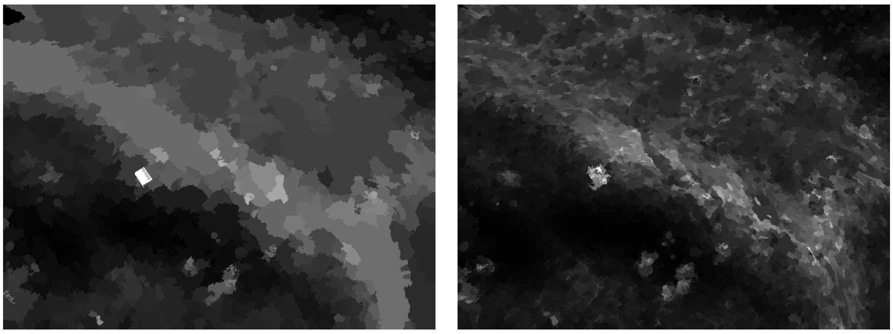
Texture based features are generally used for object detection in the remote sensing literature. If the searched object shows a distinctive visual pattern, features representing this pattern can be computed and utilized for detection. Generally, such an approach is supervised and uses a training set to specify those features. Texture based features can be also utilized for change detection. In order to define such a feature, the change object must exhibit distinctive visual texture characteristics. In this study, possible change objects do not inhibit such characteristics and can indicate any texture pattern. Therefore, it is not possible to compute and use such features for change analysis. However, instead of using specific texture patterns, local spectral values of the images can be inspected to detect such patterns in an unsupervised way. In order to detect such patterns, local ranges of the images are computed using a range filter. Range filter utilized in the study works on a $3x3$ neighborhood and returns the value *maximum - minimum* with respect to the neighborhood pixel intensity values. When an object is present at the scene, such a filter gives strong edge responses due to spectral differences between the object and its surroundings. In addition to this, unchanged regions with local illumination changes are expected to give weaker responses. Such a property can eliminate false classifications caused by local illumination variances. Range difference image acquired by the subtraction of range filter outputs and the distribution of the changed and unchanged pixel range filter difference values are shown in Figure 3.29.



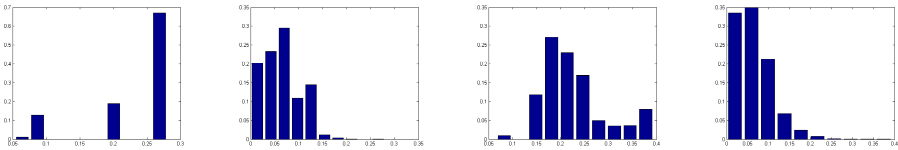
(a) Range difference image (b) Changed pixel values plot (c) Unchanged pixel values plot

Figure 3.29: Example Range Filter Difference Image and Plots

As it is seen from Figure 3.29, the edge pixels of change objects give the highest response. Unchanged regions affected from illumination variances also give responses, but they are less than the actual change object. Plot for the unchanged pixel values are similar to former feature plots, generally having small values instead of some outliers. However, plot for the pixels belonging to the changed object have many undesired low values. It is expected since all the pixels forming the object is considered in the plot but only edge pixels have high responses. The effects of double segmentation on the feature is also inspected. Segment range difference images and their value plots are given at the Figure 3.30.



(a) L_2, Segmentation for the changed image (b) L_1, Segmentation for the original image



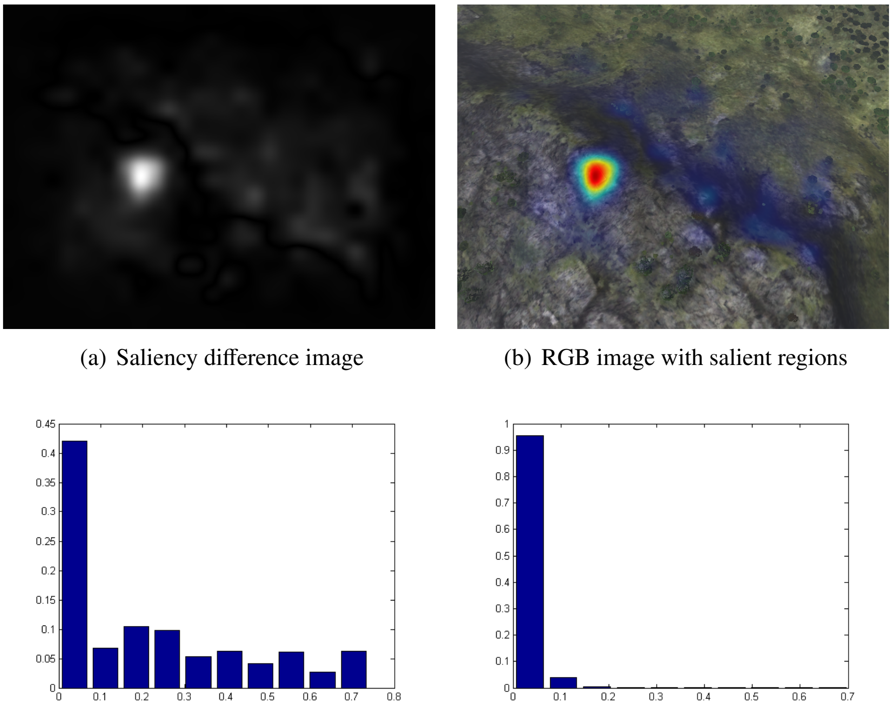
(c) Changed pixels for the L_2 image (d) Unchanged pixels for the L_2 image (e) Changed pixels for the L_1 image (f) Unchanged pixels for the L_1 image

Figure 3.30: Segmentation images and plots for range filter difference values

In L_2, the high response of the edge pixels are distributed across the whole object and the distribution of change pixels seem more discriminating. However, this scheme is not guaranteed for change objects with larger areas. For a larger change object, the change information present in the edges can be suppressed by the overwhelming number of pixels inside the object. Also, unchanged pixels generally have higher values than they had on the difference segmentation images. Information obtained from the L_1 plot does not make the situation better. Segments that include a portion of edge pixels have higher values, other segments inside the object have low values. Moreover, many unchanged pixels show change characteristics, making the discrimination harder. However, range difference image still provides essential information about the boundaries of the changed region. For better results, it can be used on pixel basis or a more efficient approach for distributing the edge values across the object can be utilized. Double segmentation is also applicable when the object sizes are not large enough.

3.4.5 Saliency

Man made objects in remote sensing images are generally visually salient. Detection of such objects before or after change analysis phase may improve the results. Such information can be used to eliminate false changes after classification. Also, saliency responses can be used as features in the classification step. For computing a visual saliency map, GBVS [31] method is used. In this study, color, orientation and contrast values for a region is used to compute saliency values. In order to compute a difference feature based on saliency, both images saliency maps are extracted. Saliency map responses can differ with respect to illumination changes so both saliency maps are normalized using image normalization. Then, these maps are subtracted to form a saliency difference map. Example saliency difference map, its visualization on RGB image and the distributions for changed and unchanged pixel values can be seen at Figure 3.31.



(a) Saliency difference image

(b) RGB image with salient regions

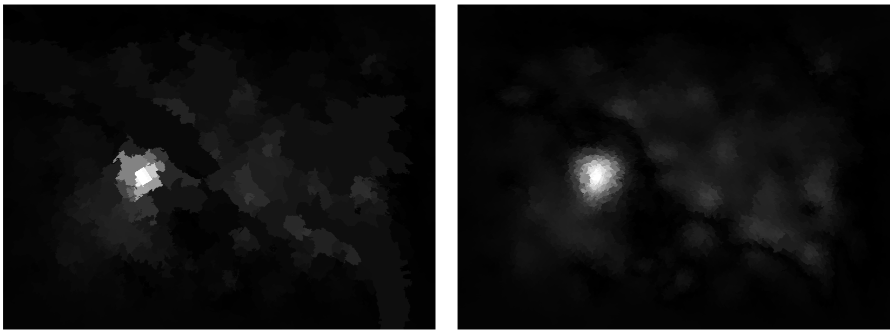
(c) Changed pixel values plot

(d) Unchanged pixel values plot

Figure 3.31: Example Saliency Difference Images and Plots

When the grayscale image and the RGB image are analyzed together, it can be clearly seen that this feature is highly discriminative. Changed regions in image clearly have higher responses than any other ones, including regions affected from the illumination change. However, there are two drawbacks that need to be addressed when utilizing this feature. Firstly, because of the intrinsic definition of the saliency, pixels surrounding highly salient regions also have higher values. This property may cause unchanged regions, which are close to the actual change area, to be marked as false changes. Second problem arises when the distributions are inspected. Saliency give perfect results for this example but when the whole subset with 15 images are considered, results are not so pleasant. Some change objects present in those images do not have salient characteristics and this can be generalized to a bigger dataset. Many changed regions have low saliency values, lower than %40 according to the plot of Figure 3.31(c). This is caused by change objects with low salient characteristics. Considering these two problems, saliency can not be used as the only feature. Double segmentation procedure is also applied to the saliency difference images and its results can be seen at Figure 3.32.

According to these segmentation results, the procedure does not greatly improve the saliency



(a) Segmentation for the changed image (b) Segmentation for the unchanged image

Figure 3.32: Segmentation images for saliency difference

difference image. However with L_2, the problem with the surrounding pixels are reduced. Since the procedure does not produce any unfavorable results, it is utilized in the proposed method.

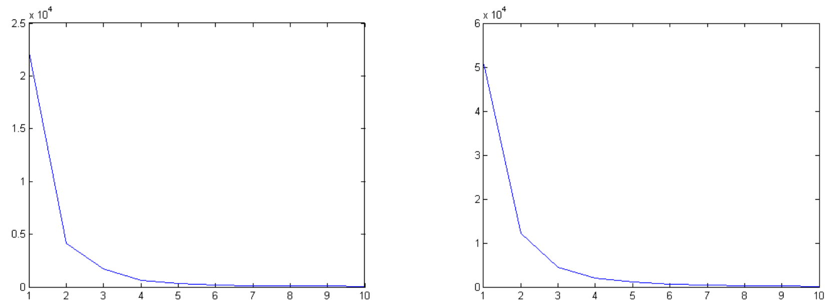
3.5 Labeling the Change

In the scope of machine learning and statistics, classification can be defined as the process of identifying an observations belonging to a possible category, class, with respect to its explanatory variables, features. In this study, two classes are present; changed and unchanged. This reduces the process to two-class classification problem. Since there are no training sets present for all datasets and it is not possible to create a training set for all possible type of changes, unsupervised approach is selected in this study. Three different unsupervised approaches are used distinctively for labeling the changes and compared, namely K-Means clustering, heuristic thresholding and Expectation-Maximization.

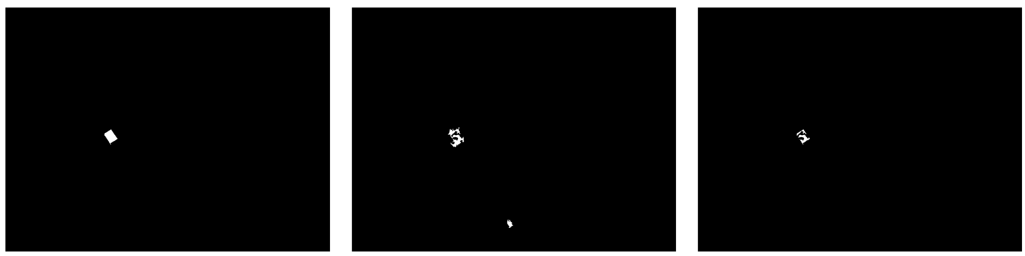
3.5.1 K-Means Clustering

Unsupervised clustering is the process of grouping sample points into clusters, according to similarity of their features. There exist different clustering algorithms in literature, based on different theoretical models. In this study, K-Means, which is a centroid based, hard clustering algorithm is chosen for the classification step. In many clustering algorithms, the main drawback is finding the appropriate number of clusters. Theoretically, in change analysis, there must be two clusters that represent changed and unchanged samples. However, in application, two clusters generally do not produce expected results. Appropriate number of clusters then should be found by inspecting the quality of clustering results for different number of clusters. K-means can be used separately with different features or with their combinations. Features explained in the former section are used for classification. All features (intensity difference values, ratio difference values, range filter difference values and saliency map difference values) are transformed to have highest values in case of a change. Samples belonging to the cluster with the highest valued centroid are chosen as the changed regions. Instead of using the whole pixels from the image as samples, mean values computed for segments are used in the classification. This decision reduces the run-time of the proposed algorithm and provides better results. Image pair shown in Figure 3.32 and its calculated features are used for classification.

In K-Means clustering, k represents the number of clusters. There exist different methods in the literature for finding the appropriate number of clusters for a dataset. Most basic approach which is known as the *rule of thumb* is directly bound to the number of samples. According to this approach, appropriate number of clusters is $\sqrt{n/2}$ where n is the number of samples. Although it produces good results in some experiments, it is not statistically correct on most conditions and can produce unwanted results. Another method is known as the elbow method. This method tries to detect the number of k where adding another cluster does not improve the modeling of the data. In this study, this method is implemented by measuring the within class variance for different values of k. For each k, average distance of samples to their class centroids are computed and plotted. This value is expected to drop by the inclusion of new clusters. When this decrease drops under a certain threshold, it can be decided that the proper number of clusters are found. For the mean intensity difference values for segments shown in Figure 3.24, the plots and resulting change segments are shown in Figure 3.33. The average distance values are computed for $k = 1 : 10$.



(a) Average distance plot for L_2 values (b) Average distance plot for L_1 values

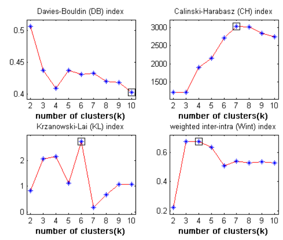


(c) Changed segments for L_2 (d) Changed segments for L_1 (e) Fused Change Image

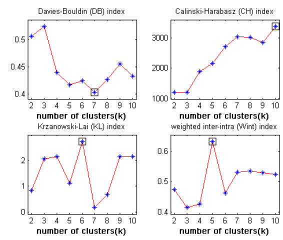
Figure 3.33: K-Means elbow method plots and outputs

According to the plots of Figure 3.33 (a,b), the best k value can be chosen as 5. Resulting two change maps with that k value are then intersected to produce the final change result. The problem with this method is that the *elbow point* can not always be identified unambiguously.

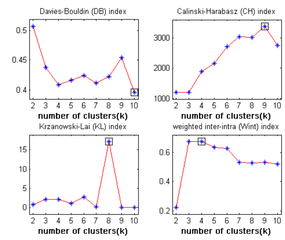
In this example, the k value can be chosen as 4 as well as 6 or 7. Such different choices greatly affect the results. In order to overcome such issues and to define a more robust (and fully automatic) process, some indexes that define the quality of clustering results are facilitated. These indexes are Davies-Bouldin (DB), Calinski-Harabasz (CH), Krzanowski-Lai (KL) and Weighted Inter-Intra Ratio (Wint) indexes which are explained in Section 2.5. These indexes are generally very sensitive to random initialization of K-Means procedure and can produce different results for different runs. Three different runs for $k = 1 : 10$ are made for visualization and plots for resulting index values are given at Figure 3.34. Difference values for L_2 are used. The best choices for k with respect to corresponding indexes are shown in the plots with a rectangle.



(a) Run 1



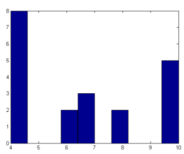
(b) Run 2



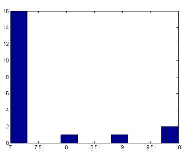
(c) Run 3

Figure 3.34: Index value plots for 3 different runs

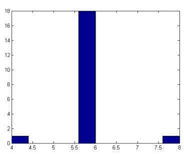
As seen from Figure 3.34, the best choice for k differs at the runs, because of the random initialization. This can produce unwanted results and reduces the robustness of the algorithm. In order to overcome this issue, the computation can be also made iteratively. Instead of computing the index values for different k values for once, computation can be made in numbers. Most frequent number of best k can then be chosen. It is observed that the best choice for k is generally the most frequent one. In order to validate this assumption, the histogram plot for best k values obtained at 20 different runs are plotted in Figure 3.35.



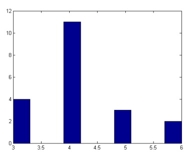
(a) DB values



(b) CH values



(c) KL values



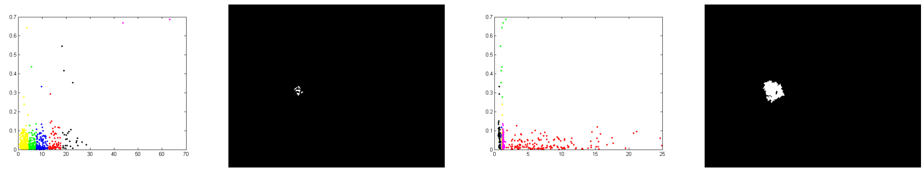
(d) Wint values

Figure 3.35: Histogram plots for best k values

According to Figure 3.35, most frequent k values for DB, CH, KL and Wint are found as 4, 7, 6 and 4, respectively. Selection of the validity index definitely effects the results of the change detection algorithm and it is inspected in detail at the following chapter.

Another issue with K-Means is the choice of features. The aforementioned features can be used in combination when using K-Means. As an example, the feature vector is created with the mean saliency map and intensity difference values for segments. Number of clusters, k

is taken as 6. In an another run, the feature vector consists mean saliency map and ratio difference values for segments. Plot for the samples in L_2 and the resulting change maps are shown in Figure 3.36.



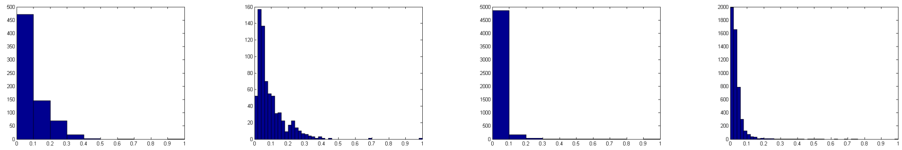
(a) Mean difference and saliency samples (b) Resulting change map (c) Mean ratio and saliency samples (d) Resulting change map

Figure 3.36: Histogram plots for best k values

In both plots, Y axis corresponds to the saliency map values. X axis shows the intensity difference and ratio difference values, respectively. Each sample is colored according to its cluster. In first plot, the changed samples are colored in pink and in second the color is green. This example shows the differences caused by the different feature vectors. Different features and their combinations are inspected in the following chapter.

3.5.2 Heuristic Thresholding

Thresholding is a widely used method in change detection and remote sensing literature. For the calculated feature values, a threshold is defined and samples that are not compatible with this threshold are discarded to form the changed regions. Hard thresholding, where a threshold is defined for all examples is the most basic thresholding technique. It can produce good results when the calculated features are guaranteed to have similar characteristics. For example, in vegetation detection problem, a hard threshold can be defined on the calculated index values, since vegetation regions have similar characteristics in general. However, when dealing with changed regions, this approach is not always applicable. Many features can be defined for identifying changed regions and all those features have different characteristics, making it impossible to define a hard threshold that complies with all of them. Apart from different features, same feature can produce different results on different image pairs, making the predefined threshold invalid. The best way to cope with this problem is to define a procedure that will find an appropriate threshold with respect to the distribution of the samples. Regardless of the feature, most samples in change detection application have similar properties. It is even easier to define a such process in the scope of this study, since it is assumed that the changed regions have small areas. Hence, it is expected that most of the samples represent unchanged regions and show similar characteristics whereas samples belonging to changed regions emerge as possible outliers. Therefore, in theory, samples belonging to changed and unchanged regions should form distinct groups that are distant to one another. In such a case, changed regions can be easily detected by visual inspection from the histogram plots of samples. The intensity difference value histograms for the image pair shown at Figure 3.22 are shown at Figure 3.37. Different sized bins are used for the plots. Samples are normalized to have values between 0 and 1.



(a) Mean difference for L_2 with 10 bins (b) Mean difference for L_2 with 50 bins (c) Mean difference for L_1 with 10 bins (d) Mean difference for L_1 with 50 bins

Figure 3.37: Histogram plots for mean segment intensity difference values

It can be clearly seen from the histogram plots at Figure 3.37 that an appropriate threshold for values obtained from L_2 will be closer to 0.7 and for L_1, it will be closer to 0.6. In order to find such thresholds automatically, a method that works on histogram of feature values is proposed. The steps of the proposed algorithm is specified as:

1. Normalize the sample values between interval $0-1$.
2. Compute the histogram of normalized values with n bins.
3. Find the peak values of the histogram.
4. Introduce a hard threshold t_1 as 0.5.
5. Find the minimum valued peak p such that $p>t_1$.
6. Assign the threshold t that defines the changed regions with a buffer as p-0.05.

The hard threshold value, t_1, introduced in step 4 is based on a simple assumption. Changed segments should have higher difference values than the fifty percent of all samples. This assumption is then validated empirically. Using the threshold values found with the proposed procedure, changed regions can be classified. It is expected that the changed regions have samples with discriminating values. Therefore, any peak with a value smaller than the predefined threshold at step 4 are discarded. Only samples with relatively higher values are considered in this method. The buffer zone in step 5 is introduced mainly for higher number of bins, to include samples closer to the threshold. Proposed method has 2 main drawbacks. Firstly, it works on one specific type of feature. Features can be fused together, but it requires to find optimal weights. Resulting change maps from different features can be intersected. However, it eliminates the relevance of features in the classification stage. Secondly, algorithm can fail to find a threshold if the values are continuous. This problem can be slightly handled by defining another hard threshold for such cases. Change maps for mean intensity difference values and saliency map difference values obtained with different bin sizes are shown at Figure 3.38.

Results from Figure 3.38 shows that the bin size drastically effects the results of the heuristic thresholding. For the intensity difference values, increasing the bin size causes the elimination of segments actually belonging to changed regions. In contrast, increase of bin size for

(a) Change map with difference values and 10 bins (b) Change map with difference values and 50 bins (c) Change map with saliency values and 10 bins (d) Change map with saliency values and 50 bins

Figure 3.38: Change maps obtained with heuristic thresholding

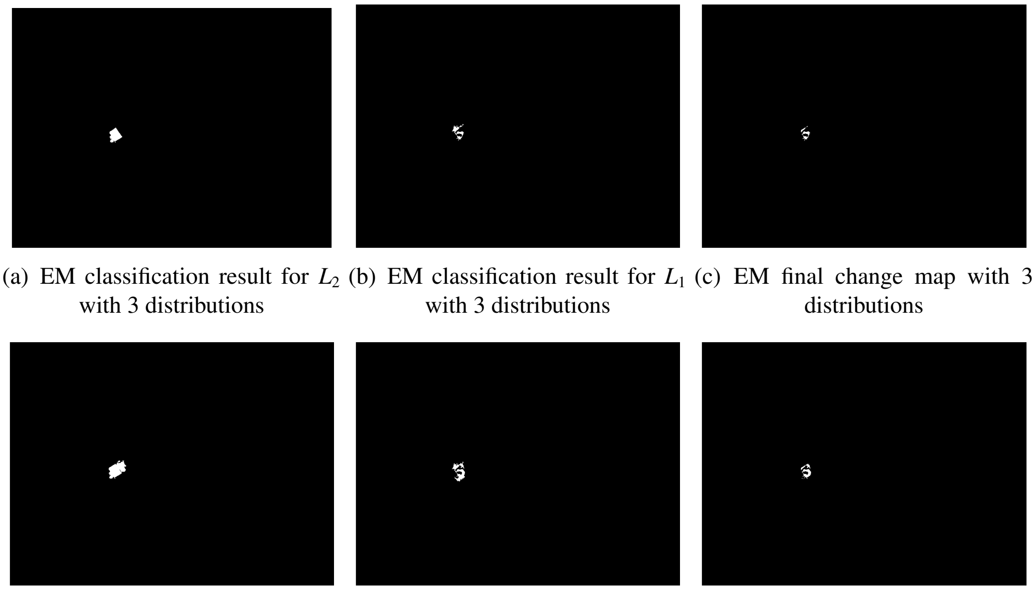
saliency values suppresses the false alarms. Utilization of different features with this technique is elaborated in the following chapter.

3.5.3 Expectation-Maximization

Another method used in the literature for clustering the data and therefore classifying the changed samples is the utilization of Expectation-Maximization (EM) algorithm. By using the EM algorithm, the classification problem can be treated as a statistical hypothesis test. The sample values can be assumed to be derived from distributions with different parameters. The change detection process would be relatively easy if the parameters of such distributions are known. In this study, these parameters are unknown. However, they can be detected by statistical methods in the literature [11]. EM can predict the maximum likelihood estimates of such parameters. Using these parameters, different classes can be distinguished from one another and changed regions can be identified.

The sample values obtained from aforementioned features can be treated as Gaussian Mixture Models with n components. Then, EM can be utilized to find the parameters of n distinct Gaussian components in the mixture. Each sample can then be checked for its belonging to the component that describes the change. Also, the risk factor for the Bayesian Decision Process can be included in the process. This approach can be employed to a feature vector of various sizes, similar to K-Means. It treats these features as the dimensions of the distribution that the sample is derived from. The main drawback, similar to K-Means, is the identification of the number of different components inside the density function. Choice of different number of components can result in different results. An example classification procedure is held for the image pair shown at Figure 3.22 with the combination of mean intensity difference, mean ratio difference and mean saliency map difference features. 3 and 5 components are assumed to be present in the mixture for different runs. Figure 3.39 shows the change maps for L_2 and L_1 and final change maps. The risk factor in this run as taken as 5.

Even though the final change maps look similar, when the intermediate results are inspected, it can be seen that the choice of number of components effects the results. These effects will be inspected in detail in the following chapter.



(a) EM classification result for L_2 with 3 distributions (b) EM classification result for L_1 with 3 distributions (c) EM final change map with 3 distributions

(d) EM classification result for L_2 with 5 distributions (e) EM classification result for L_1 with 5 distributions (f) EM final change map with 5 distributions

Figure 3.39: Change maps obtained with EM

3.6 Post Processing for Segment Reconstruction, Elimination and Saliency Analysis

This section elaborates the post-processing techniques used for enhancing the obtained change map. Inadequacies in a change map can be discussed in two topics, insufficiently labeled change objects and false positives. A two phase postprocessing procedure which addresses these issues is presented. The steps of the postprocessing are given as:

1. **Segment Elimination**

   (a) For every connected component *cc* in the final change map;

   (b) Find the intersecting segments l_1 and l_2 in L_2 and L_3.

   (c) Measure the similarity of l_1 and l_2.

   (d) If the similarity of l_1 and l_2 is higher than a predefined threshold, discard *cc* from the final change map

2. **Segment Reconstruction**

   (a) For every connected component *cc* in the final change map;

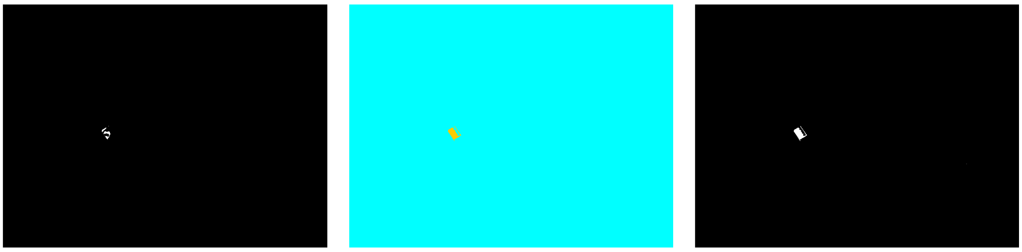
   (b) Find the intersecting segments l in L_2.

   (c) Add l to the final change map.

Each step given here are explained in detail in the following subsections. The metric and threshold values used in the Segment Reconstruction step are, also, defined.

3.6.1 Segment Reconstruction

When the sample change maps shown in the former section are examined, incomplete change objects can be seen. The main drawback of the double segmentation method is the possibility of missing parts in the changed regions. Since the final change map is formed by the intersection of two different intermediate change results, which are obtained with different segmentations, the resulting change regions generally have missing parts. However, this can be easily overcame with an assumption of prior knowledge. In the study, L_2 represents the segmentation result for the changed image, I_2. The connected components in the change map can be compared with the segments in L_2 and intersecting segments can be fetched to obtain a more solid and visually attracting change map. An example of such process is shown in Figure 3.40.



(a) Final change map (b) Intersecting segments in L_2 (c) Segment retrived change map

Figure 3.40: Segment Retrieval

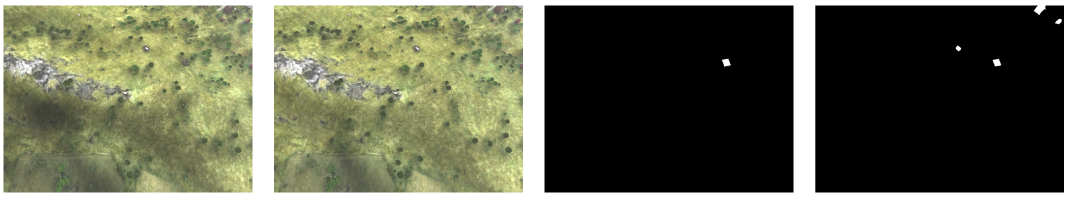
Segments in L_2 that intersect the final change map are shown in yellow color in Figure 3.40 (b). These segments then form the final change map. There may exist some gaps in the change map, but generally such cases can be handled by basic morphology operations. For actual changed regions that are not present in the change map, no post-processing method can be proposed. Therefore, this step only aims at enhancing the detected changes.

3.6.2 Segment Elimination

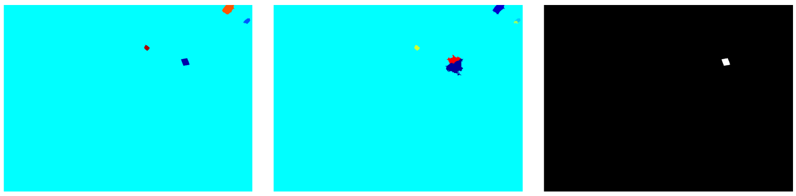
Second problem is mislabeled regions as change. Generally, due to local illumination changes, unchanged regions can be labeled as changed in the classification process. Image pairs that can produce such errors are shown at Figure 3.2 and 3.3. In order to eliminate such regions, segmentation results L_2 and L_3 are utilized. As a reminder, L_2 and L_3 are the segmentation results obtained with the same parameter set for images I_2 and I_1, respectively. Ideally, unchanged regions should be formed by similar segments in both images, regardless of the illumination changes. Such an assumption is not always true in application. Even in highly similar images, segmentation results tend to differ. If unchanged regions do not show any specific characteristics, like Figure 3.2, segments constituting them will be different. However, if the illumination change effected objects intensity values, like in Figure 3.3, segments for them in both images will be similar. This assumption can be used to detect false changes in change maps. For every connected component in the change map, its corresponding segments in L_2 and L_3 are compared. If these segments are more similar then a predefined threshold of similarity, segment is discarded from the final change map. For measuring the similarity of segments, a basic heuristic is defined. For any segment pair $[l_1, l_2]$ where $l_1 \in L_2$ $l_2 \in L_3$, the ratio R defining the similarity of segments are defined as:

$$R = \frac{l_1 \cap l_2}{(l_1 \cap l_2) + ((l_1 \cup l_2) - (l_1 \cap l_2))} \tag{3.5}$$

where dividend is the number of intersecting pixels and divisor is the number of non-intersecting pixels in l_1 and l_2. After computing R, an appropriate threshold t should be defined. Then, segment pairs with R value higher than t can be eliminated from the change mask. For exactly same segments, the similarity measure is $R = 1$. An example application of the mentioned method can be seen at Figure 3.41. Threshold value t is taken as 0.8 in this example.



(a) I_2 (b) I_1 (c) Change GT (d) Change map



(e) Intersecting segments in L_2 (f) Intersecting segments in L_1 (g) Final Result

Figure 3.41: Segment Elimination

When the segmentation results from Figure 3.41 are examined, the structural similarity of segments belonging to unchanged regions can be clearly seen. However, segments forming the changed regions in image pairs are completely different. Using the aforementioned approach, such segments can be eliminated from the change map. This elimination can be also done before feature extraction step. This way, a more reliable feature extraction and classification step can be done. Another method for eliminating the false changes is the usage of saliency. When the changed image is known, its saliency map can be utilized for detecting false changes. The saliency map can be thresholded to point salient regions only. Then, non-salient pixels in the change map can be discarded. Change objects are guaranteed to be salient in theory. However, this assumption is not always true in practice. In many experiments, some changed objects are not marked as salient regions. Therefore, such an approach can eliminate correctly found changed regions. Instead of using saliency for post processing, using it as a feature with combination of others seems more plausible.

CHAPTER 4

EXPERIMENTAL ANALYSIS OF THE DOUBLE SEGMENTATION CHANGE DETECTION FRAMEWORK

In this chapter, results of the suggested change detection method are discussed under a variety of configurations of feature sets and classification methods. Four different features are utilized in the experiments; intensity difference, ratio difference, range filter difference and saliency difference values. Three different classification methods mentioned in the former chapter are used for labeling changed regions. These classification methods are K-Means clustering, heuristic thresholding and Expectation-Maximization. For K-Means and EM, 4 features and their combinations are used as inputs, resulting in 15 different combinations. However, heuristic thresholding can be applied on a single dimensional feature vector. As a result, features are compared to each other when heuristic thresholding is utilized. The chapter is organized in three sections. First, the performance metrics used in the study are explained. In the second section, experiments done with different configurations are compared. Experiments are done with two different datasets; AICD [7] and real world satellite images obtained from Google Earth and SZTAKI Airchange dataset [3]. AICD dataset includes computer generated co-registered aerial image pairs and ground truth information is present. Ground truths for the real dataset are not avaliable. Therefore, quantitative results are given only for the AICD dataset. Extensive experiments with various classification methods and feature sets are done only on the AICD dataset. Using the implications obtained from the experiments on AICD dataset, visual inspection for real world satellite images are also provided. In the final section, three change detection algorithms [11, 14, 7] are compared with the proposed method using AICD dataset.

4.1 Performance Metrics

The output of the proposed algorithm is a binary mask, where changed regions are represented by pixel value 1. Ground truth masks also have the same convention. For comparing the results with the ground truth mask, Precision-Recall metric which is extensively used in the object detection literature is used. Precision and Recall values obtained from a pair of sample images are then combined using the F-measure metric. 4 different indicators are computed to evaluate the performance in a pixel basis:

- **Number of True Positives (TP):** This indicator is the number of pixels that are both present in the result and ground truth mask. In other words, this is the number of pixels correctly classified as change.

- **Number of False Positives (FP):** This indicator is the number of pixels that are present in the result mask, but not present in the ground truth mask. Therefore, this is the number of pixels falsely classified as change.

- **Number of True Negatives (TN):** This indicator is the number of pixels that are not present in both result and ground truth masks. TN is the number of pixels correctly classified as unchanged.

- **Number of False Negatives (FN):** This indicator is the number of pixels that are not present in the result mask, but are present in the ground truth. So, FN is actually the number of pixels misclassified as unchanged.

These terms are tabulated at the Table 4.1. Actual class represent the ground truth information and observed class is the labeling obtained after the classification.

Table 4.1: Terms used for Precision-Recall Metric Calculation

Actual / Observed	Changed	Unchanged
Changed	True Positive (TP)	False Positive (FP)
Unchanged	False Negative (FN)	True Negative (TN)

However, definitions in Table 4.1 for these terms are pixel based. Such an evaluation method can be plausible when working with scenes including large change objects. However, when working with smaller objects, this evaluation method can cause biased evaluations. AICD dataset includes many samples that have change objects with small areas. Also, proposed algorithm works on a segment basis rather than pixels. Therefore, a compatible evaluation criteria must be used in the performance assessment. When a large portion of a changed object is found, the evaluation criteria should treat this object as correctly classified. Similarly, small unchanged regions classified as change should not be treated as false positives, if they are covering a change area. An object based performance evaluation criteria that fulfills such necessities is proposed in [7]. In this approach, connected components in ground truth and result masks are used for computing TP, FP and FN values. A binary ground truth mask G can be divided into i connected components, g_i. Likewise, result mask R can be described with its connected components, r_j. Using these definitions, the terms for each connected components

i and j can be computed as:

$$TP_{g_i} = \begin{cases} |g_i|, & \text{if } \frac{|g_i \cap \bigcup_j r_j|}{|g_i|} > \frac{1}{5} \\ 5 * |g_i \cap \bigcup_j r_j|, & \text{else} \end{cases} \tag{4.1}$$

$$FN_{g_i} = \begin{cases} 0, & \text{if } \frac{|g_i \cap \bigcup_j r_j|}{|g_i|} > \frac{1}{5} \\ |g_i| - 5 * |g_i \cap \bigcup_j r_j|, & \text{else} \end{cases} \tag{4.2}$$

$$TP_{r_j} = \begin{cases} |r_j| - |r_j \cap \bigcup_i g_i|, & \text{if } \frac{|r_j \cap \bigcup_i g_i|}{|r_j|} > \frac{1}{5} \\ 4 * |r_j \cap \bigcup_i g_i|, & \text{else} \end{cases} \tag{4.3}$$

$$FP_{r_j} = \begin{cases} 0, & \text{if } \frac{|r_j \cap \bigcup_i g_i|}{|r_j|} > \frac{1}{5} \\ |r_j| - |r_j \cap \bigcup_i g_i|, & \text{else} \end{cases} \tag{4.4}$$

where i and j are the ids of the current connected components in the ground truth and result mask, respectively. According to this convention, TP value is calculated distinctively for the ground truth and result mask. If a connected component in the ground truth mask intersects with the result mask with a ratio equal or higher than 0.2, total number of pixels inside the connected component are added to TP_{g_i} (Equation 4.1). If the intersect ratio is lower, number of intersecting pixels multiplied by 5 are added to the total number of true positives (Equation 4.1). In the same configuration, if the ratio is high, number of false negatives for this connected component are taken as 0 (Equation 4.2). In a case of low ratio value, number of non-intersecting pixels are added to false negatives (Equation 4.2). When computing the true positive values for the connected components in the result image, the ratio value is defined similarly. If the ratio is equal to or higher than 0.2, all non-intersecting pixels in the connected component are considered as true positives (Equation 4.3). Intersecting pixels are not added to true positives, since they are expected to be added when considering the connected components in the ground truth mask. In the case where the ratio is smaller, intersecting pixels are still added to the true positives by multiplying with 4. Number of false positives are computed in a similar manner. If the intersection ratio is smaller than 0.2, non-intersecting pixels are added to the false positives, elsewhere, non-intersecting pixels are discarded (Equation 4.4). The hard threshold, 0.2, which is used in the performance metric is defined in [7]. In order to make a healthy comparison, same threshold is selected in this study. With this configuration, the following constraints are satisfied when all connected components in ground truth (n_1) and result mask (n_2) are considered:

$$\sum_{i=1}^{n_1} TP_g(g_i) + FN_g(g_i) = |G|, \tag{4.5}$$

$$\sum_{j=1}^{n_2} TP_r(r_j) + FP_r(r_j) = |R| - |R \cap \bigcup_i g_i|. \tag{4.6}$$

Using these newly defined terms, Precision and Recall values for a result then can be computed. Precision value is the fraction of true samples in the result and can be computed as;

$$Precision = \frac{TP}{TP + FP}. \tag{4.7}$$

A discussion emerges when computing Precision if $TP = 0$ and $FP = 0$. It will result in the operation $0/0$ and will be meaningless with respect to definition of Precision. If no samples are retrieved, what will be the fraction of true samples? In [7], where the performance metric used in this study is defined, Precision value is taken as 1 in such a case. However, this is considered to be relatively prejudiced. In this study, Precision value is not computed and included in average scores in such a case. Fraction of the true samples that are retrieved is the Recall metric and its computation is given at Equation 4.8.

$$Recall = \frac{TP}{TP + FN}. \tag{4.8}$$

When $TP = 0$ and $FP = 0$, Precision value is not computed as mentioned formerly. Recall value is computed as 0 in such a case and included in averages. These two metrics are sufficient for assessing the performance of a change detection result. However, for simplicity, these two measures are combined into one metric, F-measure. F-measure is the harmonic mean of the Precision and Recall values. This metric gives an overall performance, aggregated version of precision and recall. F-measure is computed as:

$$F = \frac{2 * Precision * Recall}{Precision + Recall}. \tag{4.9}$$

In every experiment with ground truth information, Precision and Recall values are computed. F-measure are then computed using these values. F-measure results are inspected for assessing the overall performance of the algorithm. Precision and Recall values are also analyzed separately.

4.2 Experiments Applied on Remote Sensing Data

This section is organized in three subsections. Firstly, extensive experiments are performed on a subset of the AICD dataset to understand the behavior of the proposed feature sets and classification algorithms. This AICD subset includes 15 image pairs. Images in the subset are selected to have different characteristics such as size, type and illumination. This way, significant features in the whole dataset can be encapsulated in a small scale. Several tests are performed for evaluating the contribution of different components to the proposed algorithm. Results obtained with the selection of different features and classification algorithms are compared. Also, contribution of the proposed post-processing algorithm to the results is discussed. In the second subsection, the promising combinations of features and parameters for each classification method are selected according to the analysis of the former subsection. Then, quantitative and qualitative results are obtained for the whole AICD dataset. In the final subsection, optimal configurations of the feature sets and classifiers are applied on the real world satellite images and qualitative results are represented. When performing the method on real world images, some parameters are also optimized to suit the images.

4.2.1 Experiments Performed on the AICD Subset For Feature and Method Evaluation

AICD dataset is created for benchmarking different change detection methods in the literature [7]. It consists artificially created co-registered image pairs. This property of the dataset eliminates differences caused by registration. It evaluates the performance of the actual change detection procedure. A subset of image pairs is selected for extensive experiments. This subset includes 15 image pairs with different image pairs, each depicting a different type of change object. In all experiments, images are normalized to have intensity distributions with same mean and covariance values. In the former chapter, 4 different features are defined for change analysis. In this chapter, these features and their different combinations are inspected for their contribution to change analysis. Acronyms will be used when referring to these features and their combinations. Those acronyms are:

- **D:** Mean Intensity Difference Values for Segments
- **R:** Mean Ratio Difference Values for Segments
- **F:** Mean Range Filter Difference Values for Segments
- **M:** Mean Saliency Map Difference Values for Segments

These features will also be used in combination as feature vectors in some classification methods. These acronyms are merged to represent such combinations. For example, the feature vector *[Intensity Difference, Ratio Difference]* will be represented by **DR**. Three different classification methods are mentioned in the previous chapter which are K-Means clustering, heuristic thresholding and Expectation-Maximization. Each method is inspected with different feature sets and parameters for detecting the best configuration. Also, the effects of the post-processing on the final result are analyzed for each method.

4.2.1.1 Experiment Setups

Each proposed classification method has its own parameters and feature combinations. Therefore, different number of runs are made for each image pair in the AICD subset to detect satisfying combinations. In each run, Precision, Recall and F-Measure scores are computed for resulting raw and postprocessed change masks of 15 image pairs in AICD subset. For every configuration, average Precision, Recall and F-Measure scores of 15 images are computed and compared. Feature sets and parameters for 3 classification methods are tabulated in Table 4.2.

According to Table 4.2, when using K-Means clustering, 60 different change masks are computed for every image pair. 15 different feature sets are created as possible combinations of 4 features; D, R, M and F. Four different validity indexes, DB, CH, KL and Wint, which are described in Chapter 2, are used for finding the appropriate cluster number. Therefore, for every

Table 4.2: Configurations used in the AICD Subset Experiments

	Feature Sets	Parameters	# of Runs
K-Means	15 Combinations of Features	4 Validity Indexes (DB, CH, KL, Wint)	60 Runs per Image Pair
Heuristic Thresholding	4 Separate Features (D, R, M, F)	3 Histogram Bin Numbers (20, 50, 80)	12 Runs per Image Pair
Expectation-Maximization	15 Combinations of Features	3 Mixture Components (3, 4, 5)	45 Runs per Image Pair

feature set, 4 different results are obtained with respect to validity index selection, resulting in 60 different runs. Heuristic thresholding obtains 12 different change masks for each pair. For every feature selection, 3 different change masks are obtained with respect to histogram bin number. Similar to K-Means, EM method uses 15 different feature sets. 3 different change masks are obtained for every image pair with respect to different selections of number of components in the mixture. In other words, EM calculates 45 different change masks for image pair. Also, every raw change mask is postprocessed and performance is evaluated on both raw and postprocessed ones. Therefore, the number of change masks for an image pair is doubled.

4.2.1.2 Experiments using K-Means Clustering

Average F-measure values obtained for 15 different feature combinations are provided in Table 4.3 (a). Scores shown in Table 4.3 (a) are the average F-measure values of 15 image pairs in the AICD subset with the specified feature set and validity index configuration. For each feature combination, 4 different F-measure values are provided with respect to the selection of the validity index. Evaluations are made on the raw result mask. Cluster numbers for each image pair are automatically found using the selected validity index. Therefore, it may be different for image pairs and not provided in the tables.

In Table 4.3 (a), the validity index that gives the best results for each feature set are highlighted in green. Also, feature set and validity index combinations that give acceptable results (higher than %80) are highlighted with darker green. The results can be evaluated with respect to the types of validity indexes and feature vectors. When validity indexes are inspected, it can be easily seen that utilization of DB index generally produces relatively better results. In 11 results out of 15, DB index gives the highest F-measure values. CH index also seems like a successful criterion with 3 highest F-measure values and relatively similar scores to DB. With the inspection of feature sets, it can be seen that 8 out of 15 feature combinations provide F-measure values higher than 0.80. Best result is obtained as 0.87 when only the range filter difference values (F) are used in the classification step. Interestingly, addition of other features

Table 4.3: Average F-Measure Values for K-Means Results

(a) Raw F-Measure Values

	DB	CH	KL	Wint
D	0.83	0.82	0.75	0.76
M	0.62	0.63	0.59	0.59
R	0.41	0.48	0.35	0.30
F	0.87	0.79	0.67	0.56
DM	0.83	0.82	0.79	0.77
DR	0.86	0.82	0.79	0.77
DF	0.83	0.80	0.79	0.77
MR	0.50	0.50	0.47	0.18
MF	0.63	0.51	0.57	0.57
RF	0.25	0.29	0.28	0.25
DMR	0.85	0.79	0.77	0.77
DMF	0.83	0.82	0.77	0.77
DRF	0.85	0.79	0.76	0.80
MRF	0.39	0.36	0.32	0.25
DMRF	0.85	0.79	0.77	0.75

(b) Post-Processed F-Measure Values

	DB	CH	KL	Wint
D	0.87	0.85	0.75	0.71
M	0.64	0.64	0.59	0.59
R	0.40	0.41	0.34	0.33
F	0.79	0.76	0.62	0.51
DM	0.86	0.85	0.78	0.72
DR	0.87	0.85	0.78	0.74
DF	0.87	0.85	0.79	0.74
MR	0.56	0.51	0.55	0.22
MF	0.61	0.53	0.59	0.55
RF	0.30	0.36	0.33	0.28
DMR	0.86	0.85	0.78	0.74
DMF	0.86	0.85	0.78	0.71
DRF	0.86	0.85	0.78	0.79
MRF	0.48	0.47	0.38	0.28
DMRF	0.86	0.85	0.82	0.71

to range filter is not supportive and decreases the overall value. Converse implications are, also, present in the Table 4.3 (a). For example, usage of intensity difference values (D) results in a F-measure of 0.83. Ratio difference (R) values gave poor performance of 0.41. However, when D and R combined together, a higher F-measure, 0.86, is obtained. Therefore, a feature which is insufficient for classification alone, can improve the performance of another feature when used in combination. Finally, the competitive feature sets can be named as F, DR, DMR, DMF, DRF and DMRF.

In Table 4.3 (b), average F-measure values after post-processing are shown. Table 4.2 (b) have the same display style with Table 4.2 (a). Similar to Table 4.2 (a), DB and CH indexes provide relatively higher scores than other validity indexes. 8 of 15 feature sets provide scores higher than 0.85. The highest performance value, 0.87, is obtained with feature vectors D, DR and DF. In order to analyze the effects of post-processing better, Average F-measure scores of raw and postprocessed change masks obtained with validity index DB are tabulated in Table 4.4.

Positive gains are highlighted in green whereas negative ones are shown in red in Table 4.4. In 12 cases, post-processing increases the average values. When only competitive feature combinations (D , F, DM, DR, DF, DMR, DMF, DRF, DMRF) are considered, post-processing increases the values in all but one. Average increase is found as 1.95%. This can be considered as a negligible increase, but when the visual results are inspected, significant improvements can be observed. The performance metric treats a change object as found when its 20 percent is present in the resulting mask. Visual inspection shows that raw images are not visually satisfactory even if their Recall value is 1. Post-processing step also resolves such problems. In summary, postprocessing improves the visual results even if quantitative results stay the

Table 4.4: Post Processing Gains on K-Means Results

	Raw	PP	Gain (%)
D	0.83	0.87	4.15%
M	0.62	0.64	1.97%
R	0.41	0.40	-1.36%
F	0.87	0.79	-8.20%
DM	0.83	0.86	3.32%
DR	0.86	0.87	1.62%
DF	0.83	0.87	4.15%
MR	0.50	0.56	6.19%
MF	0.63	0.61	-2.10%
RF	0.25	0.30	4.53%
DMR	0.85	0.86	1.37%
DMF	0.83	0.86	3.32%
DRF	0.85	0.86	0.81%
MRF	0.39	0.48	8.73%
DMRF	0.85	0.86	0.81%
Average			1.95%

same. In order to analyze the results, raw and post-processed Precision-Recall values obtained with feature set DF and validity index DB are presented in Table 4.5.

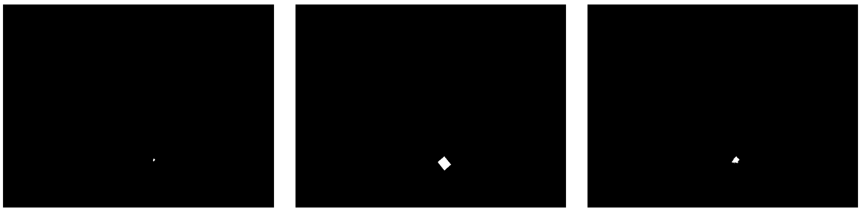
Table 4.5: Precision-Recall Values for Feature Vector DF

	Raw Results		Processed Results	
ID	Precision	Recall	Precision	Recall
1	100%	100%	100%	100%
2	-	0%	-	0%
3	100%	100%	100%	100%
4	100%	100%	100%	100%
5	100%	57%	100%	100%
6	79%	100%	54%	100%
7	100%	100%	100%	100%
8	100%	100%	100%	100%
9	100%	100%	100%	100%
10	100%	100%	100%	100%
11	43%	100%	72%	100%
12	100%	100%	100%	100%
13	100%	29%	100%	100%
14	0%	0%	0%	0%
15	100%	100%	100%	100%

Results in Table 4.5 are inspected from two different perspectives. First, the effects of post-processing on Precision - Recall values is discussed. Then, image pairs with low performance values are examined. It can be seen from Table 4.5 that, post-processing always increases the Recall values. Image pairs 5 and 13 are examples of Recall gains from post-processing.

Change masks for these image pairs are given in Figure 4.1.

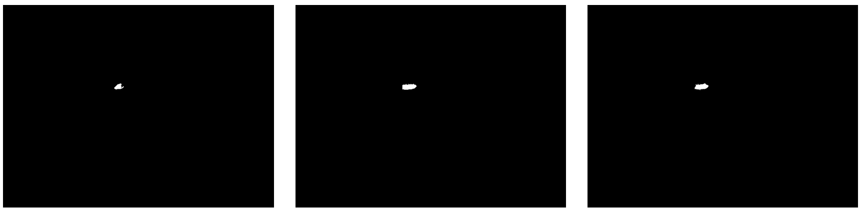
(a) Raw Result 5 (b) Ground Truth 5 (c) Post-processed Result 5



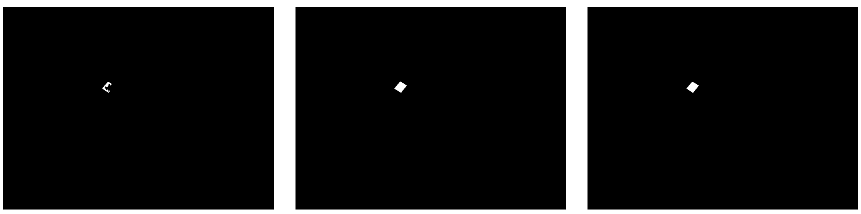
(d) Raw Result 13 (e) Ground Truth 13 (f) Post-processed Result 13

Figure 4.1: Qualitative Results of Post-processing on Image Pairs 5-13

For both image pairs 5 and 13, raw results include a small portion of the changed object. For image pair 5, post processing fetches the whole change object producing a visual result nearly identical to the ground truth mask. However, post processing can not return the whole object in 13th image pair. The change object is divided into two segments in this run and the raw result does not intersect with the second segment of the object. As a result, whole object can not be retrieved. Visual result is still more satisfying than that of the raw one. Since a bigger portion of the object is found, recall value is also increased. Like mentioned before, post-processing also enhances visual results in many cases, even if outcomes can not be perceived from quantitative values. Examples of such a case for image pairs 1 and 7 are shown at Figure 4.2



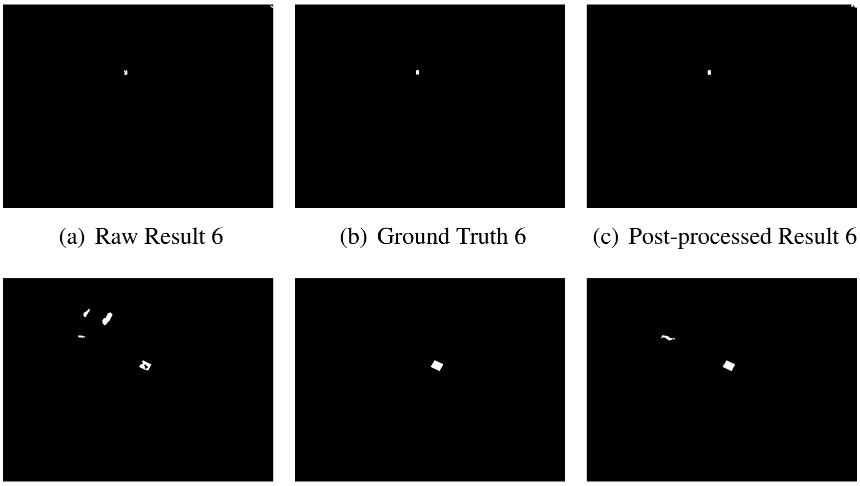
(a) Raw Result 1 (b) Ground Truth 1 (c) Post-processed Result 1



(d) Raw Result 7 (e) Ground Truth 7 (f) Post-processed Result 7

Figure 4.2: Qualitative Results of Post-processing on Image Pairs 1-7 with K-Means

Change masks of image pairs 1 and 7 have recall values 1 both before and after post-processing. However, visual enhancements can be clearly seen from Figure 4.2. Both post-processed results are nearly identical to the actual change objects. When results of all image pairs are inspected, it is seen that post-processing visually enhances all of them and creates more satisfactory results. Post-processing does not guarantee higher precision values in all cases like it did with recall values. According to Table 4.4, post-processing increases precision values in image pair 11 and reduces in 6. These results are visualized in Figure 4.3.



(a) Raw Result 6 (b) Ground Truth 6 (c) Post-processed Result 6

(d) Raw Result 11 (e) Ground Truth 11 (f) Post-processed Result 11

Figure 4.3: Qualitative Results of Post-processing on Image Pairs 6-11 with K-Means

In the change mask of image pair 6, the small segment in the upper right corner of the image can not be eliminated with post-processing. Also, post processing makes the segment bigger by fetching the intersecting segment. The change is relatively small with respect to the image size in this example. This property causes a relatively high decrease in the precision values. Change mask of the image pair 11 includes 3 segments other than the change object. Two of these segments are eliminated in the post-processing step using segment similarity. Post-processing made the third segment slightly larger. However, postprocessed mask has higher precision than before. Other two interesting image pairs are 2 and 14. For image pair 2, an empty result mask is created resulting in an uncomputable precision and zero recall value. This is because of the fact that two intermediate change masks obtained with different segmentations do not include intersecting parts when feature vector DF is used. Some feature combinations give better results in image pair 2. Usage of F, MR, RF and MRF feature vectors result in precision values of $0.71, 0.03, 0.03, 0.03$ and recall value of 1. However, utilization of these features yields relatively poor results when all 15 image pairs are considered. Resulting mask of image pair 14 includes only false positives. Feature set DF is not sufficient in defining the changed regions in this case. When the feature vectors F, M and MF are employed separately for image pair 14, changed region can be found with a F-Measure value of 1. However, when 15 image pairs are considered, selection of these features provide relatively low average F-Measure values.

4.2.1.3 Experiments using Heuristic Thresholding

Average F-measure values for raw and postprocessed change masks obtained using heuristic thresholding method are provided in Table 4.6. Heuristic thresholding method works by analyzing the histogram of features. Therefore, it cannot be used with combinations of features. Only one feature histogram is analyzed for each image pair. Scores shown in Table 4.6 are the average F-measure values of 15 image pairs in the AICD subset with the specified feature and histogram bin number selection. It should be noted that when working on ratio difference values, the values are normalized to be compatible with the heuristic thresholding method. Any value v smaller than 1 are equaled to $1/v$. This way, any possible changed region has a value higher than 1. Then, values are normalized into interval $0-1$ producing a distribution similar to other features.

Table 4.6: Average F-Measure Values for Heuristic Thresholding Results

(a) Raw F-Measure Values

	20	50	80
D	0.86	0.92	0.91
R	0.44	0.40	0.46
M	0.56	0.61	0.61
F	0.80	0.79	0.82

(b) Post-Processed F-Measure Values

	20	50	80
D	0.86	0.90	0.88
R	0.46	0.36	0.44
M	0.57	0.62	0.62
F	0.79	0.65	0.72

The layout of the Table 4.6 is similar to the Table 4.3. The rows represent the current choice of feature, whereas columns depict the number of bins. The highest performance value for a fixed feature is highlighted in green, representing the best bin number for this feature. Also, feature-bin number combinations that provide acceptable results are highlighted in a darker green color. When both tables are inspected, it can be clearly seen that the best results are obtained when the intensity difference values are selected as the feature. Feature D produced F-Measure values of 0.92 and 0.90 with raw and post-processed results, respectively. These values are higher than the best result, 0.87, obtained with K-Means clustering method. Range filter difference values, F, also seem as a good candidate with scores of 0.82 and 0.79. Other two features, R and M, give inapplicable results and are not considered for further usage. For nearly each feature, best resulting number of bins is different. Therefore, the selection of bin numbers should be feature specific. For feature D, 50 and 80 are acceptable number of bins. In order to discuss effects of postprocessing in detail, the post-processing gains for each feature and bin number are shown at Table 4.7.

When the bin number 20 is selected, post-processing increases the F-Measure values . Only a slight decrease is observed for the feature F. The overall gain is found as 0.65%. However, as it is mentioned in the previous K-Means experiments, visual results are generally enhanced. Post-processing generally decreases the overall performance results in other bin number selections. In order to understand the possible causes for the decrease, Precision - Recall values for each combination are inspected. As an example, raw and post-processed Precision - Recall

Table 4.7: Post-processing Gains on Heuristic Thresholding Results

(a) Bin Number 20

	Raw	PP	Gain (%)
D	0.86	0.86	0.32%
R	0.44	0.46	1.78%
M	0.56	0.57	1.30%
F	0.80	0.79	-0.80%
Avg			0.65%

(b) Bin Number 50

	Raw	PP	Gain (%)
D	0.92	0.90	-2.03%
R	0.40	0.36	-4.56%
M	0.61	0.62	1.52%
F	0.79	0.65	-13.38%
Avg			-4.61%

(c) Bin Number 80

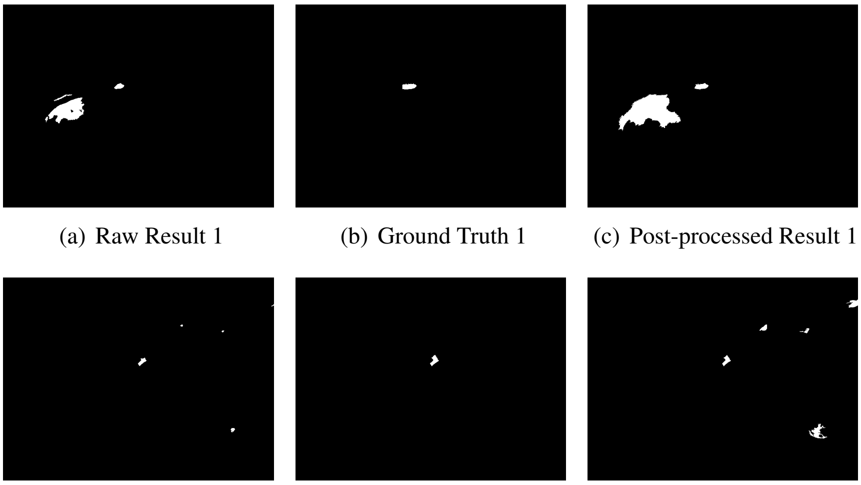
	Raw	PP	Gain (%)
D	0.91	0.88	-3.07%
R	0.46	0.44	-2.58%
M	0.61	0.62	1.39%
F	0.82	0.72	-9.52%
Avg			-3.44%

values for the Feature D obtained with bin number 50 are given at Table 4.8.

Table 4.8: Precision-Recall Values for Feature D

	Raw Results		Processed Results	
ID	Precision	Recall	Precision	Recall
1	11.01%	100.00%	5.37%	100.00%
2	82.98%	100.00%	80.75%	100.00%
3	100.00%	100.00%	100.00%	100.00%
4	100.00%	100.00%	100.00%	100.00%
5	100.00%	100.00%	100.00%	100.00%
6	78.21%	100.00%	54.22%	100.00%
7	100.00%	100.00%	100.00%	100.00%
8	100.00%	100.00%	100.00%	100.00%
9	100.00%	100.00%	100.00%	100.00%
10	100.00%	100.00%	100.00%	100.00%
11	43.12%	100.00%	72.11%	100.00%
12	100.00%	100.00%	100.00%	100.00%
13	100.00%	100.00%	100.00%	100.00%
14	67.50%	100.00%	18.82%	100.00%
15	100.00%	100.00%	100.00%	100.00%

When the results in Table 4.8 are compared with the visual results, some interesting behaviors of the proposed method are detected. First of all, proposed classification method generally provides high recall values, especially when used with intensity or range filter difference features. Also, the method relies heavily on the distribution of feature samples. One sample can completely change the outcome of the algorithm. Therefore, the proposed method is not robust as the others. It does not always guarantee a good classification result even if the feature distribution is easily distinguishable. Even before post-processing, the average recall value in Table 4.8 is 100%. Therefore, segment reconstruction step of post-processing does not effect the numerical recall values. Visual enchantments on the change masks are clearly visible. Hence, a slight decrease on average performance values is acceptable since the procedure enhances visual results. There are 5 image pairs with relatively low precision scores. In 4 of them, post-processing made the results even worse. In these examples, relatively bigger segments (generally combination of smaller ones) that belong the unchanged regions are present in the raw result mask. Post-processing method is not able to eliminate such segments. Since it also reconstructs these segments to create better visual results, their areas are increased. This results in a lower precision score. Visual results of post-processing on image pairs 1 and 14 are shown at Figure 4.4.



(a) Raw Result 1 (b) Ground Truth 1 (c) Post-processed Result 1

(d) Raw Result 14 (e) Ground Truth 14 (f) Post-processed Result 14

Figure 4.4: Qualitative Results of Post-processing on Image Pairs 1-14 with Heuristic Thresholding

As mentioned before, large segments are not eliminated in these examples. However, change objects resemble ground truth masks more after post-processing. These results conclude that, even in cases that it reduces precision values, post-processing is necessary. In summary, proposed heuristic thresholding method generates results with high recall scores when used with features D or F. However, heuristic thresholding highly relies on the distribution of samples and can produce results with low precision values. Bin number selection should be specific to the utilized feature. Therefore, if a method with high recall guarantee is needed, heuristic thresholding can be utilized.

4.2.1.4 Experiments using Expectation - Maximization

Average F-measure values for raw and postprocessed change masks obtained using Expectation-Maximization method are provided in Table 4.9. The EM algorithm allows us to use multidimensional feature vectors. Hence, similar to K-Means, 15 different feature combinations are used. Estimated number of component density is the only parameter for the experiments. In this study, number of components in the density function is taken as 3, 4 and 5. Scores shown in Table 4.9 are the average F-measure values of 15 image pairs in the AICD subset with the specified feature combination and estimated component number.

Table 4.9: Average F-Measure Values for EM Method

(a) Raw F-Measure Values

	3	4	5
D	0.78	0.67	0.65
M	0.58	0.58	0.61
R	0.46	0.41	0.33
F	0.75	0.71	0.68
DM	0.73	0.74	0.76
DR	0.84	0.75	0.74
DF	0.81	0.70	0.72
MR	0.66	0.72	0.65
MF	0.73	0.70	0.72
RF	0.82	0.77	0.74
DMR	0.81	0.78	0.74
DMF	0.77	0.76	0.78
DRF	0.88	0.74	0.50
MRF	0.68	0.68	0.60
DMRF	0.81	0.83	0.74

(b) Post-processed F-Measure Values

	3	4	5
D	0.88	0.84	0.84
M	0.57	0.57	0.57
R	0.48	0.48	0.48
F	0.81	0.78	0.74
DM	0.76	0.78	0.79
DR	0.83	0.81	0.79
DF	0.90	0.90	0.89
MR	0.61	0.69	0.70
MF	0.73	0.70	0.71
RF	0.83	0.83	0.79
DMR	0.87	0.84	0.82
DMF	0.61	0.69	0.70
DRF	0.99	0.83	0.73
MRF	0.71	0.72	0.62
DMRF	0.87	0.92	0.81

The layout and the highlighting scheme of the Table 4.9 is the same as that of Table 4.3. According to Table 4.9 (a), best performance scores are obtained when the number of components is taken as 3. The only meaningful result obtained with component number 4 is the score 0.83% which is achieved when DMRF is used as the feature vector. Since this feature combination is expected to captivate all information in different features, component number 4 is considered in further experiments. Component number 5 produces relatively low performance scores than other selections. Best performance score for raw values, 0.88%, is acquired with the feature set DRF and component number 3 configuration. DR, DF, RF, DMR and DMRF are other promising feature combinations which provide relatively high scores in raw masks. When the postprocessed results in Table 4.8 (b) are inspected, feature set DRF stands out as the highest score ever obtained with any configuration. This score, 0.99, is obtained when number of components is selected as 3. Feature set DMRF, which is depicted as a promising combination on raw results, produces the second highest score of 0.92. Similar to raw mask results, this score is obtained when the component number is selected as 4. In conclusion, DRF and DMRF appear as the most proper feature selections for Expectation-Maximization method. DF can be, also, included in this subset of appropriate feature selections with a score of 0.90. In order to analyze the effects of post-processing, raw and postprocessed performance scores of different feature sets are compared in Table 4.10.

According to the Table 4.10, post-processing generally increases the performance scores. The

Table 4.10: Post-Processing Gains on EM Results

	Raw	PP	Gain (%)
D	0.78	0.88	10.53%
M	0.58	0.57	-0.93%
R	0.46	0.48	2.47%
F	0.75	0.81	5.50%
DM	0.73	0.76	3.01%
DR	0.84	0.83	-1.13%
DF	0.81	0.90	8.16%
MR	0.66	0.61	-5.01%
MF	0.73	0.73	-0.80%
RF	0.82	0.83	0.70%
DMR	0.81	0.87	5.93%
DMF	0.77	0.61	-15.49%
DRF	0.88	0.99	11.25%
MRF	0.68	0.71	6.48%
DMRF	0.81	0.87	0.06
			2.25%

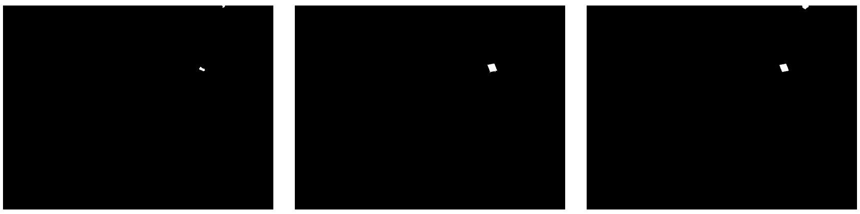
overall increase is found as 2.25%. If only promising feature combinations (DF, DRF, DMRF) are considered, overall gain is computed as 8.63%. Visual inspection of change masks shows that, EM generally produces raw change masks that include only small portions of change objects, without any false positives. This is especially true when the number of components is selected as 3. Hence, precision values for change masks are generally high. Segment reconstruction increases the recall values, causing an increase in the overall F-Measure values. Since the precision values are already high, this method produces best results among other classification methods tested in this study. Precision - Recall scores for the best scoring combination, DRF and 3 components, is shown in Table 11.

Only in one image pair, 2, the precision value is lower than 100%. Proposed configuration provides results with relatively high precision values. The precision values of image pair 2 are %92 and %76 for the raw and postprocessed masks, respectively. These performance scores are, also, acceptable. When compared to precision scores, recall values are not very satisfactory when raw results are considered. However, none of the recall values equals to 0, meaning in all samples, a portion of the change object is found. Postprocessing reconstructs these portions, resulting in recall scores of %100 for all examples. Visual results for image pair 2 are given in Figure 4.5.

Post-processing fails to eliminate the false change area in the upper region of the image. Whole segment is fetched and this results in a minor decrease in precision score. Like the previous visual examples, post-processing enhances the representation of the change object. This analysis implies that, EM method with correct configurations produces relatively high scores compared to the K-Means and heuristic thresholding methods tested in this study. EM method generates results with relatively high precision values, in contrast to other mentioned

Table 4.11: Precision-Recall Values for Feature Vector DRF

	Raw Results		Processed Results	
ID	Precision	Recall	Precision	Recall
1	100.00%	32.24%	100.00%	100.00%
2	92.28%	100.00%	76.36%	100.00%
3	100.00%	100.00%	100.00%	100.00%
4	100.00%	100.00%	100.00%	100.00%
5	100.00%	57.06%	100.00%	100.00%
6	100.00%	67.08%	100.00%	100.00%
7	100.00%	36.23%	100.00%	100.00%
8	100.00%	100.00%	100.00%	100.00%
9	100.00%	83.98%	100.00%	100.00%
10	100.00%	57.52%	100.00%	100.00%
11	100.00%	70.01%	100.00%	100.00%
12	100.00%	100.00%	100.00%	100.00%
13	100.00%	78.29%	100.00%	100.00%
14	100.00%	100.00%	100.00%	100.00%
15	100.00%	100.00%	100.00%	100.00%



(a) Raw Result 2 (b) Ground Truth 2 (c) Post-processed Result 2

Figure 4.5: Qualitative EM Results of Post-processing on Image Pair 2

methods. Therefore, EM can be considered as a convenient method if high precision values are aimed.

4.2.2 Results of Experiments Performed on the AICD Dataset

After deciding on the promising feature sets and parameter combinations for each method in the former section, these configurations are used on the full AICD dataset [7]. AICD dataset includes 1000 image pairs for 100 different scenes. For each scene, there are 5 image pairs obtained from different viewing angles. Also, each scene has 2 distinct versions with soft and hard shadows. In the experiments, one viewpoint angle is chosen for each scene with soft shadows. There are some problems with the change objects in some image pairs. Since the dataset is created artificially in a 3D modeling program, one type of object (a tent) has artifacts on it. Ground truth and the actual change object do not match in these examples. Therefore, image pairs that include this object are discarded from the dataset. A visual example of the mentioned case can be seen at Figure 4.6.

(a) Example Incomplete Object (b) Ground Truth for the Object

Figure 4.6: Example Object with Artifacts

The example object in Figure 4.6 has some incomplete parts and seems like an artifact on the scene, which makes it harder to detect as changes. Even in full detection, numerical performance assessment cannot be done because of the ground truth differences. These scenes are not valid for measuring the performance of the proposed change detection method. 17 image pairs include such objects and discarded from the dataset. The final dataset includes 83 image pairs with soft shadows. Each image in the dataset has a resolution of 800x600 pixels. Size of the change objects vary with the scene ranging from 91 to 1271 pixels. Therefore, change objects cover between 0.019 and 0.265 percent of the images [7].

4.2.2.1 Experiment Setups

In the former section, 3 different methods are examined and discussed; K-Means clustering, heuristic thresholding and EM. Promising feature set and parameter configurations detected in the former section are tabulated in Table 4.12 for each method.

Table 4.12: Promising Configurations Found at AICD Subset Experiments

	Feature Sets	Parameters
K-Means	D, DR, DF, DM, DRF, DRM, DMF, DRMF	Validity Indexes DB and CH
Heuristic Thresholding	D and F	Histogram Bin Numbers 20, 50, 80
Expectation-Maximization	DF, DRF, DMRF	Density Component Numbers 3 and 4

For the K-Means method, 8 feature sets out of 15 are chosen as promising combinations. Validity index Davies-Bouldin produces the change masks with highest performance scores according to the experiments conducted on the AICD subset. Calinski-Harabsz index, also, provides overall performance scores relatively close to DB. Instead of obtaining two different results for each image pair with respect to DB and CH, suggested cluster numbers are combined in the experiments conducted on the AICD dataset. Each validity index finds a feasible cluster number for the provided feature set. Final cluster number, which is used in the experiments, is taken as the mean value of the cluster numbers suggested by DB and CH indexes. This way, a single result is obtained for a feature set selection, instead of two. This procedure may produce different number of clusters for each image pair. Therefore, cluster numbers for average performance scores are not presented.

According to experiments conducted on AICD subset, promising feature selections for heuristic thresholding are D and F. These experiments do not implicate a successful choice of histogram bin number. Therefore, heuristic thresholding experiments on AICD subset are conducted with features D and F and histogram bin numbers, 20, 50 and 80.

Lastly, in the experiments conducted with EM method on the AICD subset, features DF, DRF and DMRF are emerged as successful combinations. With component numbers 3 and 4, highest performance scores are obtained. Hence, tests conducted on AICD dataset use combinations of these features and component number selections.

4.2.2.2 Experiments using K-Means Clustering

Average F-Measure values obtained on the AICD dataset for each promising feature combination are given at Table 4.13.

Table 4.13: Average F-Measure Values for K-Means Method

	Raw	PP
D	0.64	0.63
DR	0.61	0.54
DF	0.64	0.56
DM	0.57	0.55
DRF	0.50	0.43
DRM	0.51	0.49
DMF	0.49	0.48
DRMF	0.54	0.50

The best resulting feature selections according to Table 4.13 are D and DF. They both provide the highest average F-Measure value in the raw results with a score of 0.64. The outcome is similar in the post-processed results case, with the highest two scores of 0.63 and 0.56, respectively. The results in Table 4.13 are relatively compatible with the results in the former subsection, that are provided in Table 4.4 where D and DF are among the best scoring features. Therefore, D and DF can be declared as the best performing features for K-Means method. The Precision - Recall values for features D and DF are shown in Table 4.12.

According to Table 4.14, feature DF provides a result with a higher recall value than D, in consequence of lower precision. When other feature combinations for K-Means are inspected,

Table 4.14: Average Precision-Recall Values For Best Feature Selection in K-Means Method

	Raw		PP	
	Precision	Recall	Precision	Recall
D	0.67	0.62	0.59	0.68
DF	0.54	0.76	0.43	0.81

it can be seen that DF has the highest recall value among others. On the other hand, feature D provides a more balanced result, with similar precision and recall values. Therefore, final selection should be done according to the purpose of the change detection application. When D is selected, 37 image pairs out of 83 have precision and recall scores of %100. Hence, approximately %45 of the image pairs are correctly classified. This value drops to %25 when DF is selected. 56 image pairs have perfect precision scores when D is utilized, whereas this number increases to 66 when DF is selected.

4.2.2.3 Experiments using Heuristic Thresholding

The average F-measure values obtained on full AICD dataset with features D and F are shown in Table 4.15. Performance scores for 3 different histogram bin number selections, 20, 50 and 80, are provided.

Table 4.15: Average F-Measure Values for Adaptive Thresholding Method

(a) Raw F-Measure Values

	Raw		
	20	50	80
D	0.67	0.71	0.72
F	0.44	0.52	0.48

(b) Post-Processed F-Measure Values

	Post-Processed		
	20	50	80
D	0.64	0.68	0.68
F	0.42	0.46	0.44

The results shown in Table 4.15 are consistent with the former experiments conducted on the subset. Best raw results are obtained as %71 and %72, when the feature D is used with bin numbers 50 and 80, respectively. These combinations yield best results as %92 and %91 (Table 4.5) on the AICD subset. Post-processed results are relatively worse with score of %68. When the visual contributions of post-processing are considered, these drops on the quantitative scores are negligible. Utilization of F provides scores lower than %50. This result is, also, consistent with the former experiments conducted with feature F, which produced lower scores than D. However, performance scores in this experiment for feature F are not acceptable as the former ones. Therefore, only applicable feature for the heuristic thresholding is found as D, with respect to results obtained on AICD dataset. Also, heuristic thresholding method produces better results in overall than the K-Means approach. Precision - Recall values for the feature D with different bin number selections are shown in Table 4.16.

Table 4.16: Average Precision-Recall Values for Adaptive Thresholding Method with D

	Raw		PP	
	Precision	Recall	Precision	Recall
20	68.19%	66.14%	56.37%	73.20%
50	68.06%	74.67%	58.48%	80.43%
80	68.09%	76.03%	57.66%	82.84%

According to Table 4.16, the optimal bin numbers are 50 and 80. For these two bin numbers, post-processing results in a faster drop on the precision values than the increase in the recall values. 34 and 37 image pairs are classified with F-measure score of %100 with bin numbers 50 and 80, respectively. These numbers are less than the ones obtained with K-Means method but heuristic thresholding method is able to classify changed regions in more image pairs. Therefore, one may conclude that heuristic thresholding seems like a better method for applications with both high precision and recall aims.

4.2.2.4 Experiments Expectation-Maximization

The average F-measure values obtained on AICD dataset with features sets DF, DRF and DMRF are shown in Table 4.17. Performance scores for 2 different number of component selections are provided.

Table 4.17: Average F-Measure Values for EM Method

	3		4	
	Raw	PP	Raw	PP
DF	0.67	0.67	0.63	0.70
DRF	0.60	0.67	0.55	0.63
DMRF	0.55	0.59	0.54	0.60

When the results in Table 4.17 are inspected, it can be seen that feature sets DF and DRF provide slightly better results independent of the component number. In contrast with other methods, post-processing increases the overall performance results. This is most likely caused by the fact that, EM method generally provides results with no false positives in the AICD dataset. Therefore, post-processing does not cause any precision drops. The choice of the component number does not effect the results with DF significantly. However, increasing the component number provides lower average results when DRF is utilized. The highest performance, %70, is obtained by the combination of feature set DF and component number 4. Contrary, in the former experiments conducted on the subset, best quantitative result is obtained with DRF. The Precision-Recall values for features D and DRF are given at Table 4.18.

As expected, EM method produces the highest precision values among all other methods. For

Table 4.18: Average Precision-Recall Values for EM Method with D and DRF

	Raw		PP	
	Precision	Recall	Precision	Recall
DF-3	0.72	0.62	0.64	0.72
DF-4	0.75	0.55	0.70	0.70
DRF-3	0.73	0.51	0.70	0.65
DRF-4	0.72	0.45	0.70	0.58

both feature selections, precision score is equal to or greater than %70. When Table 4.18 is inspected, it can be seen that feature DRF has no superiority over DF and can be discarded. When DF is used with the component number 3, 41 image pairs are classified with precision and recall values of %100. This configuration correctly classifies nearly %50 of image pairs in the AICD dataset. In conclusion, EM method provides change masks with highest precision scores in general.

4.2.2.5 Comparison of Methods and Feature Sets

The highest performance scores for postprocessed change masks obtained with each method are shown in Table 4.19.

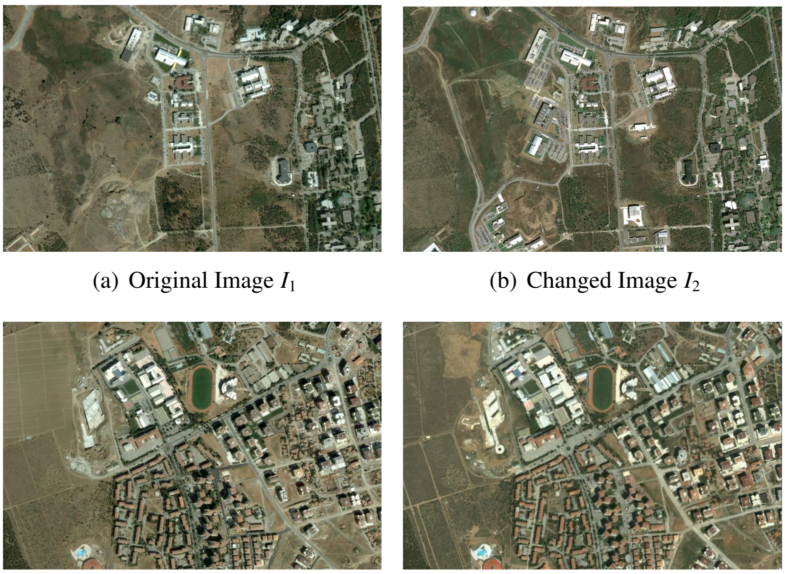
Table 4.19: Highest Performance Scores of Classification Methods

	Highest Average F-Measure Score	Utilized Feature Set
K-Means	0.63	D
Heuristic Thresholding	0.68	D
Expectation-Maximization	0.70	DF

According to Table 4.19, heuristic thresholding and Expectation-Maximization methods perform slightly better than K-Means Clustering. All highest performance scores are obtained with feature sets that include intensity difference values feature, D. Hence, feature D can be selected as the most discriminative feature among others. When performance scores for other configurations are inspected, range filter difference values, F, emerges as the most supportive feature. Including F in the feature set generally increases performance values for this dataset. For this dataset, K-Means Clustering method does not seem as a plausible selection. Heuristic thresholding provides results with high recall scores. Contrary, change masks obtained with EM method have high precision scores. Therefore, if one aims to achieve a high recall score in consequence of possible low precision, heuristic thresholding seems to be the choice. However, if the aim is to obtain a change map with high precision, EM method should be chosen.

4.2.3 Experimental Analysis of the Proposed Change Detection Framework on Real World Data

In this subsection, the suggested change detection framework is tested on the real world data. Finding and/or designing a dataset with co-registered remote sensing image pairs with change characteristics is not a trivial task. Also, definition of change is pretty flexible and depends on the application domain. Therefore, in this subsection, quantitative evaluation is not made due to lack of ground truth information. 5 different image pairs are used during the experiments. First two of them are obtained from Google Earth application and covers the METU Technopolis and surrounding areas of METU, respectively. Other 3 image pairs are obtained from the SZTAKI AirChange Benchmark dataset [4]. Images from SZTAKI dataset have ground truth information, but they are roughly labeled. They also include non man-made areas which are not the focus of this study. Before applying change detection, each image pair is registered using the registration method mentioned in the background section. Similar to former two subsections, 3 different methods with different parameters and feature configurations are applied on these 5 image pairs. Only post-processed change masks are shown, since they are more meaningful to our visual perception. Experiment setup is entirely same with the former subsection. The feature set and parameter combinations for each method that are tabulated in Table 4.12 are utilized for each image pair. Resulting change masks are shown for visual inspection. Image pairs in the first subset is shown in Figure 4.7.

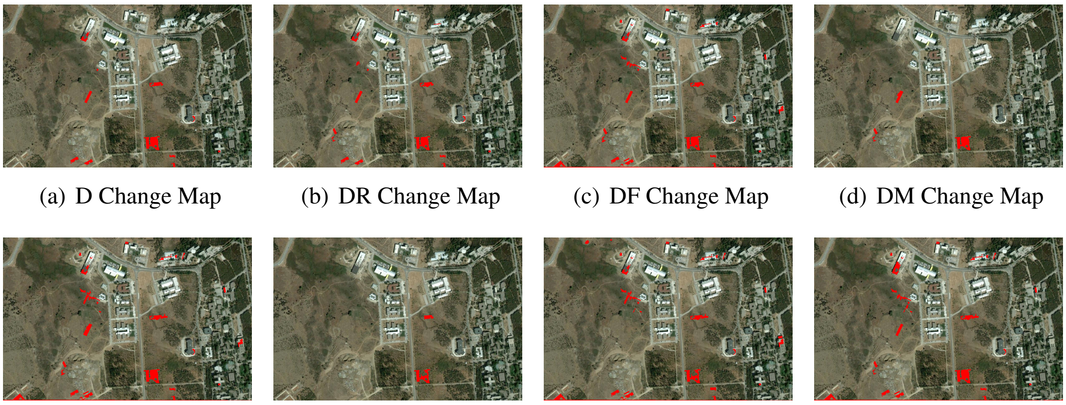


(a) Original Image I_1 (b) Changed Image I_2

(c) Original Image I_1 (d) Changed Image I_2

Figure 4.7: Image Pairs Obtained from Google Earth

As it can be seen from the first image pair, many new buildings are constructed in the scene. There is a slight viewpoint angle change between two images which can effect change detection procedure. Shadows of buildings appear differently, which is also a problem. Firstly, K-Means procedure is applied on the image pair. Resulting change maps are visualized on the original image, I_1, with red color. When segmenting the first two image pairs, minimum region area, spatial bandwidth and range bandwidth parameters are taken as 50, 10 and 6 for the changed image, I_2. For the original image, I_1, these parameters are taken as 10, 3 and 2. These parameters are found by visual inspection. Images are segmented with different parameter sets, similar to Subsection 3.3.1. Then, each segmentation result is inspected visually to decide on an appropriate parameter set. Change masks obtained with different feature sets are shown in Figure 4.8.



(a) D Change Map (b) DR Change Map (c) DF Change Map (d) DM Change Map

(e) DRF Change Map (f) DRM Change Map (g) DMF Change Map (h) DRMF Change Map

Figure 4.8: Google Earth Image Pair 1 K-Means Change Maps

The change maps look relatively similar and it is hard to detect a feature combination as the best performing one with visual inspection. In nearly all examples, changed regions correctly reflects the geometric properties of newly built regions. No feature selection is able to detect newly constructed roads, but buildings are generally detected. Feature selections DM and DRM produce change masks with few regions labeled as changes when compared to others. Many actual change objects are not labeled with these selections. However, with these feature selections, all observed changes in the change map are actual changes. Change map produced by feature D includes more correctly labeled change objects when compared to features DM and DRM. However, with this feature selection, two shadow regions are also mislabeled as change. Change maps produced by DR, DF, DRF and DMRF look fairly similar. These change maps include more regions labeled as change when compared to previously mentioned features DM, DRM and D. As a consequence, more change objects are correctly classified compared to former feature selections, but the number of mislabeled regions are also increased. Feature set DMF produces a change map with an interesting property. No other feature set is able to correctly classify the dome on the upper-left corner. With these change maps, it is hard to decide on an optimal feature combination. The selection should be done according to the application domain. It should be noted that, many mislabeled regions are caused by shadow and illumination differences. Change maps for the second image pair in the subset are given in Figure 4.9.

Change maps in Figure 4.9 are consistent with the change masks of first image pair. With this image pair, selection of feature D produces the change mask with fewest regions labeled as change. Other feature selections produce change maps with more correctly labeled change

(a) D Change Map

(b) DR Change Map

(c) DF Change Map

(d) DM Change Map

(e) DRF Change Map

(f) DRM Change Map

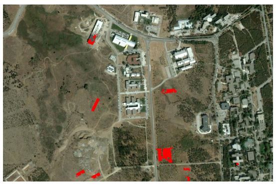
(g) DMF Change Map

(h) DRMF Change Map

Figure 4.9: Google Earth Image Pair 2 K-Means Change Maps

objects compared to feature D. However, shadows still cause problems and more regions are mislabeled as change when other feature combinations than D are selected. Different from the previous image pair, some change objects are not detected entirely, only a portion of objects are labeled. Therefore, visual results for this image pair are less satisfying than the ones for the first image pair.

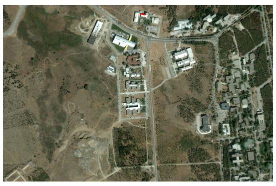
Change maps for the first image pair, when heuristic thresholding is utilized, are shown in Figure 4.10.



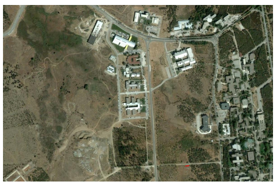
(a) D Bin Size: 20

(b) D Bin SİZe: 50

(c) D Bin Size: 80



(d) F Bin Size: 20



(e) F Bin Size: 50

(f) F Bin Size: 80

Figure 4.10: Google Earth Image Pair 1 Heuristic Thresholding Change Maps

According to Figure 4.10, bin size selection 20 for feature D produces the results with fewest regions labeled as change. There are some false change regions caused by shadows in this result, however they are less in number when compared to other bin number selections. Bin numbers 50 and 80 produces same results in this example. With these bin number selections, number of correctly labeled change objects are increased when compared to bin number selection 20, in exchange for a slight increase in mislabeled regions. Therefore, it can be said that 50 or 80 is the correct bin number in this case. Feature F fails to produce any meaningful results with any bin number selection. Feature F does not have discrimination capabilities when used alone on real world satellite images. For this reason, we suffice to display the results for the second image pair for feature D and in Figure 4.11.

(a) D Bin Size: 20 (b) D Bin Size: 50 (c) D Bin Size: 80

Figure 4.11: Google Earth Image Pair 2 Heuristic Thresholding Change Masks

Change maps are consistent with the ones obtained with the first image pair. Employing the bin number 20 produces a change mask of similar characteristics with the change mask of the first image pair. Change masks obtained with bin number 50 and 80 are fairly similar for this image pair. However, change mask produced by bin number selection 50 includes more correctly classified change objects. With this information, it can be concluded that bin number 50 is the best selection for this subset.

Finally, EM algorithm is utilized on the Google Earth images subset. When conducting experiments on real data with EM, experiment setup slightly changes. The experiments conducted on the AICD dataset show that most satisfactory feature selections are DF, DRF and DMRF. Promising component number selections are found as 3 and 4. However, these assumptions are not valid for experiments conducted on the real world images. Feature combinations DRF and DMRF fail to produce satisfactory change maps. Also, component numbers 3 and 4 generally produce change maps which do not include most of the changed objects. Therefore, component number 2 is also used for experiments on real world data. Change maps obtained with feature set DF for both image pairs are shown in Figure 4.12.

For the first image pair, most visually satisfying change map is obtained with the component number 2. Nearly all observed change regions in the change mask are actual change regions. Increasing the component number results in losing correctly labeled change regions. However, results are not satisfactory for the second image pair. For component number selections 3 and 4, EM method produces an empty change mask. Change mask obtained with component number 2 includes some correct classified regions but they are only a slight portion of the actual changes. In the former experiments on the AICD dataset, EM proved to be the method with highest precision guarantee. Supporting this claim, nearly all observed regions in change masks shown in Figure 4.12 are actual changes. If a higher precision value is aimed, EM method is still applicable.

The second subset, 3 images acquired from the SZTAKI Airchange Benchmark, have slightly lower spatial resolution according to visual inspection. No prior information about resolution of these images are present. Hence, this assumption relies on visual inspection. Therefore,

(a) First Image Pair Component Number: 2 (b) First Image Pair Component Number: 3 (c) First Image Pair Component Number: 4

(d) Second Image Pair Component Number: 2 (e) Second Image Pair Component Number: 3 (f) Second Image Pair Component Number: 4

Figure 4.12: Google Earth Image Pairs EM Change Masks

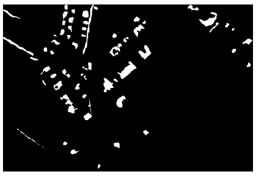
different parameter sets are used in the segmentation step. Minimum region area, spatial bandwidth and range bandwidth parameters are taken as 10, 6 and 4 respectively, for the changed image, I_2. These parameters are taken as 5, 2 and 1 for the original image, I_1. These parameter sets are also found empirically with visual inspection. Three image pairs in the SZTAKI dataset are shown in Figure 4.13.

(a) Change Image I_2 (b) Original Image I_1 (c) Change Image I_2

(d) Original Image I_1 (e) Change Image I_2 (f) Original Image I_1

Figure 4.13: Image Pairs from SZTAKI Airchange Benchmark Set

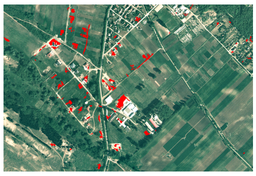
Same configurations of feature sets and parameters are used for the previous Google Earth Image dataset are utilized during the experiments. For each image pair, a roughly drawn ground truth mask exists. This ground truth masks include some areas that are not in the scope of this study. Also, they only give information about the area that change occurred, do not remark its boundaries, in contrary to current study. However, they are provided in this section for easier visual assessment. Change maps for the first image pair, obtained with K-Means method are shown in Figure 4.14.



(a) Ground Truth Mask

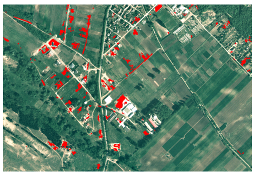
(b) D Change Map

(c) DR Change Map

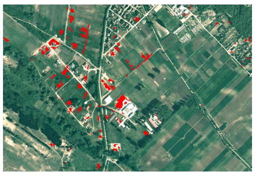


(d) DF Change Map

(e) DM Change Map



(f) DRF Change Map



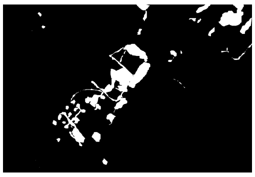
(g) DRM Change Map

(h) DMF Change Map

(i) DRMF Change Map

Figure 4.14: SZTAKI Image Pair 1 K-Means Change Masks

Similar to experiments conducted on Google Earth images, change map obtained with feature D labels fewer regions as change when compared to other feature selections. Other feature selections, such as DRF, DMF and DRMF, are successful at detecting most of the changes in the scene with few false positives, with respect to provided ground truth. For other two image pairs in the SZTAKI Airchange dataset, results of promising feature selections D, DMF and DRMF are shown in Figure 4.15.



(a) Image Pair 2 Ground Truth Mask

(b) Image Pair 2 D Change Map

(c) Image Pair 2 DMF Change Map

(d) Image Pair 2 DMRF Change Map

(e) Image Pair 3 Ground Truth Mask

(f) Image Pair 3 D Change Map

(g) Image Pair 3 DMF Change Map

(h) Image Pair 3 DMRF Change Map

Figure 4.15: SZTAKI Image Pairs 2, 3 K-Means Change Masks

Resulting change masks in Figure 4.15 are consistent with other change masks obtained on real world images using K-Means which are shown in Figure 4.8, 4.9 and 4.14. In general, many feature combinations give satisfactory performance on the SZTAKI Airchange subset, according to the rough ground truths.

Former experiments on the Google Earth images show that second method, heuristic thresholding, performs competitively only when the feature D is used. Change maps obtained with heuristic thresholding on the SZTAKI Airchange images with utilization of feature D are shown in Figure 4.16.

(a) Image Pair 1 Ground Truth Mask (b) Image Pair 1 Bin Number 20 (c) Image Pair 1 Bin Number 50 (d) Image Pair 1 Bin Number 80

(e) Image Pair 2 Ground Truth Mask (f) Image Pair 2 Bin Number 20 (g) Image Pair 2 Bin Number 50 (h) Image Pair 2 Bin Number 80

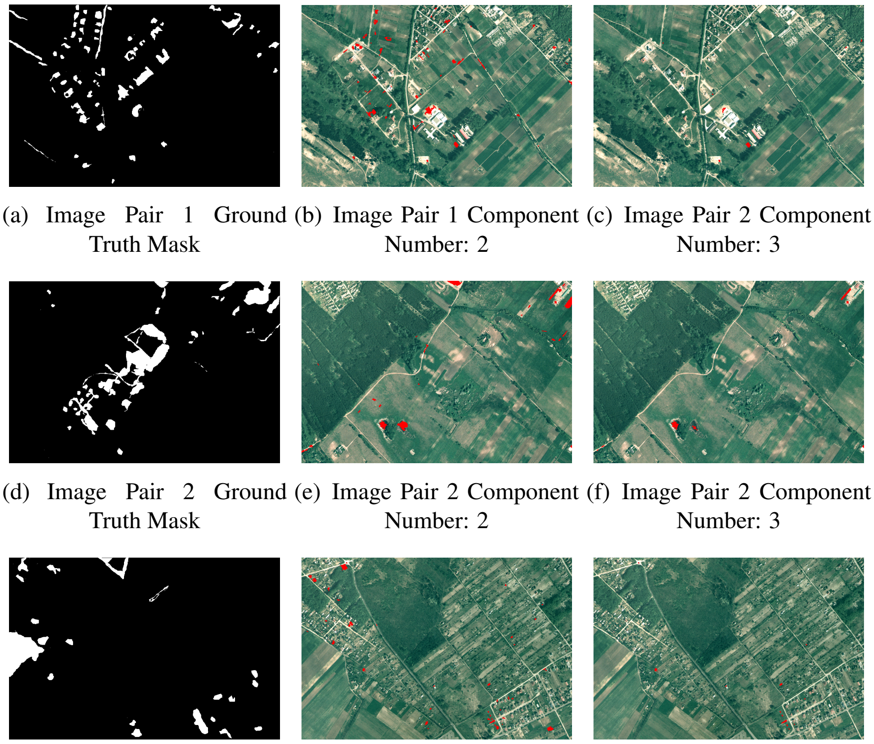
(i) Image Pair 3 Ground Truth Mask (j) Image Pair 3 Bin Number 20 (k) Image Pair 3 Bin Number 50 (l) Image Pair 3 Bin Number 80

Figure 4.16: SZTAKI Image Pairs Heuristic Thresholding Change Masks

According to Figure 4.16, heuristic thresholding fails to detect most of the changed objects if the bin number is selected as 20. Similarly, selection of bin number as 50 produces change masks which do not include most of the changed regions for image pairs 1 and 3. Bin number selection 80 results in competitive results with the K-Means method on same image pairs. Therefore, bin number 80 is definitely the best selection for SZTAKI Airchange dataset.

Experiments on the Google Earth image pairs with EM method showed that the most competitive feature combination is DF. Change masks obtained using EM method with 2 and 3 component number assumptions are shown in the Figure 4.17.

It can be seen from Figure 4.17 that selecting component number as 3 fails to produce any meaningful results with SZTAKI Airchange dataset. Change maps obtained with component number selection 2 are not significantly different than the ones produced by component number 3. Many of the change objects are not labeled in the change maps created by different component number selections. In conclusion, EM method performs worse than other



(a) Image Pair 1 Ground Truth Mask (b) Image Pair 1 Component Number: 2 (c) Image Pair 2 Component Number: 3

(d) Image Pair 2 Ground Truth Mask (e) Image Pair 2 Component Number: 2 (f) Image Pair 2 Component Number: 3

(g) Image Pair 3 Ground Truth Mask (h) Image Pair 3 Component Number: 2 (i) Image Pair 3 Component Number: 3

Figure 4.17: SZTAKI Image Pairs EM Change Masks

proposed algorithms in SZTAKI Airchange dataset. This low performance is most probably caused by the relatively low size of change objects. EM method treats such objects as negligible in change classification.

When the change maps obtained in this subsection on real world images with different methods are inspected, K-Means and heuristic thresholding methods emerge as the most competitive solutions. If high precision values are critical, EM method can be also utilized.

4.3 Comparison with Existing Algorithms

The proposed change detection method is compared with existing algorithms in the literature. 3 different change detection procedures, one being supervised and other two unsupervised, are chosen for this purpose [7, 11, 14]. First, the quantitative results of these algorithms on the AICD dataset are compared with the proposed algorithm. Some visual results are provided in the comparison. Then, resulting change masks for the real world dataset including images from Google Earth and SZTAKI Air Benchmark are compared. First algorithm selected for comparison is the supervised procedure based on optical flow presented by Bourdis, Marraud & Sahbi [7]. Authors of this algorithm created the AICD dataset and made it public for benchmarking change detection algorithms. In order to avoid confusion, this optical flow based algorithm is denoted as "Algorithm 1" in this section. PCA and K-Means based unsupervised change detection method proposed by Çelik in 2009 is chosen as the second method for comparison [14]. In this section, denotation "Algorithm 2" is used for referring to this method developed by Çelik. EM based method developed by Bruzzone and Prieto in 2000 is selected as the last method for comparison [11]. This EM based algorithm is mentioned as "Algorithm 3" in this section. Explanations of these mentioned algorithms are given in Section 2. Quantitative results on the AICD dataset for Algorithm 1 is given in the published article [7]. For measuring the performance of Algorithm 2 and 3 on the AICD dataset, these methods are implemented in MATLAB environment. Performance scores on AICD dataset for Algorithm 1,2,3 and proposed method are given in Table 4.20.

Table 4.20: Highest Performance Scores of Classification Methods

	F-Measure	Precision	Recall
Algorithm 1	0.57	0.56	0.58
Algorithm 2	0.10	0.06	0.51
Algorithm 3	0.10	0.05	0.99
K-Means	0.63	0.59	0.68
Heuristic Thresholding	0.68	0.58	0.80
Expectation-Maximization	0.70	0.64	0.72

First three rows of Table 4.20 show the average F-Measure, Precision and Recall values for the algorithms selected from the literature, Algorithm 1, 2 and 3. Following three rows shows the highest F-Measure values obtained using the proposed method, using three different classification methods with optimum configurations. According to [7], highest performance scores for Algorithm 1 are obtained when constrained optical flow is used on image pairs with no angle differences. For Algorithm 1, performance scores obtained with mentioned configuration is shown in Table 4.20. When implementing Algorithm 2 [14], image intensity values in the difference image are used as the main feature. 2x2 non-overlapping blocks are used. Then, feature vector space is created by projecting these blocks onto eigenvector space using PCA. First four principal components are used. Using K-Means clustering, this feature vector is clustered into 3 components. The proposed cluster number was 2 in the article [14]. However, this cluster number selection created results with many false positives. With this implication, cluster number is changed to 3. Algorithm 3 is implemented according to [11]. In [11], two similar approaches are presented. Both approaches work on the intensity difference values obtained from the difference image. First approach is pixel-based and second one uses MRF to model neighborhood relations. Pixel-based method is implemented as Algorithm 3. As it proposed in the article [11], 2 components are assumed to exist in mixture distribution of the difference image values. Parameters of these components are found using EM method. Then, difference image is grouped into two clusters with minimum Bayes error criterion. Cluster with the highest valued centroid is chosen as the change map. Before utilizing Algorithm 2 and 3 in the experiments, image pairs are registered and normalized.

According to Table 4.20, proposed method provides higher F-Measure, precision and recall scores than Algorithm 1. Each three classification methods presented in the study produces higher performance scores than Algorithm 1 when used with correct configurations. Proposed method, also, outperforms Algorithm 2. Algorithm 2 creates change maps with many false positives on AICD dataset, resulting in relatively low average precision score of 0.06. Also, average recall score of Algorithm 2, 0.51 is not competitive when compared to proposed method. Results in Table 4.20 shows that Algorithm 3 is successful in labeling the change objects. Average recall score of Algorithm 3, 0.99, is not exceeded by any configuration of the proposed method. Algorithm 3 successfully labels change objects in 80 image pairs out of 83 in AICD dataset. However, its relatively low average precision score, 0.05, makes Algorithm 3 an unsuccessful change detection method for AICD dataset. In conclusion, proposed method outperforms all three algorithms on the AICD dataset. Change masks for 3 different scenes, obtained with Algorithm 2, 3 and proposed method are shown in Figure 4.19 for visual comparison. Change masks for the proposed method, Figure 4.19 (b, f, j), are obtained with heuristic thresholding. Feature D and histogram bin number 50 is used in the experiments.

(a) Ground Truth Scene 1 (b) Proposed Methods Change Mask Scene 1 (c) Algorithm 2 Change Mask Scene 1 (d) Algorithm 3 Change Mask Scene 1

(e) Ground Truth Scene 2 (f) Proposed Methods Change Mask Scene 2 (g) Algorithm 2 Change Mask Scene 2 (h) Algorithm 3 Change Mask Scene 2

(i) Ground Truth Scene 3 (j) Proposed Methods Change Mask Scene 3 (k) Algorithm 2 Change Mask Scene 3 (l) Algorithm 3 Change Mask Scene 3

Figure 4.18: Visual Comparison of Proposed Algorithm on AICD Dataset

For first two scenes in Figure 4.19, proposed method generates change masks nearly identical to the ground truth. In the change map of the third scene, a small mislabeled area exists. Algorithm 2 fails to label the changed objects in first two scenes. Change mask for the third scene produced by Algorithm 2 includes the changed object. However, similar to previous two scenes, many other regions are falsely labeled as changes. As expected from the results of quantitative evaluation, Algorithm 3 correctly labels the changed regions in all three scenes.

However, all change maps produced by Algorithm 3 include many false positives. The results are similar for all image pairs in the AICD dataset. Visual comparison strengthens the implications derived from the quantitative results.

Proposed algorithm is tested and optimized on the the AICD dataset. For preventing any positive bias towards the proposed algorithm, results are also compared on the real world images obtained from Google Earth and SZTAKI Airchange Benchmark dataset. No quantitative comparison is made due to lack of ground truth information. Resulting change masks are provided for visual comparison. Each change map is overlaid onto the original image with red color. Rough ground truths for the SZTAKI AirChange are also provided. Results for the two Google Earth image pairs are shown in Figure 4.20. These two image pairs are already shown in Figure 4.7. Change masks for the proposed method, Figure 4.20 (a,d) are obtained with heuristic thresholding. Feature D and histogram bin number 50 is used in the experiments.

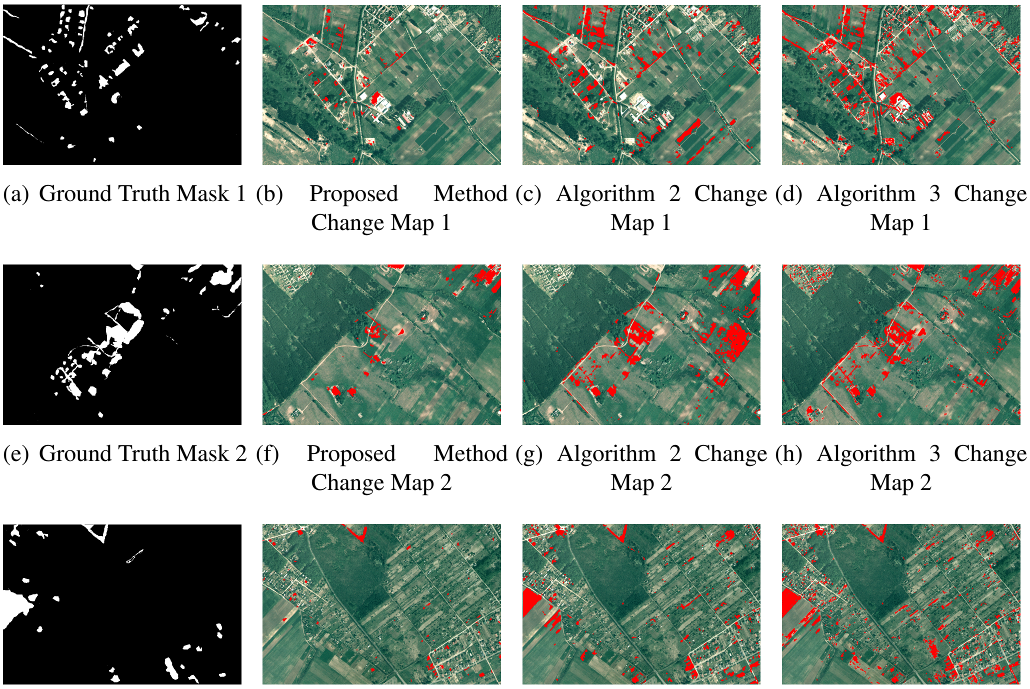
(a) Proposed Method Change Map 1 (b) Algorithm 2 Change Map 1 (c) Algorithm 3 Image Change Map 1

(d) Proposed Method Change Map 2 (e) Algorithm 2 Change Map 2 (f) Algorithm 3 Change Map 2

Figure 4.19: Visual Comparison of Proposed Algorithm on Google Earth Images

When all change masks in Figure 4.20 are inspected, it can be seen that Algorithm 2 and 3 label more change objects correctly than the proposed method. However, change masks produced by Algorithm 2 and 3 include more mislabeled change regions. These mislabeled regions are generally caused by illumination and shadow differences. Thus, it can be said that Algorithm 2 and 3 are more vulnerable to illumination and shadow changes than the proposed method. Comparisons are also made on image pairs from SZTAKI Airchange dataset and resulting change maps are shown in Figure 4.21. Change masks of the proposed method is obtained with heuristic thresholding utilizing feature D. Since image pairs from SZTAKI Airchange Dataset have lower resolution and smaller change objects, histogram bin number is selected as 80 in the experiments.

Change masks shown in Figure 4.21 show similar characteristics with the ones shown in Figure 4.20. Change masks produced by Algorithm 2 and 3 label more regions as changes. Naturally, in these change masks, more change objects are labeled correctly when compared



(a) Ground Truth Mask 1 (b) Proposed Method Change Map 1 (c) Algorithm 2 Change Map 1 (d) Algorithm 3 Change Map 1

(e) Ground Truth Mask 2 (f) Proposed Method Change Map 2 (g) Algorithm 2 Change Map 2 (h) Algorithm 3 Change Map 2

(i) Ground Truth Mask 3 (j) Proposed Method Change Map 3 (k) Algorithm 2 Change Map 2 (l) Algorithm 3 Change Map 2

Figure 4.20: Visual Comparison of Proposed Algorithm on SZTAKI Images

to change masks produced by the proposed method. As expected, change masks of Algorithm 2 and 3 include more mislabeled regions than the change masks of the proposed method. No quantitative evaluation is made, but a deduction can be made with respect to visual inspection. Algorithm 2 and 3 generally produce change masks with higher recall values than the proposed method. However, proposed method provides change masks with higher precision scores. According to change masks shown in Figure 4.21, proposed method is more efficient at distinguishing man-made changes. This property can be clearly seen when the change masks of the third image pair in Figure 4.21 are examined. A large field in the left border of the image is ploughed in changed image. Such changes are not in the scope of this study. Change masks obtained with Algorithm 2 and 3 include this region whereas proposed method successfully labels this region as unchanged. In this subsection, only the change masks obtained with heuristic thresholding method are used for comparison. According to visual inspection done in the former section, Figures 4.8, 4.9, 4.14, 4.15, 4.16, K-Means clustering method also provides relatively successful change masks. In these change masks, more regions are labeled as change when compared to heuristic thresholding results. Therefore, they are more similar to change masks obtained with Algorithm 2 and 3. This implication shows that classification method should be selected with respect to application domain and aims.

CHAPTER 5

CONCLUSION

5.1 Summary

A new and robust method for detecting changed samples between two observations of the same scene is proposed in this dissertation document. This methods application to remote sensing area is then presented by developing a complete change detection framework that works on satellite images. This framework combines methods for preprocessing, feature extraction, change classification and postprocessing to obtain a flexible and competitive change detection algorithm. The main contribution of the procedure is the paradigm named as double segmentation. Utilization of double segmentation helps to create change masks in an object based approach, without requiring prior information about objects in question.

The preprocessing part presents methods used for normalizing and registering input image pairs. A similarity measurement metric is also included in the preprocessing step. With preprocessing, input image pairs are prepared for actual change detection procedure.

Four different features; intensity, ratio, range filter and saliency map difference values are used for classification. These features separate and combined usages for change detection are compared. Proposed double segmentation method is utilized in the feature extraction step. Instead of directly using feature values for each pixel in classification, segmentation results for image pairs are integrated into feature extraction process. Original and changed images are segmented using optimal parameters, producing two different segmentation results as L_1 and L_2. Then for each segment, mean feature values are calculated. This mean feature values are treated as samples in the classification step. Since there are two different segmentation results, classification is done separately for two different feature vectors. This procedure results in two intermediate change maps for the scene. Then, these change maps are intersected to obtain the final change map. According to quantitative and visual evaluations, which are presented in the former section, double segmentation approach has two main strengths. First, using a segment based approach creates rigid change maps that reflects geometric characteristics of change objects without requiring any prior information. This property is clearly seen when the visual results of the proposed algorithm is compared with the pixel-based methods in the former section. Secondly, obtaining two different change maps with respect to two different segmentations produces final change maps with higher precision scores. Intensity difference values are emerged as the most distinguishing feature for change detection in the experiments. Utilization of other features, especially range filter difference values, generally increase the change detection performance.

Three different methods, K-Means Clustering, heuristic thresholding and Expectation-Maximization are used for classification. Unlike K-Means Clustering and Expectation-Maximization methods, which are well known in the literature, heuristic thresholding is developed for this study. Experiments conducted on artificial and real world images show that, each classification method has its advantages. K-Means Clustering and heuristic thresholding generally produce change masks with high recall scores whereas, change masks obtained with EM have high precision scores. These experiments show that classification method selection should be specific to application domain and aims.

A novel segment elimination and reconstruction method is presented for postprocessing. Segment elimination step aims to detect and eliminate segments which show similar geometric characteristics on segmentations of original and changed images. A metric for defining the similarity of segments is presented. Segment reconstruction step aims to enhance the change maps obtained with double segmentation. Since the final change map is generated by the combination of two change maps, resulting connected components generally do not inherit the geometric characteristics of the change objects. In order to enhance such results, connected components in the change map are inspected and reconstructed with respect to segmentation result of the changed image. Visual comparisons of raw and postprocessed change masks show that segment reconstruction provides change masks which represent change object geometries significantly better.

5.2 Discussion and Future Work

Proposed algorithm is applied on artificial and real world images. AICD dataset is the artificial one, created for benchmarking different change detection algorithms. It has ground truth information, therefore quantitative evaluation is made on this dataset. Real world images are obtained from Google Earth and SZTAKI Airchange Benchmark dataset. For real world images, change masks are provided for visual inspection.

When compared to other algorithms in the literature [7, 11, 14], proposed algorithm provides higher performance scores on the AICD dataset, regardless of the classification method selection. Average F-Measure scores of 0.63, 0.68 and 0.70 are obtained when K-Means clustering, heuristic thresholding and EM are utilized for classification with optimum configurations, respectively. These performance scores outperform other three algorithms [7, 11, 14] in the literature. Precision and recall scores of mentioned classification methods strengthen the claim that classification method selection must be done with respect to application domain and purposes.

According to visual inspection done on change masks, K-Means and heuristic thresholding methods provide more meaningful results when compared to EM method on real world images. EM method produces change masks with high precision scores on real world images. However, many changed objects are not labeled in change masks obtained with EM method. Comparison on the real world images shows that proposed method is more successful on distinguishing man-made changes than the algorithms present in the literature [7, 11, 14].

Conducted experiments have shown some flaws of the proposed algorithm. First of all, preprocessing procedure is not sufficient in detecting and eliminating hard shadows. These shaded regions are generally mislabeled as changes in experiments. In some cases, they can suppress the detection of actual change objects. Utilization of a procedure that can detect and eliminate such shaded regions can improve the change detection results.

Second issue is related to the newly proposed heuristic thresholding algorithm. With the current implementation, heuristic thresholding is highly vulnerable to outliers in feature distribution. More importantly, this method works on singular features. Experiments with other methods show that combining different features can enhance final results. Different histograms can be combined into one by the introduction of heuristics. Such an approach should be defined and its parameters need to be optimized by experimenting to obtain more robust results.

Double segmentation procedure creates two different change maps with respect to segmentation results of original and changed images. In the current implementation, final change map is computed by intersecting these two intermediate change maps. In some cases, intersection eliminates correctly found change regions. Such regions can be preserved by utilizing a more probabilistic approach for the integration of intermediate change maps, rather than intersection. Instead of binary change masks, change maps that reflect the change possibility values can be computed. Then, final change mask can be constructed by considering the possibilities in intermediate change maps. Such a method can increase the accuracy of final change masks and prevent losing correctly classified objects.

Experiments shows that segment elimination process is not sufficient for eliminating all mislabeled regions. Not all unchanged segments have similar geometric properties in different segmentations. Therefore, many mislabeled regions persist after elimination step. A second elimination step should be introduced for eliminating such regions. Man-made objects generally have distinctive geometric characteristics that make them distinguishable from natural regions. Such properties can be used to eliminate remaining false changes.

REFERENCES

[1] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *Computer Vision–ECCV 2006*, pages 404–417. Springer, 2006.

[2] Y. Bazi, L. Bruzzone, and F. Melgani. An unsupervised approach based on the generalized gaussian model to automatic change detection in multitemporal sar images. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(4):874–887, 2005.

[3] C. Benedek, X. Descombes, J. Zerubia, et al. Building extraction and change detection in multitemporal aerial and satellite images in a joint stochastic approach. 2009.

[4] C. Benedek and T. Szirányi. Change detection in optical aerial images by a multilayer conditional mixed markov model. *Geoscience and Remote Sensing, IEEE Transactions on*, 47(10):3416–3430, 2009.

[5] K. Bennett, A. Demiriz, et al. Semi-supervised support vector machines. *Advances in Neural Information processing systems*, pages 368–374, 1999.

[6] T. Blaschke. Object based image analysis for remote sensing. *ISPRS journal of photogrammetry and remote sensing*, 65(1):2–16, 2010.

[7] N. Bourdis, D. Marraud, and H. Sahbi. Constrained optical flow for aerial image change detection. In *Geoscience and Remote Sensing Symposium (IGARSS), 2011 IEEE International*, pages 4176–4179. IEEE, 2011.

[8] F. Bovolo and L. Bruzzone. A theoretical framework for unsupervised change detection based on change vector analysis in the polar domain. *Geoscience and Remote Sensing, IEEE Transactions on*, 45(1):218–236, 2007.

[9] F. Bovolo, L. Bruzzone, and M. Marconcini. A novel approach to unsupervised change detection based on a semisupervised svm and a similarity measure. *Geoscience and Remote Sensing, IEEE Transactions on*, 46(7):2070–2082, 2008.

[10] F. Bovolo, G. Camps-Valls, and L. Bruzzone. A support vector domain method for change detection in multitemporal images. *Pattern Recognition Letters*, 31(10):1148–1154, 2010.

[11] L. Bruzzone and D. F. Prieto. Automatic analysis of the difference image for unsupervised change detection. *Geoscience and Remote Sensing, IEEE Transactions on*, 38(3):1171–1182, 2000.

[12] L. Bruzzone and D. F. Prieto. An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images. *Image Processing, IEEE Transactions on*, 11(4):452–466, 2002.

[13] T. Caliński and J. Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

[14] T. Celik. Unsupervised change detection in satellite images using principal component analysis and< formula formulatype=. *Geoscience and Remote Sensing Letters, IEEE*, 6(4):772–776, 2009.

[15] T. Celik. Change detection in satellite images using a genetic algorithm approach. *Geoscience and Remote Sensing Letters, IEEE*, 7(2):386–390, 2010.

[16] P. S. Chavez. Image-based atmospheric corrections-revisited and improved. *Photogrammetric engineering and remote sensing*, 62(9):1025–1035, 1996.

[17] C.-C. Chen and H.-T. Chu. Similarity measurement between images. In *Computer Software and Applications Conference, 2005. COMPSAC 2005. 29th Annual International*, volume 2, pages 41–42. IEEE, 2005.

[18] Y. Cheng. Mean shift, mode seeking, and clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(8):790–799, 1995.

[19] 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, 2002.

[20] X. Dai and S. Khorram. Remotely sensed change detection based on artificial neural networks. *PE & RS- Photogrammetric Engineering and Remote Sensing*, 65(10):1187–1194, 1999.

[21] D. L. Davies and D. W. Bouldin. A cluster separation measure. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (2):224–227, 1979.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[23] R. D. Eastman, J. Le Moigne, and N. S. Netanyahu. Research issues in image registration for remote sensing. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[24] A. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proceedings of the IEEE*, 90(7):1151–1163, 2002.

[25] O. Faugeras, P. Fua, B. Hotz, R. Ma, L. Robert, M. Thonnat, and Z. Zhang. Quantitative and qualitative comparison of some area and feature based stereo algorithms. *W. F orstner and St. Ruwiedel, editors, Robust Computer Vision. Wichmann*, 1992.

[26] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[27] J. Freixenet, X. Muñoz, D. Raba, J. Martí, and X. Cufí. Yet another survey on image segmentation: Region and boundary information integration. In *Computer Vision—ECCV 2002*, pages 408–422. Springer, 2002.

[28] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *Information Theory, IEEE Transactions on*, 21(1):32–40, 1975.

[29] A. Ghosh, N. S. Mishra, and S. Ghosh. Fuzzy clustering algorithms for unsupervised change detection in remote sensing images. *Information Sciences*, 181(4):699–715, 2011.

[30] S. Gilles. *Robust description and matching of images*. PhD thesis, University of Oxford, 1998.

[31] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. In *Advances in neural information processing systems*, pages 545–552, 2006.

[32] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988.

[33] J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

[34] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*, volume 2. Cambridge Univ Press, 2000.

[35] P. J. Howarth and G. M. Wickware. Procedures for change detection using landsat digital data. *International Journal of Remote Sensing*, 2(3):277–291, 1981.

[36] J. Im, J. Jensen, and J. Tullis. Object-based change detection using correlation image analysis and image segmentation. *International Journal of Remote Sensing*, 29(2):399–423, 2008.

[37] L. Itti and C. Koch. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision research*, 40(10):1489–1506, 2000.

[38] T. Kadir and M. Brady. Saliency, scale and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001.

[39] R. J. Kauth and G. Thomas. The tasselled cap–a graphic description of the spectral-temporal development of agricultural crops as seen by landsat. In *LARS Symposia*, page 159, 1976.

[40] W. J. Krzanowski and Y. Lai. A criterion for determining the number of groups in a data set using sum-of-squares clustering. *Biometrics*, pages 23–34, 1988.

[41] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[42] D. Lu, P. Mausel, E. Brondizio, and E. Moran. Change detection techniques. *International journal of remote sensing*, 25(12):2365–2401, 2004.

[43] L. Luccheseyz and S. Mitray. Color image segmentation: A state-of-the-art survey. *Proceedings of the Indian National Science Academy (INSA-A). Delhi, Indian: Natl Sci Acad*, 67(2):207–221, 2001.

[44] W. A. Malila. Change vector analysis: an approach for detecting forest changes with landsat. In *LARS Symposia*, page 385, 1980.

[45] M. Mesbah. Gradient-based optical flow: a critical review. In *Signal Processing and Its Applications, 1999. ISSPA'99. Proceedings of the Fifth International Symposium on*, volume 1, pages 467–470. IEEE, 1999.

[46] F. Pacifici. Change detection algorithms: State of the art. *URL: http://www. disp. uniroma2. it/earth_observation/pdf/CD-Algorithms. pdf (Accessed on 4 November, 2011)*, 2007.

[47] F. Pacifici and F. Del Frate. Automatic change detection in very high resolution images with pulse-coupled neural networks. *Geoscience and Remote Sensing Letters, IEEE*, 7(1):58–62, 2010.

[48] N. R. Pal and S. K. Pal. A review on image segmentation techniques. *Pattern recognition*, 26(9):1277–1294, 1993.

[49] E. Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076, 1962.

[50] B. T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975.

[51] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: a systematic survey. *Image Processing, IEEE Transactions on*, 14(3):294–307, 2005.

[52] P. Rosin. Thresholding for change detection. In *Computer Vision, 1998. Sixth International Conference on*, pages 274–279. IEEE, 1998.

[53] P. L. Rosin. Edges: saliency measures and automatic thresholding. *Machine Vision and Applications*, 9(4):139–159, 1997.

[54] P. J. Rousseeuw and A. M. Leroy. *Robust regression and outlier detection*, volume 589. Wiley, 2005.

[55] A. Singh. Review article digital change detection techniques using remotely-sensed data. *International journal of remote sensing*, 10(6):989–1003, 1989.

[56] S. Theodoridis and K. Koutroumbas. Pattern recognition. *Academic Press*, 525:92101–4495, 1998.

[57] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`, 2008.

[58] V. Walter. Object-based classification of remote sensing data for change detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58(3):225–238, 2004.

[59] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4):600–612, 2004.

[60] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and vision computing*, 21(11):977–1000, 2003.