GLOBAL APPEARANCE BASED AIRPLANE DETECTION FROM SATELLITE IMAGERY

A THESIS SUBMITTED TO
THE GRADUATESCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLEEASTTECHNICALUNIVERSITY

BY

DUYGU ARSLAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JULY 2012

Approval of the Thesis

GLOBAL APPEARANCE BASED AIRPLANE DETECTION FROM SATELLITE IMAGERY

Submitted by **DUYGU ARSLAN** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering, Middle East Technical University,** by,

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

Prof. Dr. İsmet Erkmen
Head of Department, **Electrical and Electronics Engineering** _____

Prof. Dr. A. Aydın Alatan
Supervisor, **Electrical and Electronics Eng. Dept., METU** _____

Examining Committee Members

Prof. Dr. Uğur Halıcı
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. A. Aydın Alatan
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. Gözde Bozdağı Akar
Electrical and Electronics Engineering Dept., METU _____

Dr. Kubilay Pakin
ASELSAN _____

Dr. Emre Başeski
HAVELSAN _____

Date: 26.07.2012

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, Lastname : Duygu Arslan

Signature :

ABSTRACT

GLOBAL APPEARANCE BASED AIRPLANE DETECTION FROM SATELLITE IMAGERY

Arslan, Duygu

M.S., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. A. Aydın Alatan

July 2012, 79 pages

There is a rising interest in geospatial object detection due to not only the complexity of manual processing of such huge amount of data provided by high resolution satellite imagery but also for military application needs. A fundamental and yet state-of-the art approach for object detection is based on methods that utilize the global appearance. In such a holistic approach, the information of the object class is aimed to be modeled as a whole in the learning phase. And during the classification, a decision is taken at each window of the test image. In this thesis, two different discriminative methods are investigated for airplane detection from satellite images. In the first method, Haar-like features are used as weak classifiers for the airplane class representation. Then the AdaBoost learning algorithm is used to select the critical visual features that represent the airplanes best. Finally, a cascade of classifiers is constructed in order to speed-up the classifier. In the second method, a computationally efficient appearance-based algorithm for airplane detection is presented. An operator exploiting the edge information via gray level differences between the target and its background is constructed with Haar-like polygon regions using the shape information of the airplane as an invariant. The airplanes matching the operator are supposed to yield higher responses around the centroid of the object. Fast evaluation of the operator is achieved by means of integral image. The proposed algorithm has promising

results in terms of accuracy in detecting aircraft type geospatial objects from satellite imagery.

Keywords: Haar features, AdaBoost, object detection, global appearance, integral image.

ÖZ

UYDU GÖRÜNTÜLERİNDEN BÜTÜNSEL GÖRÜNÜŞ TEMELLİ UÇAK TESPİTİ

Arslan, Duygu

Yüksek Lisans, Elektrik-Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. A. Aydın Alatan

Temmuz 2012, 79 sayfa

Yüksek çözünürlüklü uydu görüntüleri ile sağlanan büyük miktardaki verinin elle işlenmesinin zorluğu ve askeri uygulama ihtiyaçları dolayısıyla uydu görüntülerinden otomatik nesne tanıma problemine yükselen bir ilgi vardır. Nesne tespiti konusunda temel ve geçerli yöntemlerden birisi de bütünsel görünüş temelli yaklaşımlardır. Böyle bir bütüncül yaklaşım içinde, nesne sınıfının bilgilerinin öğrenme aşamasında bir bütün olarak modellenmesi hedeflenmektedir. Sınıflandırma sırasında, test resminin her bir penceresinde karar verilir. Bu tezde, iki farklı ayrımcı yöntem uydu görüntülerinden uçak tespiti için incelenmiştir. İlk yöntemde, Haar benzeri öznitelikler uçak sınıfını betimlemek üzere zayıf sınıflandırıcı olarak kullanılmaktadır. Daha sonra AdaBoost algoritması uçakları en iyi temsil eden görsel öznitelikleri belirlemek için kullanılır. Son olarak, hız artırımı için sınıflandırıcılar çoklu bir sırayla kullanılarak oluşturulur. İkinci yöntemde, uçak tespiti için hesaplama açısından verimli bir görünüm tabanlı algoritma sunulmaktadır. Hedef ve arka plan arasındaki gri yoğunluk seviyesi farklılıkları üzerinden kenar bilgisini kullanan bir operatör, bir değişmez olarak uçağın şekil bilgileri kullanılarak Haar benzeri çokgen bölgeler ile oluşturulmuştur. Operatör ile eşleşen uçakların, nesne ağırlık merkezi etrafında yüksek yanıtlar vermesi öngörülmüştür. Operatörün hızlı değerlendirilmesi entegral görüntü ile elde edilir. Önerilen algoritma uydu

görüntülerinden uçak tipi yer uzamsal nesneleri algılamada doğruluk açısından umut verici sonuçlar vermiştir.

Anahtar Kelimeler: Haar öznitelikleri, AdaBoost, nesne tespiti, bütünsel görünüm, entegre görüntü.

To My Parents

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my thesis adviser, Prof. Dr. A. Aydın Alatan for his guidance, patience, broad vision and advice. I regard it as a rewarding and thrilling experience to work with him in this thesis study.

I would also like to thank all the people in Multimedia Research Group for making the office a great research environment and sharing valuable ideas.

I thank the source of funding I have throughout the study of this thesis provided by TUBITAK.

Special thanks to Ulya Bayram for her friendship, support, great patience and motivation.

Finally, I would like to thank my family for their endless love and support.

This work is partially funded by HAVELSAN Inc.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

LIST OF TABLES

TABLES

CHAPTER 1

INTRODUCTION

Thanks to today's advanced satellite technology, satellite image of an area could be captured and stored easily in electronic format in a computer. However, unfortunately, a computer cannot interpret the image in a higher semantic level as humans do. Furthermore, the huge amount of data coming from the satellites causes a requirement for automatic systems that could replace a human operator for some military purposes, such as target detection and recognition. In this thesis, the aim is to be able to make automatic detection and localization of airplanes in a satellite image by using global appearance methods. The global appearance based methods, being computationally very simple and efficient, have proven to be robust to shadowing effects and background clutter.

1.1 Overview of the Thesis

In this thesis, the problem of automatic airplane detection from satellite imagery is studied. The proposed solutions exploit global appearance-based methods.

In the first method, the global representation of the airplanes is achieved by the help of popular Haar-like rectangular features. The fast computation of those features is achieved by integral image representation for both the training and classification phases. In order to reduce the excessive number of features, a variant of boosting algorithm, namely *AdaBoost* is used. Using various positive

and negative training examples, at each iteration of the AdaBoost algorithm a critical visual feature is added to the classifier so that the final classifier is a combination of features which gives a holistic representation. Finally, the execution speed of the detector is increased by cascading some strong classifiers so that more complex stages only focus on complicated parts of the image. For scale invariant detection, unlike many existing methods that rescale the image, the detector is rescaled which further increases the speed. Multiple classifiers are also trained for in-plane rotation invariant detection.

In the second method, using the "silhouette" information of the airplane class as an invariant, an operator (filter) is presented. The operator, which is constructed using Haar-like polygon regions, exploits the shape information of the airplane. Integral image paradigm is utilized for faster evaluation of this operator given a test image. The final decision of detection is obtained after clustering the points that yield higher responses to the operator.

1.2 Literature Review

In general, there exists an extensive literature for the object recognition problem. Among various methods proposed over years for different problems, those of the state-of-the-art approaches that could be utilized for object detection from satellite images could be categorized as follows:

- Local Feature (Appearance) Based Approaches
- Global Appearance Based Approaches
- Shape Based Approaches

Unfortunately, there has been relatively limited literature available for object detection problem from aerial or high resolution satellite imagery. In the following three sections, a brief review is presented for both the efforts that

construct the basis for each one of the aforementioned three approaches as well as the studies that exploit those methods for geospatial object detection problem.

1.2.1 Local Feature (Appearance) Based Approaches

In general, local feature based approaches are mainly composed of two steps: region of interest detection together with its description and classification. A local feature is extracted from a highly repetitive salient region, named as *interest point*, characterized by corners, edges etc. Some of the most popular and successful region of interest *detectors* in terms of repeatability, runtime and invariance to affine distortions, scale and illumination changes can be listed as follows: Harris or Hessian point based detectors, Difference of Gaussian (DoG) Points detector, Maximally Stable External Regions (MSER), Entropy Based Salient Region Detector (EBSR), Intensity and Edge Based Regions (IBR, EBR) [1]. A local feature descriptor represents the region that is identified by region of interest detector. The most popular *descriptors* in the literature are Scale Invariant Feature Transform (SIFT), Gradient Location Orientation Histograms (GLOH), complex and steerable filters and Locally Binary Patterns. An extensive study for comparison of those descriptors is presented in [1]. Once the descriptions of region of interest points are obtained, the object class is then learned using some popular discriminative classifiers, such as Support Vector Machines (SVM) [2], Boosting [3], linear discriminant analysis (LDA) [4] etc.

As a particular technique that aims automatic target detection from aerial imagery, Tao et al. [5] propose a new method that exploits a set of SIFT keypoints to describe an airport for airport detection from high-spatial-resolution satellite images. In order to discard the redundant matched points and locate the possible region of candidates containing the target, after obtaining matched keypoints, a novel region location algorithm is proposed that exploits the clustering information from matched SIFT keypoints and the region information that is

extracted through image segmentation. By applying the prior knowledge to the candidate regions airport recognition is achieved.

Similar to the previous method by Tao et al. [5], Sahli et al. [6] present a local feature based vehicle detection approach from high resolution aerial imagery. The authors exploit SIFT to extract keypoints in the image. After obtaining keypoints, the local structure in the neighboring of those points are described by 128 gradient orientation based features. In order to be able to predict whether the SIFT keypoints belong to car structure in the image, a Support Vector Machine (SVM) is utilized for creating a car model. Finally, the car labeled collection of SIFT keypoints are clustered in the geometric subspace into subsets such that each subspace is associated to one car.

In another local feature based approach, Sun et al. [7] propose a novel procedure to solve the problem of geospatial object detection from high resolution satellite images. By applying a multiscale segmentation algorithm first, the authors represent each image as a segmentation tree. Then, all of the tree nodes are described as coherent groups, instead of binary classified values. After obtaining the subcategories, which are found by selecting the maximally matched subtrees, they organize those subcategories to learn the embedded taxonomic semantics on object categories. This approach allows categories to be defined recursively and express both explicit and implicit spatial configuration of categories. Finally, using the learned taxonomic semantics; detection, recognition and segmentation of the geospatial objects in a new image are simultaneously conducted.

As a general remark, it can be argued that local feature based approaches initially proposed for matching and recognition purposes of exactly the same object in another environment possibly in a different orientation. Considering the possible challenges faced in a satellite image, such as shadows and background clutter, which drastically changes a local appearance, and hence the local features, a local feature based approach is not robust. All of the above approaches utilize local

feature based methods which mostly tend to detect keypoints in a geospatial object caused by self-shadow, which is repeatable in some specific category of the object class, but not all others. The major drawback of all these work is lack of keypoint repeatability caused by intra-class variation of object classes. Furthermore, most of the aforementioned presented work is limited to be applicable to high resolution aerial images in which details are identifiable better than regular satellite imagery.

1.2.2 Global Appearance Based Approaches

Appearance based approaches, which could also be possibly named as classifier based methods for object detection framework formulate the object detection/recognition as a classification problem. The image is partitioned into a set of overlapping windows and a decision is taken at each window about whether it contains a target object or not. The problem can be formulated as a binary classification which aims at learning an object class and discriminating it from background. The classification function is learned through a set of labeled positive and negative examples together with some features extracted from them.

An appearance based system is typically composed of two parts: the features that represent a target class and a classifier that discriminates a target object from background. Haar wavelets, Haar filters (rectangle features), Histogram Distance on Haar Regions (HDRD), edges together with chamfer distance, edge fragments and Histogram of Oriented Gradients (HOG) are used as features in [17], [12, 13], [18], [19], [11, 20] and [21, 29], respectively. As classifiers Neural Networks, SVM or Boosting are utilized in [10, 25, 26, 27], [21, 28, 29] and [12, 13, 24], respectively.

The early work in this area starts with face detection problem. Given some description of a visual object, which can be linguistic, pictorial, procedural etc., Fischler and Elschlager [8] formulate the problem as to find that object (face) in

an actual photograph through image-matching process. The visual objects are described by breaking down them into a number of more primitive parts and specifying a range of spatial relations, named as “springs”, that the primitive parts should satisfy for the object to be present. These descriptions used by the authors for terrain scenes employ texture and shape components. Their work mainly aims at finding the satisfactory set of primitives which can be used as the basis for the reference (template) component description. In their work, the authors achieved limited results due to lack of computational resources.

A complementary approach utilizing global appearance characteristics is presented by Turk and Pentland [9]. The authors take the advantage that the faces are present upright in the images so that they may be described by a small set of 2D characteristic views. In mathematical terms, they find the principal components, in other words the eigenvectors of the covariance matrix of the set of face images. These eigenvectors are thought of a set of features that characterize the variation between face images and the main idea of the PCA analysis is to find vectors best account for the distribution of the face images within the entire image space. Since the eigenvectors of the covariance matrix of the original face images are face-like in appearance, they are referred to as “eigenfaces”. For the testing scenario, the eigenfaces are computed by projecting the input image onto each of the eigenfaces and then determine whether the input is a face by checking sufficient closeness of the image to the face space. Modeling the image globally, a PCA based system has some drawbacks such as illumination change, background clutter and occlusion each of which would cause a change in the eigenvector representation of the image.

A state-of-the art appearance based visual object detection framework is proposed by Viola and Jones [12, 13] that is able to process images rapidly achieving high detection rates. The authors introduce three contributions: Integral Image concept, which allows the rectangle features to be computed quickly by the help of look-up tables; a learning algorithm that is capable of selecting critical visual features for

detection, and finally, a method for combining classifiers in a cascade that allows quick processing of the test images in real time. The features they utilize are rectangle features, which are similar to 2D Haar wavelets that are able to represent the gray-scale intensity differences within their region. During the training phase, Haar features are computed for all training images for all positions and scales by taking the difference of sum of pixels in black and white regions. Then, these values are fed to the AdaBoost [22, 23] classifier and the most discriminative features are selected by AdaBoost in a cascaded system. The outputs of the classifier are then combined linearly as a final strong classifier. Although, the approach is quite effective in terms of computation with a high efficiency, it is applicable for the cases, where there is a specific texture-color change within an object or across an object and background.

Similar to the method proposed by Viola and Jones [12, 13], Papageorgiu and Poggio [17] present a system for object detection, specifically human and face classes, that describes an object class in terms of an overcomplete set of local, oriented, multiscale intensity differences between adjacent regions that is computable as a Haar wavelet transform. The authors transform the input image from pixel space to the space of local and oriented intensity difference features, and then, use them as input to a SVM classifier.

In addition, Heisele et al. [14] present an object detection framework that uses Support Vector Machine (SVM) as classifiers. As features they use the view-based gray-level pixel values that are generated from 19x19 face and non-face images. In the first step of their algorithm they reduce the features by employing a wrapper method [15, 16] that attempt to search through the space of features using the criterion of the classification algorithm. Similar to [12, 13] they build a hierarchy of classifiers to speed-up the system.

A similar SVM based method is proposed by Dalal& Triggs [21] that utilize locally normalized histogram of gradient orientation features, called *Histogram of Oriented Gradients* (HOG) descriptors in a dense overlapping grid. In this way,

the object appearance and shape are characterized well by the distribution of local intensity gradients or edge directions. The authors study those features for human detection. As a test case, an SVM based classifier is adopted.

The most important limitation of the above methods proposed for face and human detection problem is that the detection is not invariant to rotation of the object; i.e. the faces are only detected when they appear upright on an image. Alternatively, one should train a new classifier for each of the different pose or view of an object class or detection should be performed by rotating the test image.

A particular solution to geospatial object detection problem is proposed by Perrotton et al. [30] by a work that exploits the appearance characteristics of the geospatial objects, such as aircrafts in cluttered background on aerial or satellite images. In order to select discriminating features, a boosting algorithm is utilized. A new descriptor is introduced, Histogram Distance on Haar Regions (HDHR), which is robust to background and target texture variations. Unfortunately, gathering the large amount of data required for AdaBoost learning algorithm is quite hard to achieve considering the unavailability of such large annotated database. Therefore, using image synthesis, the authors generate large amounts of learning data using 3D object model. Yet, the representational power of such data that is to take into account the variability of real operational scenes is controversial.

Alternatively, a new approach for detecting airplanes on panchromatic satellite images is presented by Cai and Su [31]. The authors introduce a circle-frequency filter (CF-filter) that extracts the candidate points of airplane centers. Finally, a clustering algorithm is utilized to locate the airplane centers. The method lacks the global representational power, since the CF-filter takes only the specific intensity change around a circle fitted on an airplane center into account. Hence, the system would naturally yield higher false alarms on regions with man-made objects, lane lines with cross shape arrangements etc.

On the other hand, Moon et al. [33] present a complementary model-based vehicle detection algorithm for aerial parking lot images. The experiments are performed by combining four elongated edge operators to collect edge responses from the sides of a vehicle. As a statistical diagnostic tool for the detection performance, bootstrap is used in this work. It is assumed that the orientation of the parking lot is known in this method, which is the main limitation of the algorithm.

1.2.3 Shape Based Approaches

Shape based approaches rely on some shape characteristics of the objects. In order to obtain these characteristics, an initial segmentation is required. Segmented regions are then described by some popular region based descriptors, such as Hu moments [34], Zernike moments [35], Angular Radial Transform [36] or contour based shape descriptors, such as Fourier descriptor and curvature scale space representation [37]. For the classification of a segmented candidate region, matching of the learned descriptors from the training set is utilized.

In a particular shape based approach, Zhao and Nevatia [38] present a system to detect cars in aerial images. Starting from psychological tests to find important features for human detection of cars; the boundary of the car body, the boundary of the front windshield and the shadow are selected as features. A Bayesian network structure is exploited to integrate all features.

Similarly, Choi et al. [39] proposed an approach for automatic vehicle detection from aerial images. The initial extraction of candidate vehicles is achieved by mean-shift algorithm with the assumption of symmetric character of blob-like car structure. In the subsequent stage, log-polar shape descriptor is used for measuring similarity of the blob to a vehicle.

Both of the methods [38] and [39] are applicable for aerial imagery where the details of an object such as windshield of a car are identifiable.

Another method to describe geospatial objects possibly at any shape is presented by Iisaka et al. [40]. The shape of the object is approximated using expansion of the object with a series of pattern primitives or structural elements and the first few expansion coefficients of a small number of different primitive expansion sets.

In addition, Eikvil et al. [41] introduce segmentation based vehicle detection from high resolution satellite imagery. The resulting regions from segmentation are described by gray level and spatial features. The spatial features they used such as area, compactness, Hu moments, height and width are selected to distinguish cars from other objects using their shapes. Then a rule based classifier is used to discard non-vehicle objects. Finally, the potential vehicles are classified with two different statistical classifiers namely, Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA).

Similar to the method presented by Eikvil et al. [41], Hsieh et al. [42] proposed a method for aircraft type recognition from satellite images. A novel symmetry based algorithm is applied to estimate optimal orientation of the aircraft first. Then distinguishable features are derived and a boosting algorithm is utilized to learn a set of proper weights from training samples for feature integration. The aircraft recognition is handled assuming that the initial detection is achieved which is a hindrance for this method.

Considering all these techniques, one could conclude that shape based methods require an initial segmentation, which is limited to some favorable cases. Unfortunately, in real scenarios, disturbances implied by background clutter, illumination changes, shadows and occlusions by other objects cause shape based approaches to fail. Therefore, segmentation step should always be avoided.

Since characteristics of appearance can explain implicitly the various effects of sensors, textures and no initial segmentation is required, they are more robust compared to local feature based and shape based approaches. Therefore, in this thesis, global appearance based methods are selected as a solution for geospatial object detection problem.

1.3 Outline of the Thesis

The structure of this thesis is as follows: In Chapter 2, a leading global appearance based object detection approach proposed by Viola and Jones [12], is examined. A detailed description of Haar-like rectangle features, integral image representation, a variant of Boosting algorithm to select critical features, cascaded classifier construction and in-plane orientation independent detection scheme are presented. Furthermore, the experiments on the performance of the algorithm are performed for our problem and results are presented.

Chapter 3 is devoted to the main contribution of this thesis to the examined problem. A global appearance based geospatial object detection framework is introduced. The invariant geospatial object operator and the detection algorithm using the operator are explained. The performance tests are conducted for aircraft type geospatial object detection. The experimental results of the proposed algorithm are given.

Finally, in Chapter 4, the conclusions of the thesis are drawn and some future work is discussed.

CHAPTER 2

RELATED WORK ON GLOBAL APPEARANCE BASED OBJECT DETECTION

In this chapter, the problem of mobile object detection from satellite imagery using a state-of-the art global appearance based approach is studied. The detection task is a composition of two parts: feature extraction and classification. Global appearance based visual description of the object class is achieved by simple rectangle features. The features are simple to evaluate, yet, are proven to be capable of representing the object quite efficiently in terms of detection performance, robustness to shadows, noise and change in illumination [12]. The feature selection is based on a machine learning method, namely *AdaBoost*.

The organization of this chapter is as follows: In Section 2.1, the rectangle features together with the approach for fast evaluation of the features for various scales are presented. In Section 2.2, AdaBoost algorithm for selecting important visual features out of an excessive number of features is explained. A cascaded structure for combining classifiers to speed up the overall system is examined. Orientation independent detection procedure is presented in Section 2.3. Finally, the experimental results are presented in Section 2.4.

2.1 Features

Global appearance based features try to model the whole content of an image regarding all the pixels within the image [1]. Conventional global appearance based methods exploit simple statistical measures, such as feature histograms, mean values, etc. or more sophisticated *subspace methods* based on dimensionality reduction techniques, such as Principal Component Analysis (PCA) [43], Independent Component Analysis (ICA) [44] or Non-negative Matrix Factorization (NMF) [45]. Projection of the original data onto a subspace which represents the data optimally and sufficiently is the idea behind the subspace methods [1].

In this thesis, Haar-like rectangle features [12] are exploited, rather than using simple pixel values or dimensionality reduction methods, since this rectangle feature based system operates much faster than other systems while being capable of modeling the object of interest efficiently.

2.1.1 Haar-like Rectangle Features

In this thesis, the reminiscent of Haar basis functions, namely Haar-like rectangle features illustrated in Figure 2.1, which are introduced by Paul Viola [12] and later improved by Rainer Lienhart [46], are utilized. An overcomplete set of features (see Figure 2.1) is obtained by varying the size and the location of the features within a detection window. Such rectangular features are also named as *weak classifiers* [12], since each feature might contribute to the final decision after being compared to a threshold value learned during training and yet, is incapable of discriminating the object from background with adequate detection performance without a combination of several different features.

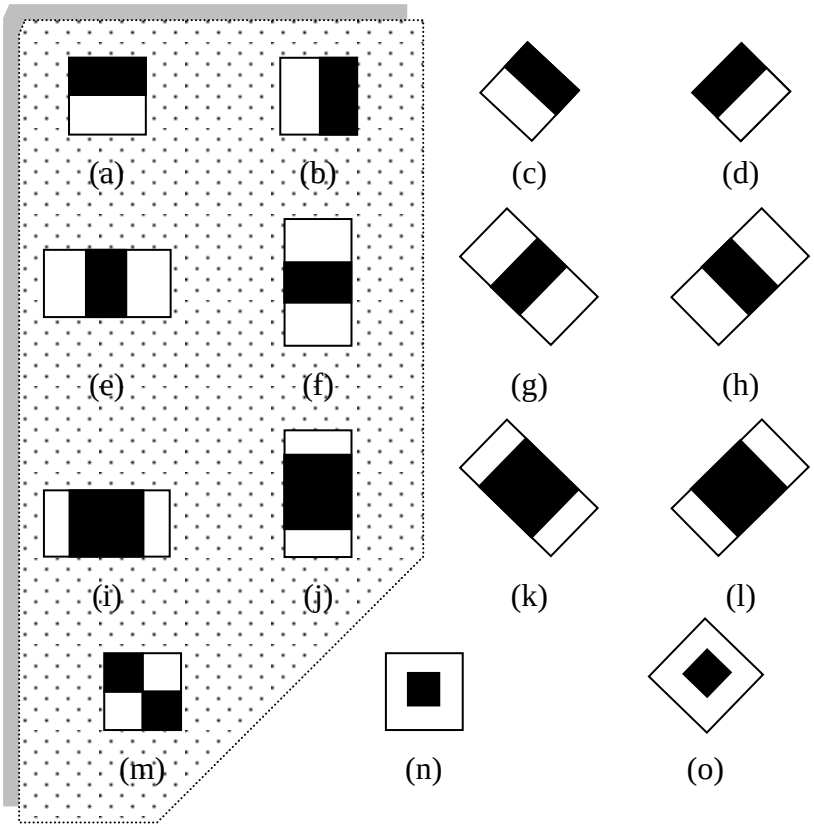
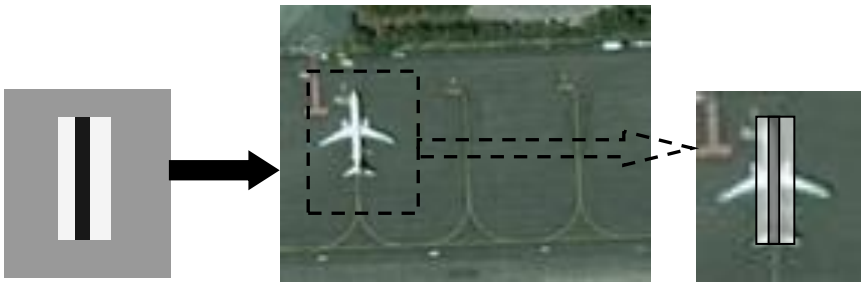


Figure 2.1: Rectangle features (a) and (b): regular edge (two-rectangle) features, (c) and (d) tilted edge (two-rectangle) features (e), (f), (i), (j): regular line (three-rectangle) features, (g), (h), (k), (l) tilted line (three-rectangle) features (m): four-rectangle feature, (n) and (o) center surround features .The feature values are calculated by subtracting the sum of pixels within white region(s) from sum of pixels within black region(s). For a base detection window of size 20x20 pixels, the total number of rectangle features; 125,199; is far greater than the number of pixel elements in the window, 400. Hence the rectangle feature set is overcomplete. (A complete set should consist of linearly independent basis elements). (Highlighted features are introduced in [12] and later expanded with additional tilted and center surround features in [46]).

Haar feature based detection system consists of constructing the mesh of features from the overcomplete set (see Figure 2.1) which best visually describes the object class such that when learned set of features are evaluated on top of the object of interest, it yields higher values. This feature selection process is referred to as *training*. The combination of several single rectangle features can be considered as a representation of the object for computers to perceive and interpret it.

Although the rectangle features are sensitive to edges, lines and such simple structures, they are coarse. However, Viola and Jones [12] empirically show that by generating features of arbitrary aspect ratio and of finely sampled location, they provide a rich representation of images supporting effective learning.

When searching of the object of interest in an image, a detection window is scanned through the test image with, possibly, various Haar-like rectangle features of different sizes and positions within the window which are learned through training. Figure 2.2 illustrates this evaluation process with a single feature.



(a) (b) (c)

Figure 2.2: (a) The detection window is slided across (b) the test image. (c) At each position, the feature with fixed location inside the sub-window is evaluated to determine whether it contains the object, in this example an airplane, or not.

Haar-like rectangle features exploit the object specific appearance characteristics. A representative example is shown in Figure 2.2, where the body region of an airplane is compared to relatively darker background.

The evaluation of the features requires simple pixel intensity addition as shown in Figure 2.1 and threshold comparison. In other words, each pixel in the corresponding rectangle feature has to be visited, which could yield lots of lookups that can be time consuming depending on the area of the selected features. Therefore, in this thesis, the feature evaluation process is accelerated by utilizing Integral Image representation, as proposed in [12] and [46].

2.1.2 Integral Image

An alternative image representation can be constructed by turning a new image into a column and row-wise summed, i.e. double integrated, image. This new representation is called *Integral Image* [12]. The pixel value of the integral image at a location *x, y* is calculated by the summation of the pixels above and to the left of that location. Figure 2.3 demonstrates the construction of integral image by using the following equation:

$$II(x,y) = \sum_{x' \le x,\ y' \le y} I(x',y') \tag{2.1}$$

Where *I*(*x*,*y*) and *II*(*x*,*y*) represent input image and corresponding integral image at location, (*x*, *y*); respectively.

The convolution operation could be accelerated, if derivatives of the operands, e.g. *a* and *b,* can be made sparse [12]. When an invertible linear operation is applied to *a* and its inverse is applied to *b*, the result of the convolution operation does not change:

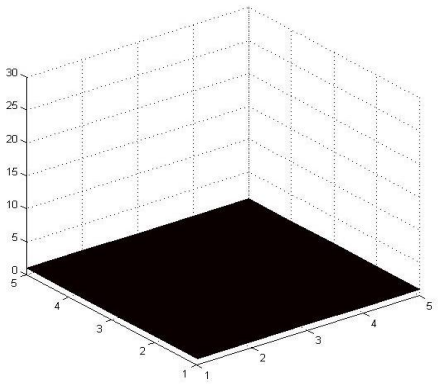
$$a * b = (a'') * \iint b \tag{2.2}$$

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

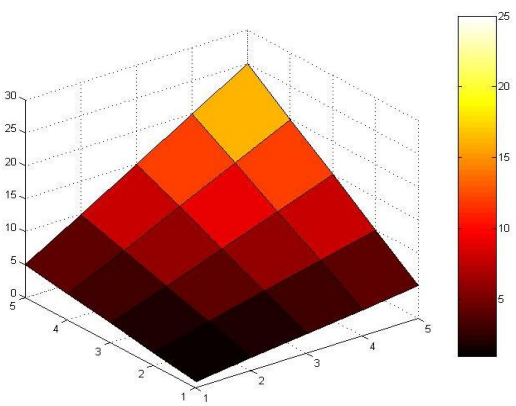
(a)

1	2	3	4	5
2	4	6	8	10
3	6	9	12	15
4	8	12	16	20
5	10	15	20	25

(b)



(c)



(d)

Figure 2.3: Integral Image illustration: (a) Input image in 2D (c) Input image in 3D (b) Integral image in 2D (d) Integral image in 3D.

The computation of rectangle sum could be further accelerated in this framework, if one denotes the rectangle sum as a dot product *r.I*, where *r* and I represent the rectangle box and input image, respectively:

$$r \cdot I = (r'') \cdot \iint I \tag{2.3}$$

The terms, $\iint I$ and (r'') in (2.3) can be interpreted as the integral image of the original image and the second derivative of the rectangle box, or equivalently, four delta functions at corners of the box, respectively. The rectangular box summation procedure is shown in Figure 2.4.

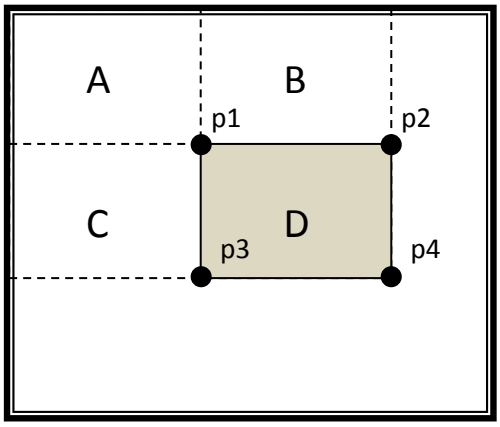


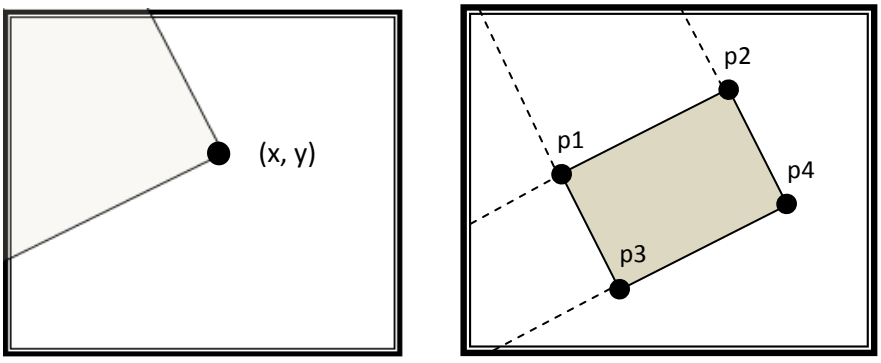
Figure 2.4: The shaded area, D, can be computed in four look-up operations from a table: p4 + p1 – (p2 + p3) if we denote the integral image values at dotted points by p1, p2, p3 and p4 such that p1 = A, p2 = A+B, p3 = A+C and p4 = A+B+C+D.

As stated earlier, in order to enhance the representational power, additional features introduced in [46] (see tilted features in Figure 2.1) are utilized, as well as regular rectangle features (see highlighted features in Figure 2.1) in this work. However, considering its properties, regular integral image representation is only utilizable for axis aligned features. Therefore, in this thesis, additional rectangle features are evaluated via a new auxiliary image representation [46] referred to as

Rotated Integral Image. The calculation scheme of 45º Rotated Integral Image given in (2.4) and tilted rectangular box summation are shown in Figure 2.5.

$$RII(x, y) = \sum_{x' \le x,\ x' \le x - |y - y'|} I(x', y') \tag{2.4}$$

Unlike conventional scale independent detection approaches where a pyramid of images is prepared and a fixed scale detector is used, the detector scaling method is exploited benefitting the computational power of integral image representation. In this study, the detector scans the input image at various scales starting from base resolution. Therefore, one needs to evaluate the combination of rectangle features at several scales. Using Integral Image and Rotated Integral Image, any rectangular sum can be computed in four lookups (see Figures 2.4 and 2.5). Since the two, three and four rectangle features in Figure 2.1 have adjacent rectangle regions, their values can be computed in six, eight and nine lookups, respectively, whereas center surround features are computed in eight lookups at any scale and location. Given the computational efficiency provided by the rectangle features and integral image, the rectangle feature based detection process is faster than any other procedure that requires construction of an image pyramid [12].



(a) (b)

Figure 2.5: (a) 45ºRotated Integral Image (b) Calculation scheme of 45º rotated rectangular box. Shaded area can be computed in four lookup operations: p4 + p1 – (p2 + p3), where points p1, p2, p3 and p4 represent the rotated integral image values at corresponding locations.

2.2 Feature Selection

As stated earlier, the total number of features within the 20x20 base resolution window used in this thesis is equal to 125,199 and this number is approximately 313 times greater than the number of pixels within the window. Computation of all the features associated with the sub-window is quite expensive, even though each feature is evaluated quite efficiently with only a few lookups and arithmetic operations. Furthermore, among all possible features, only a few of them are expected to yield high results, when evaluated on top of the target object class. Therefore, the goal is to find those features that are to form an effective classifier as mentioned earlier.

In this thesis, a modified version of the AdaBoost algorithm introduced in [12], which was originally developed by Freund and Shapire [47], is used for feature selection and classifier training purposes.

2.2.1 AdaBoost Algorithm

Given a positive and negative training set together with a feature set, there are several machine learning algorithms available to learn the classification function, such as Support Vector Machine, Neural Network, Maximum Likelihood Model, etc. [4]. Having shown favorable results compared to single classifier systems, an ensemble based system has the extensive benefits of using several classifiers before making the final decision. The strategy of an ensemble based system is to construct several single classifiers each having errors on different instances and combine their outputs such that the combination can reduce the total error. This process, however, can be achieved with *diverse* set of single classifiers where each classifier has adequately different decision boundary from others [48].

In this thesis, a variant of the state-of-the art ensemble based system, namely AdaBoost [12], which is a more general version of the boosting algorithm introduced in [47] is utilized. AdaBoost learning algorithm is originally used for boosting the classification performance of individual classifiers through weighted majority voting. Using samples from iteratively updated distribution of training data, the algorithm collects weak learning functions and combines them to construct a strong classifier. The simple learning algorithm is called *weak learner*, since it is expected to correctly classify the training data slightly better than chance, i.e. at least 51% of the time. Boosting of the weak learner is achieved by utilizing it to solve a sequence of learning problems where the training data is weighted in each round, so that incorrectly classified instances are more likely to be contained in the consecutive classifier. Hence, increasingly hard-to-classify examples are included more as the iterations increase.

2.2.1.1 Weak Classifier

Drawing the analogy between the weak learner and rectangle features, the AdaBoost algorithm could be considered as a feature selection procedure such that the weak learners are composed of a set of classification functions each of which has a single rectangle feature in our system. Hence, the weak learning algorithm searches for the rectangle feature that separates the object and the non-object class examples in the training set with lowest misclassification. A weak classifier; *h(x,f,p,θ)*, is represented as a quadruple as in the following:

$$h(x, f, p, \theta) = \begin{cases} 1 & if\ pf(x) < p\theta \\ 0 & otherwise \end{cases} \tag{2.5}$$

Where x is a 20 x 20 pixel sub-window of an image, f represents a rectangle feature, p is the polarity and θ is the threshold deciding the classification of x as an object or non-object. Hence, the weak classifiers are simply composed of thresholded single rectangle features.

2.2.1.2 Strong Classifier

A strong classifier is a composition of several weak classifiers each of which are given weights during training based on their classification performance on training set.

The pseudo code of the modified AdaBoost algorithm [12], which is used to select the best possible combination of features to form the strong classifiers out of a total 125,199 features, is shown in Figure 2.6.

The weak classifier selection process proceeds as follows. Each feature is evaluated for all negative and positive training examples and a sorted list based on the feature values is prepared. Using this sorted list, the optimal threshold, θ_t and polarity, p_t for the particular feature can be obtained in a single pass that minimizes the error term ϵ_t in Figure 2.6. The error is simply the summation of the weights of the misclassified examples. After selecting the classifier with minimum error, the selected feature is evaluated for all the examples in the training set and based on the performance of the selected weak classifier, the weights of the examples, which are initialized to be uniform, are updated so that the weights of correctly classified examples are reduced, while the weights of misclassified examples are remained the same. The weights are updated to form a distribution to add up to the value one in each round for normalization. This weight update procedure can be interpreted as a training example set sampling procedure. The penalty for misclassifying an example that is also misclassified by the previous classifier increases as the iterations proceed. Hence, the harder examples to classify by the current weak classifier are given more attention in the consecutive weak classifier. The strong classifier $C(x)$ is a summation of selected weak classifiers each of which are weighted by a scalar, α_t based on their classification error.

- Given example images $(x_1, y_1),\ldots, (x_n, y_n)$ where $y_i = 0, 1$ for negative and positive images, respectively.
- Initialize weights $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ for $y_i = 0, 1$, respectively, where m and l are the number of negative and positive images, respectively.
- For $t = 1,\ldots,T$:
  1. Normalize the weights, $w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$
  2. Select the best weak classifier with respect to the weighted error
  $$\epsilon_t = min_{f,p,\theta} \sum_i w_i |h(x_i, f, p, \theta) - y_i|$$
  3. Define $h_t(x) = h(x, f_t, p_t, \theta_t)$ where f_t, p_t and θ_t are the minimizers of ϵ_t .
  4. Update weights:
  $$w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$$
  where $e_i = 0$ if example x_i is classified correctly, $e_i = 1$ otherwise and $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$.
- The final strong classifier is:
$$C(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & otherwise \end{cases}$$
where $\alpha_t = \log \frac{1}{\beta_t}$.

Figure 2.6: Modified AdaBoost algorithm: The final strong classifier is a weighted linear combination of T weak classifiers. The weights are inversely proportional to the training errors [12]. The AdaBoost requires $\epsilon_t < 1/2$ and hence, $0 < \beta_t < 1$.

The training error *E*, of the strong classifier is proven to be upper bounded by the following relation [47]:

$$E < 2^T \prod_{t=1}^{T} \sqrt{\epsilon_t(1-\epsilon_t)} \tag{2.6}$$

Hence, in each round the ensemble training error, *E* is guaranteed to decrease, since $\epsilon_t < 1/2$. For most of the cases, the error is decreased rapidly. Furthermore, AdaBoost algorithm achieves large margins yielding a better generalization performance. The margin of an example is simply the difference between sum of the weights of classifiers that correctly classify the example and the highest weight of classifier that incorrectly classifies it [48].

Further improvement on the computational efficiency of the detector is achieved by a key concept, namely *cascade of classifiers,* as introduced by Viola and Jones [12].

2.2.2 Cascade of Classifiers

When searching for an object in a given image, most of the sub-windows scanned are considered to be belonging to background. Owing to this fact, Viola and Jones [12] come up with a series of strong classifiers combined in a system which they refer to a as the *attentional cascade*. All the stages in the cascade are trained by AdaBoost algorithm described in the previous section. The cascaded system is designed so that the detector would discard sub-windows belonging to background and pay more *attention* to the sub-windows that pass through the previous stages of the cascade. Such a system discards non-objects at the initial stages of the cascade rapidly without evaluating all the features in the detector. Figure 2.7 illustrates the detection scheme of the cascaded system.

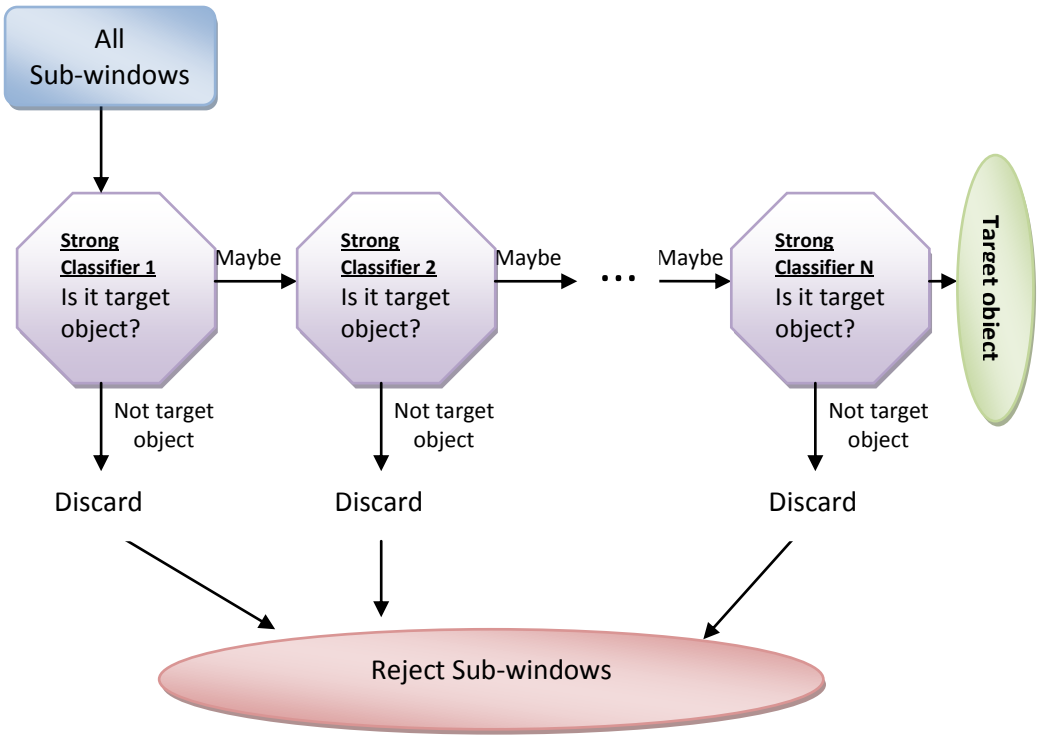


Figure 2.7: A cascade of *N* strong classifiers.

The initial stages of the cascade are designed to have higher detection rates, e.g. 100%, at the expense of higher false positive rates by adjusting the AdaBoost threshold $\frac{1}{2}\sum_{t=1}^{T}\alpha_t$ (see Figure 2.6) to a lower value. Although the detection performance of such a single stage is not sufficient, one would accept that with only a few number of features, it can easily reduce the number of sub-windows for further stages to process.

2.2.2.1 Training The Cascade

For a trained cascade with *N* classifiers, the overall correct detection rate *CD* and false positive rate *FP* of the cascade are equal to:

$$CD = \prod_{i=1}^{N} cd_i\,, \qquad FP = \prod_{i=1}^{N} fp_i \tag{2.7}$$

where cd_i and fp_i represent the correct detection rate and false positive rate of the *i*th classifier, respectively.

Given the overall performance goals in terms of correct detection rate and false positive rate, the corresponding performance rates of each stage in the classifier could be determined by the relation in (2.7). For example, for a *N*-stage classifier, a target correct detection rate *CD* could be achieved, if each of the classifiers in the cascade has a correct detection rate of $\log_N CD$. A correct detection rate of 0.9 could easily be achieved by using 10 strong classifiers each with a correct detection rate of 0.99. Although training a strong classifier with a quite high correct detection rate might sound as a difficult task, it could be made easier, if each stage achieves a relatively high false positive rate, e.g.0.3, making the overall false positive rate equal to 0.6×10^{-9} which is sufficiently low.

In order to meet the user defined performance rates *CD* and *FP*, strong classifiers are trained by using AdaBoost learning algorithm, as explained in Section 2.2.1.2. The number of features for each strong classifier in the cascade is increased by one in each round until the user defined maximum acceptable correct detection rate cd_i and false positive rate fp_i are achieved. A stage *i* in the cascade is trained using all given positive examples in the training set and the false positives of the (*i-1*)*th* classifier (*i=2,…,N*). The algorithm for constructing a cascade of classifiers is given in Figure 2.8

- User selects the maximum acceptable false positive rate *fp* and minimum acceptable correct detection rate *cd* per stage.
- User selects overall target false positive rate *FP*.
- *P* is the set of all positive examples in the training set.
- *N* is the set of all negative examples in the training set.
- $FP_0 = 1$; $CD_0 = 1$
- $i = 0$
- while $FP_i > FP$
    1. $i \leftarrow i+1$
    2. $n_i = 0;\ FP_i = FP_{i-1}$
    3. while $FP_i > fp \times FP_{i-1}$
        - $n_i \leftarrow n_i + 1$
        - Use *P* and *N* to train a classifier with n_i features using AdaBoost
        - Evaluate current cascaded classifier on validation set to determine FP_i and CD_i
        - Decrease threshold for the *i*th classifier until the current cascaded classifier has a detection rate of at least $cd \times CD_{i-1}$
    4. N $\leftarrow \emptyset$
    5. If $FP_i > FP$ then evaluate the current cascaded detector on the set of negative examples and put any false detections into set *N*.

Figure 2.8: Cascaded classifier training [12].

2.3 Orientation Independent Detection

Learning approach explained in previous sections is useful, when detection is limited to a single view of the target object, e.g. upright faces as in the work of Viola and Jones [12]. However, in the attacked problem, where a geospatial object is to be detected from satellite imagery, each of different in-plane rotated object yields a different appearance (an illustration of which is shown in Figure 2.9 for plane class). Hence, these different views of object have to be learned separately, since each different appearance would be represented with different weak learners.

Figure 2.9: Airplanes with different in-plane rotations.

In this thesis, a collection of cascades each of which is trained for a single orientation is constructed. Detection is achieved by collecting the results of each of the cascades on test images. Block diagrams explaining the overall training and detection algorithms are given in Figures 2.10 and 2.11, respectively. Implementation details are explained in the following section.

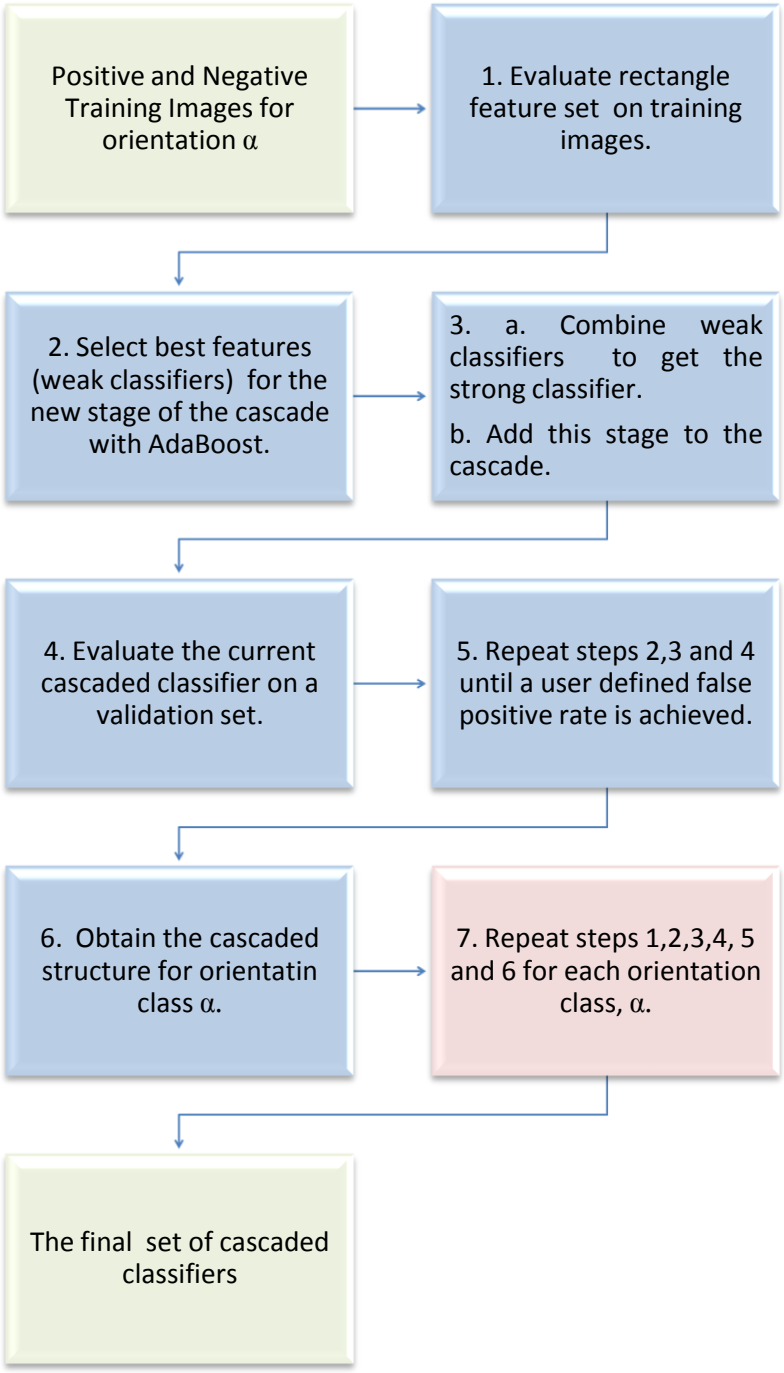


Figure 2.10: Outline of the orientation independent training algorithm.

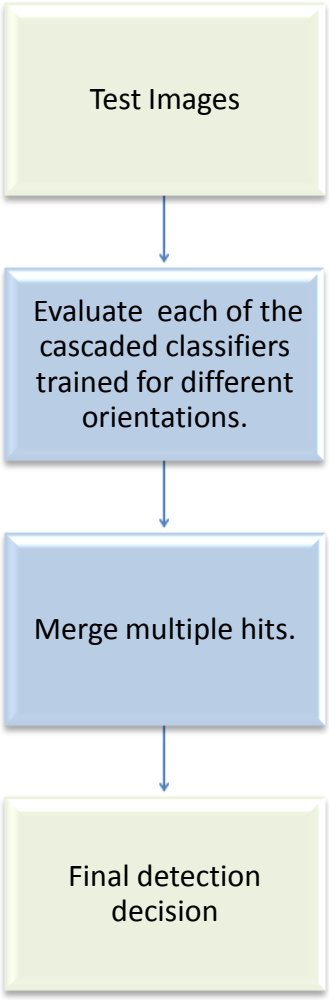


Figure 2.11: Outline of the orientation independent detection algorithm.

2.4 Experimental Results

In this thesis, the tests are conducted for airplane class object. This section describes the details about training the cascaded structures and the performance results on both real and synthetically generated data set, respectively.

2.4.1 Training Dataset Generation

In order to reflect the reality one needs a large amount of annotated learning data. Considering each single training procedure, one needs to follow for differently oriented planes; hence, it is quite difficult to collect sufficiently representative datasets, reaching order of thousands in the number. Therefore, in this thesis, it is suggested to generate hybrid synthetic data which avoids annotation difficulty.

The adopted approach aims to combine the real acquisitions to generate large number of positive training set. For this purpose, various airplane images are collected from Google Earth. Then, those 50 different passenger airplanes are cropped from the images and each of these airplanes is overlaid on 85 different 168 x 168 pixel backgrounds that are randomly selected from various airplane regions. Next, the cropped airplanes, which are directed towards east are clockwise rotated by α degrees (α=15º, 30º, 45º,…, 345º) and the same procedure is applied for different orientations to generate the 24 datasets each with a 4250 positive training example. Some positive examples from the synthetic data sets are shown in Figure 2.12.

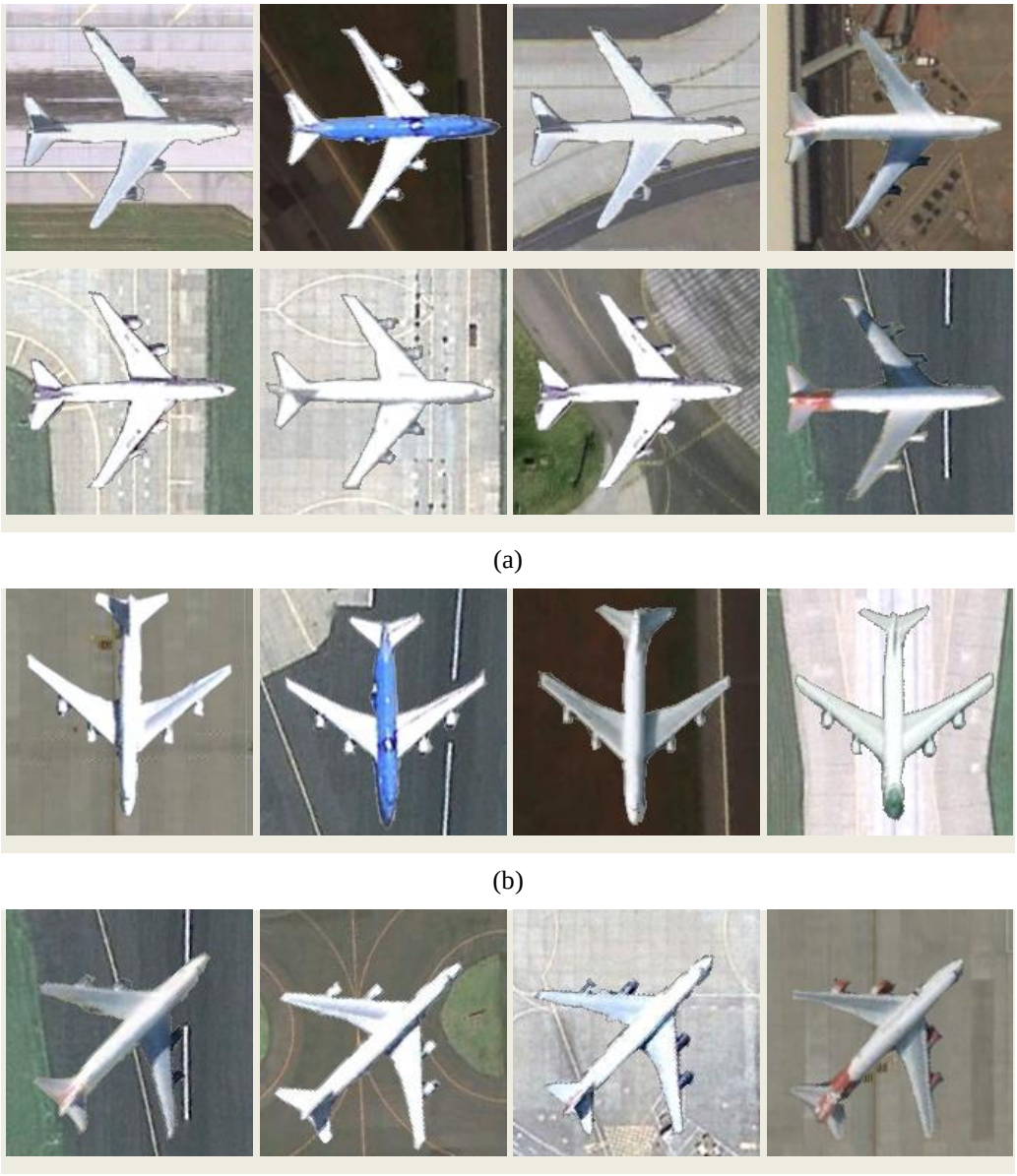
The airplanes are aligned in the middle of 168 x 168 pixel sub-windows. All the positive examples are converted to gray value and variance normalized in order to reduce the effects of different lightning conditions. The variance normalization is achieved by the following relation:

$$f_N(x,y) = \frac{f(x,y) - m_f}{\sigma_f} \quad (2.8)$$

where $f_N(x,y)$, $f(x,y)$, m_f and σ_f represent the 168x 168 pixel variance normalized sub-window, 168x 168 pixel input sub-window, the mean and the standard deviation of the input sub-window, respectively.

After variance normalization, each positive training sub-window is resized to 20 x 20 pixels to reduce the amount of unnecessary information to be learned as well as for fast training purposes, since amount of all possible features within a sub-window is dependent on the size of the sub-window. The total possible amount of rectangle features for a 20 by 20 pixel sub-window is equal to 125,199.

The raw material for generating negative examples is collected from Google Earth images of various airport regions that do not contain airplanes. In this thesis, the detection is aimed to be performed on 0.5 m resolution imagery. Therefore, the raw material is adjusted to be 0.5 m resolution by setting the eye altitude to the sum of the elevation terrain and 685 m of distance which has empirically shown to give the desired resolution in Google Earth. Typical large sized airplanes, such as passenger planes, occupy 150 by 150 pixels in 0.5 m imagery. Therefore, the raw data is cropped into 150 by 150 pixel sub-windows, which the final detector can be expected to meet, to obtain 805 negative examples. Typical negative examples used for training are shown in Figure 2.13.



(a)

(b)

(c)

Figure 2.12: Positive training examples for (a) regular airplanes, (b) 90º rotated and (c) 315º rotated airplanes.

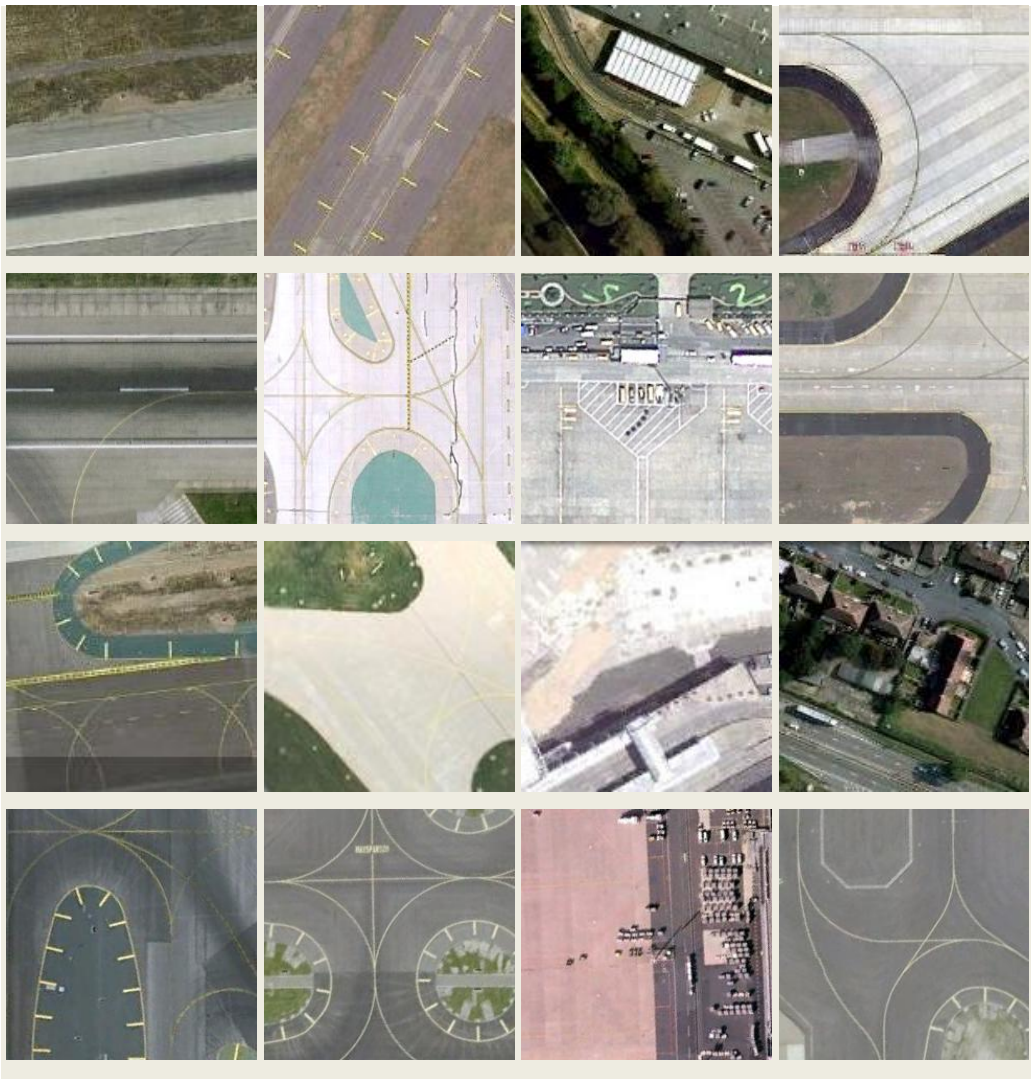


Figure 2.13: Some negative examples generated for training.

2.4.2 The Final Cascaded Structures

For each orientation class of airplane object as defined in the previous section, the maximum acceptable false positive rate per layer in the cascade *fp*, the minimum acceptable detection rate per layer in the cascade *cd* and the target overall false positive rate *FP*, which are described in Figure 2.8, are chosen to be 0.9998, 0.5 and 2.38419 10^{-7}, respectively. The desired rates are achieved with 18 stage classifiers for each rotated airplane class. The training time for a single orientation is approximately 10 hours on a 2.80 GHz Intel Core i7 processor PC with 8 GB RAM. Hence, the total training for all orientations is completed in 10 days.

The final 18 stage cascaded structure includes a total of 220 features for regular eastward-oriented airplanes (see Figure 2.12 (a)). The first six features from the initial stage of the cascade are presented in Figure 2.14. The first four features seem to represent the fact that the horizontal body region of the airplane appears brighter than its background. The features in the later stages could not easily be interpreted visually as the first few features and are highly dependent on the positive and negative examples used for training.

The number of features learned for each orientation class is presented in Table 2.1. The learned features in the first stage of the 270º clockwise rotated airplanes are illustrated in Figure 2.15. One conclusion that can be reached is that the features selected for a particular orientation (α=15º, 30º, 45º,…, 345º) are not necessarily the rotated versions of the features learned for regular eastward-oriented airplanes. One reason for this is that the available tilted features are limited to 45º rotation. The other reason for this fact is the background appearance change for each orientation class resulting from the orientation change of the airplane which changes the occluded parts of the background for each different orientation (see Figure 2.16). Therefore, each learning procedure is considered as a separate case for different oriented airplanes, since the learning procedure is highly dependent on the object and background texture variation.

Table 2.1: Number of features for different orientation classes of airplanes.

Orientation	Total Number of Features Learned	Orientation	Total Number of Features Learned
0º (regular east oriented)	220	**180º**	221
15º	231	**195º**	234
30º	244	**210º**	227
45º	179	**225º**	198
60º	252	**240º**	257
75º	256	**255º**	232
90º	318	**270º**	479
105º	240	**285º**	282
120º	234	**300º**	269
135º	195	**315º**	182
150º	237	**330º**	234
165º	270	**345º**	231

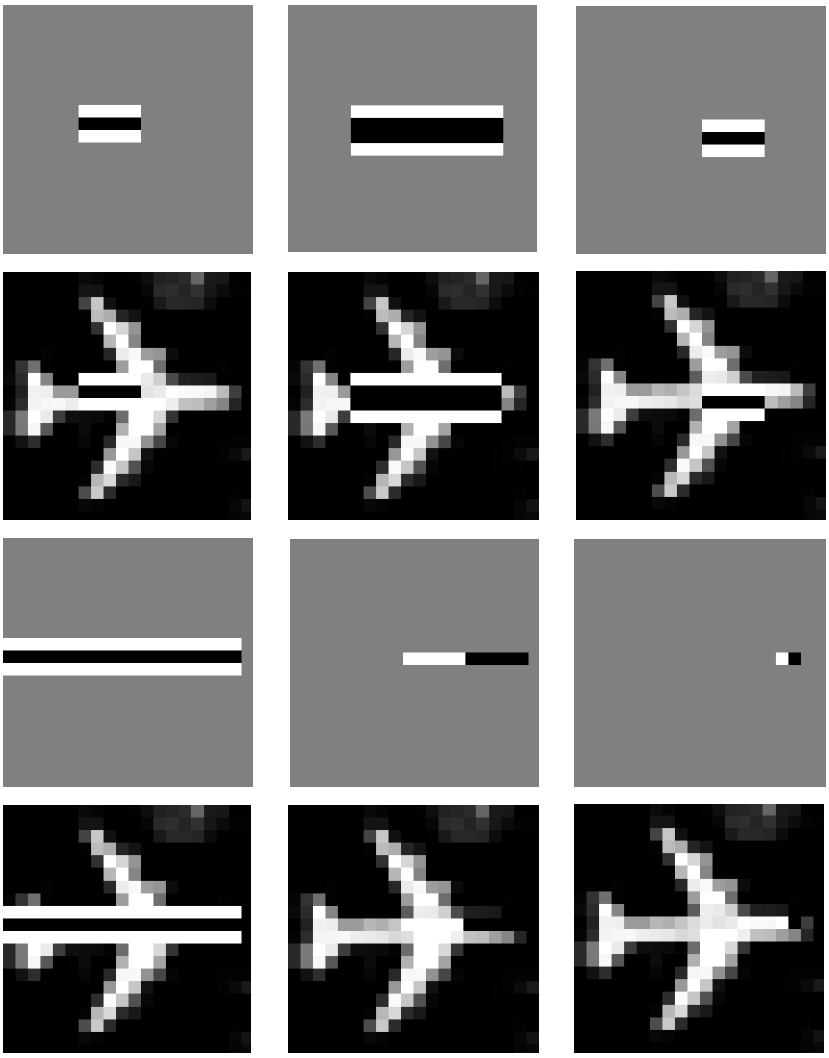


Figure 2.14: First 6 features learned for eastward-oriented airplanes. 1st and 3rd rows represent the features; 2nd and 4th row illustrates the corresponding features on a typical training image.

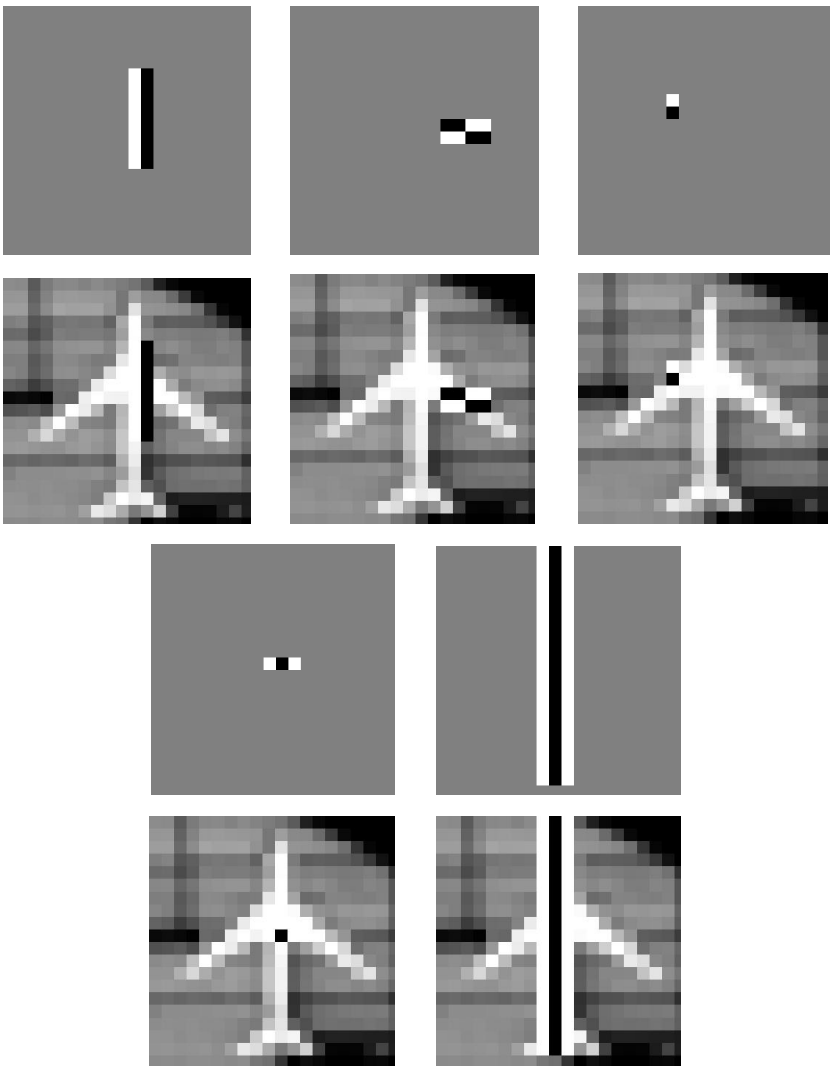
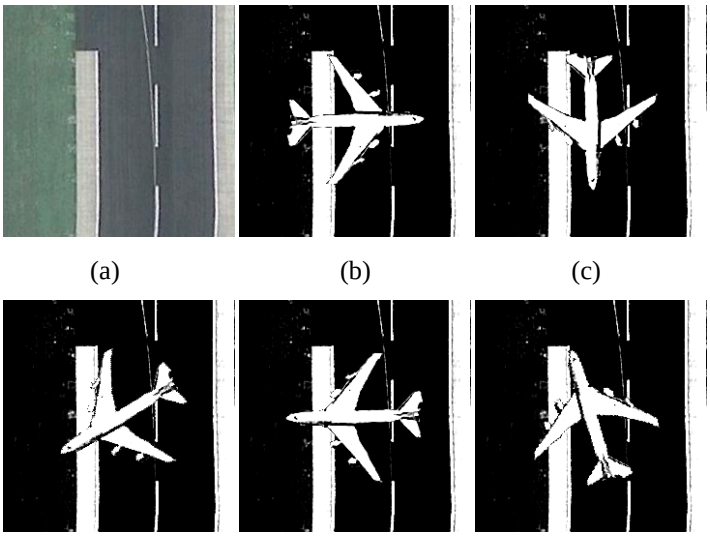


Figure 2.15: First 5 features learned 270º rotated airplanes. 1st and 3rd rows represent the features; 2nd and 4th row illustrates the corresponding features on a typical training image.



(a) (b) (c)

(d) (e) (f)

Figure 2.16: (a) An example background used for synthetic data generation, (b) Regular east oriented variance normalized training example, (c) 90º rotated, (d) 135º rotated, (e) 180º rotated, (f) 255º rotated variance normalized training examples. Lane lines and side lines are occluded differently yielding different appearances to tackle in the learning procedure.

2.4.3 Performance Tests and Discussions

In order to evaluate the performance of our cascaded detectors two set of experiments are conducted. The first experiment is performed on a synthetically generated test set. Toward this end, from the airport regions used to collect training data set, a set of airplanes, which are not used during training, are collected, cropped along the boundary of the object and pasted on 12 different 1920x1032 pixels airport background regions at various orientations. Each synthetically generated test image contains 24 airplanes at various orientations yielding 288 airplanes in total.

The second experiment is performed on a large set of real images containing various airplanes. The data set consists of 20 Google Earth images containing a total of 300 airplanes. For both of these experiments, Google Earth images are adjusted to be 0.5 m resolution by setting the eye altitude to the sum of the elevation terrain and 685 m of distance providing the desired resolution.

For each of the two experiments, 18 stage detectors for all orientations are scanned across the test images at various locations. As mentioned in the previous section, typical airplanes in 0.5 m resolution imagery are sized 150x150 pixels on the average. In order to be independent across small variations in the size the detectors scan the test image with some starting scale of axa pixels, where $a \in \{145, 150\}$. Then, the detectors are scaled by a scale s. Scaling is stopped, when the sub-window reaches 200x200 pixels in the area, since this limit is observed to be the maximum size that the airplanes could occupy. In each scale, the detectors scan the test image over various locations by shifting the detection window by $[\Delta s]$ where Δ is step size and [] represent the rounding operator. The choice of selection of a, s and Δ effect the performance in terms of correct detections and false alarms. For example, larger step sizes and scales tend to decrease the false positives in the expense of decreasing the detection rate.

The final detectors are insensitive to small changes in translation and scale. Furthermore, detectors with consecutive orientations tend to yield multiple detections on an airplane with an orientation close to those associated detectors. In order to have single detection per airplane, overlapping detections must be combined. Toward this end, the detection results are post processed to merge multiple hits. The set of detections, whose centers are fall in a circular region with diameter 60 pixels, are assigned to a single detection with a center and bounding region of the average of the centers and bounding regions of the detections in the set using mean shift clustering algorithm described in [52]. Since smallest possible detection window is selected as 145 pixels, the utilized merging circular region radius is acceptable.

In order to evaluate the performance Recall-versus-Precision curve representation is exploited. The terminology used for the precision and recall rates calculation is defined as follows if the outcome of our binary classifier is labeled either with positive (*p*), i.e. an airplane is detected in a patch by the detector or with negative (*n*), i.e. the patch is rejected by the detector as not containing an airplane.

- *True Positive*: The outcome of the detector for a patch is *p* and the actual value of the patch is also *p*.
- *True Negative*: The outcome of the detector for a patch is *n* and the actual value of the patch is also *n*.
- *False Positive*: The outcome of the detector for a patch is *p* but the actual value of the patch is *n*.
- *False Negative*: The outcome of the detector for a patch is *n* but the actual value of the patch is *p*.

Using the above explanations, precision and recall rates are defined as in the following equations:

$$precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{2.9}$$

$$recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{2.10}$$

In order to create a recall vs. precision curve, the thresholds of the final layers in the cascades structures for different orientation classes should be adjusted from $-\infty$ to $+\infty$. A $+\infty$ threshold yields a recall rate 0 and a false positive rate 0, whereas a $-\infty$ threshold would increase both the recall rate and false positive rate to a certain point, since neither of these rates could be higher than the rate of the detection cascade minus the final layer [12]. Therefore, in order to construct a recall versus precision rate, the layers in the cascaded detectors for each orientation are removed simultaneously by one to obtain the points in the curve.

In order to count the number of true positives, false positives and false negatives binary ground truth masks are prepared. Only one of the multiple detections falling inside the same masks are counted as true positive whereas the others are counted as false positives.

The performance tests are carried under three different cases:

- Case 1: Starting scale of 145x145 pixels, scale factor s = 1.26
- Case 2: Starting scale of 150x150 pixels, scale factor s = 1.35
- Case 3: Starting scale of 150x150 pixels, scale factor s = 1.1

For each single orientation, at least 2 detections are required for a decision of detection of an airplane, i.e. single detections are rejected by the detector in order

to decrease the false positives. Moreover, for each of the three cases a step size, Δ of 1 is used.

The final detector can perform detection task for a 1920x1032 pixels image within 1.5 seconds on a 2.80 GHz Intel Core i7 processor PC with 8 GB RAM.

Figure 2.17 illustrates the recall vs. precision curve comparisons for three cases explained earlier for the experiments carried on the synthetically generated test set. The best performance of the algorithm with precision and recall rates, 86% and 92%, respectively, is obtained for Case 1 by the cascaded detectors containing 18 stages. The corresponding sample detection results are shown in Figure 2.18 where the correct detections, misses and false positives are represented with green, red and blue circles, respectively.

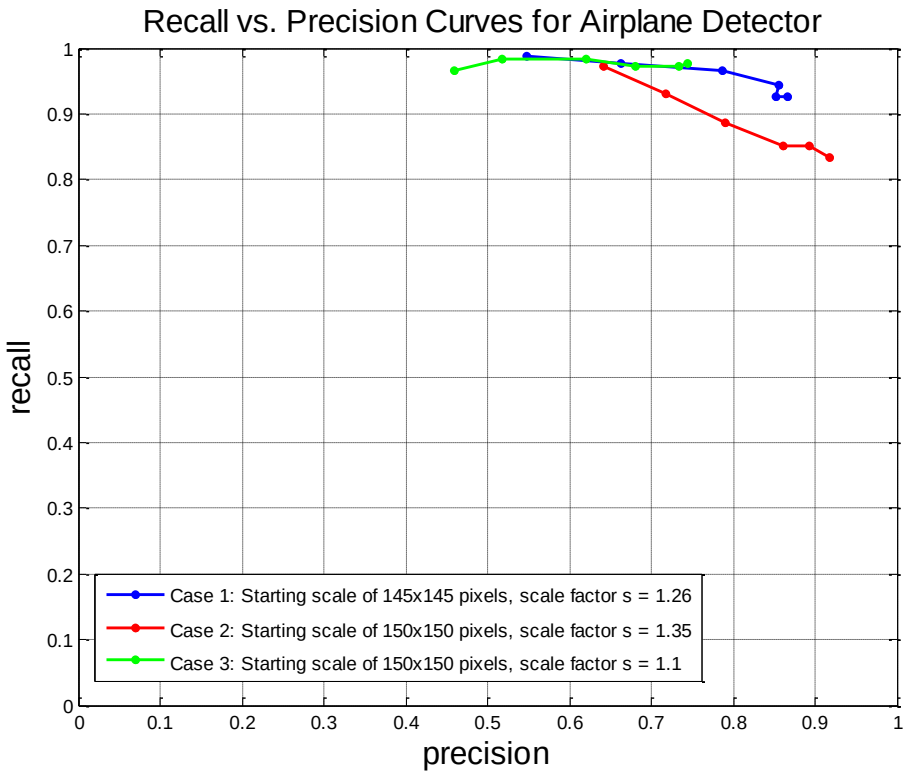


Figure 2.17: Precision vs. Recall curves for our airplane detector on the synthetically generated data set.

(a)

(b)

Figure 2.18: (a) and (b) Examples of typical detection results on synthetically generated test set for Case 1.

Figure 2.19 illustrates the recall vs. precision curve comparisons for three cases for the experiments carried on a real data set. The best performance of the algorithm, for which there is competence in both precision and recall rates, 64% and 81%, respectively, is obtained for Case 1 with a cascaded detectors containing 17 stages. Typical detection results at the corresponding best point are presented in Figures 2.20, 2.21, 2.22 and 2.23; in which correct detections, misses and false positives are represented with green, red and blue circles, respectively.

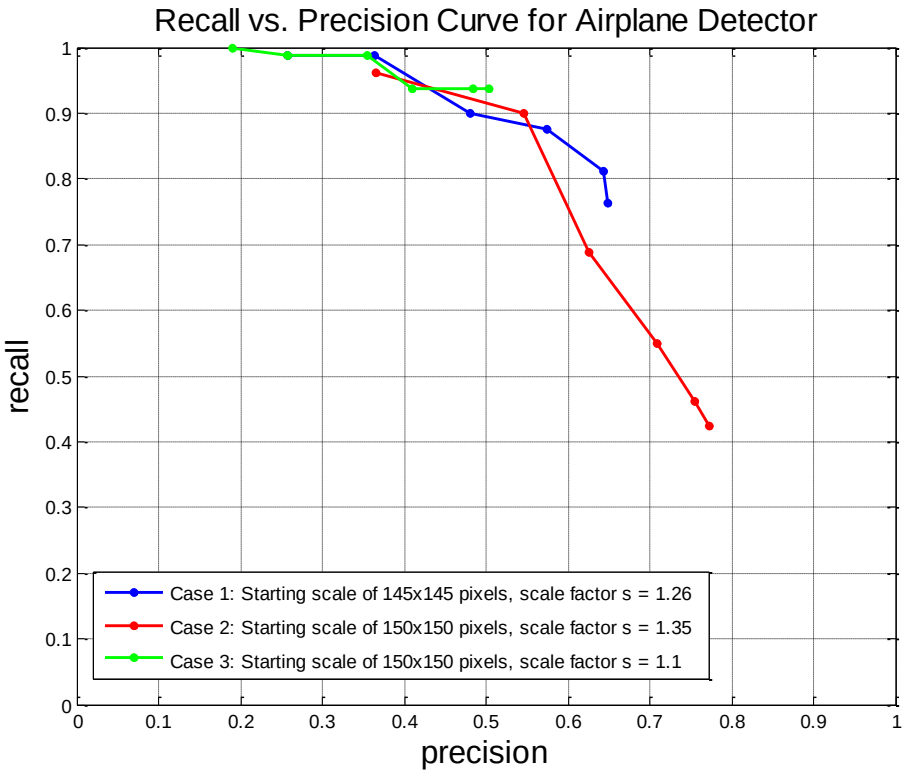


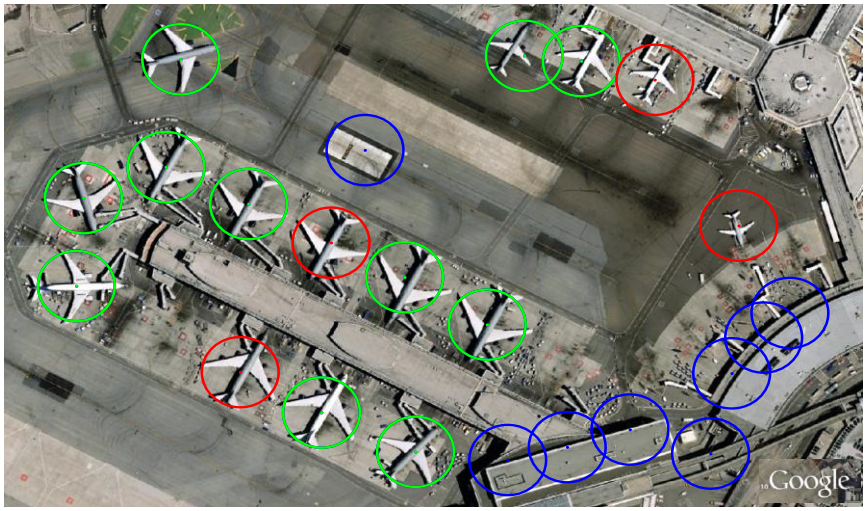
Figure 2.19: Precision vs. recall curves for our airplane detector on the real dataset.

The curves illustrated in Figures 2.17 and 2.19 deviate from an expected ideal non-increasing, convex behavior at some points. This deviation is resulted from the merging procedure. Although the points on the Recall vs. Precision curve associated to the higher number of classifiers are expected to yield higher

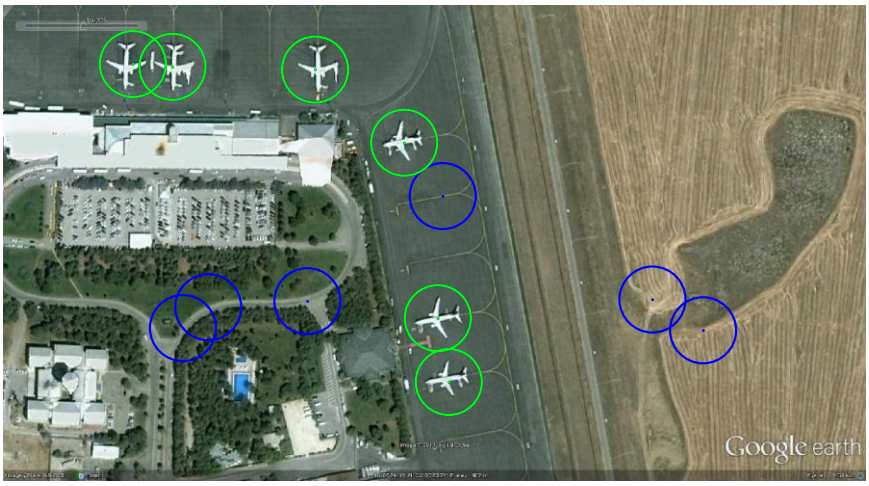
precision and lower recall rates, the individual detections prior to the merging procedure become sparser and multiple detections around local neighborhoods decrease as more classifiers involved in the detector. Consequently, single detections are expected to remain without being merged to other points and this could possibly decrease the precision rate unlike expected.

For each of these two experiments, Case 1 is observed to yield higher performance over Case 2 and Case 3. By a scale factor 1.26, 145x145 and 182x182 pixel sub windows are scanned through the test images in Case1. Using a scale factor of 1.35, effectively, only 150x150 pixel sub window is scanned through the test images for Case2, since 200x200 pixels is the maximum sub window size. Therefore, the curve belonging to Case 2 yields higher precision rates at the expense of lower recall rates as expected, since as the total number of patches scanned in the test image decreases, the false alarms decrease. Having a scale factor of 1.1; 150x150, 165x165, 181x181 and 199x199 pixel sub windows are effectively utilized during the scanning process for Case 3. While having the highest recall rates compared to Case1 and Case2, Case3 could not reach high precision rates.

As it can be interpreted from the test results presented in Figures 2.17 and 2.19, the cascaded detectors trained on the synthetically generated data are observed to have a degraded performance on the real dataset. This result is primarily due to the fact that the statistical training methods require a sufficiently large, representative data and yet, considering the intra class variations of airplanes, illumination changes depending on the time and season of the images are captured and background clutter it is quite difficult to account for all possible appearances for each orientation with synthetically generated data. Moreover, collecting a real annotated training set that is adequately representative in terms of the conditions explained above is not feasible for our case of airplane object class training.



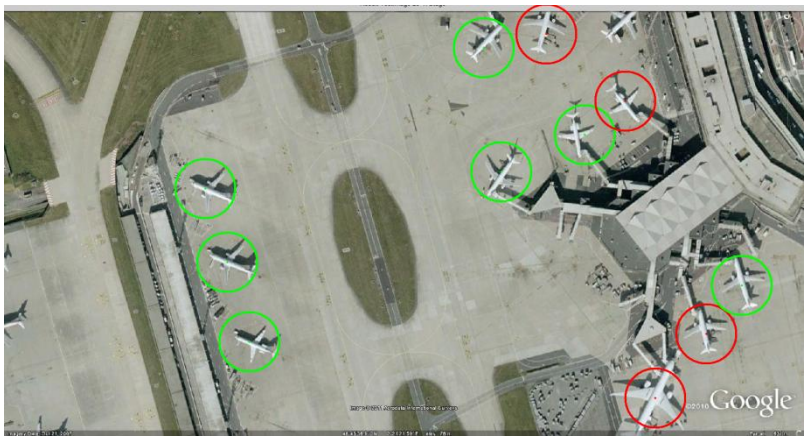
(a)



(b)

(c)

Figure 2.20: (a), (b) and (c) Examples of typical detection results on real test set for Case 1.

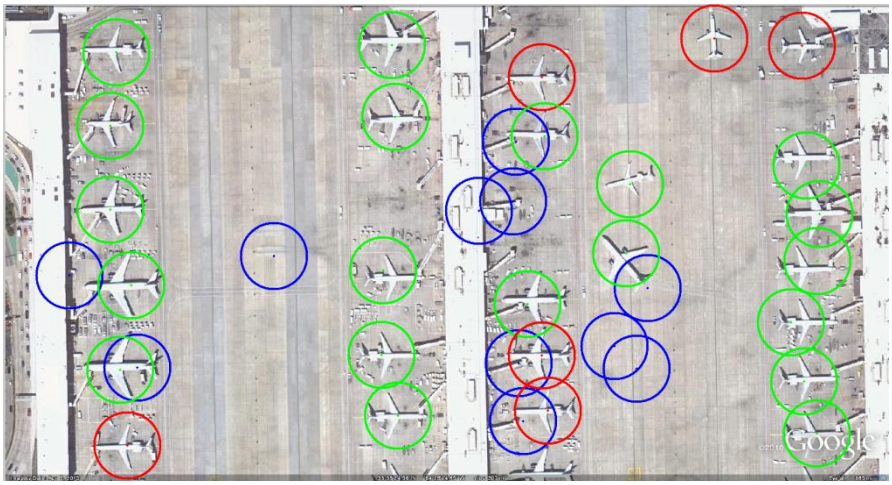


(a)

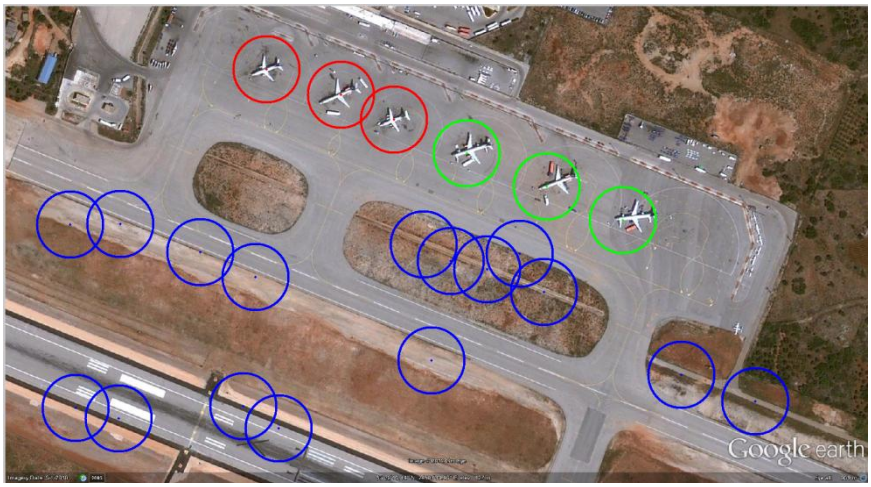
(b)

(c)

Figure 2.21: (a), (b) and (c) Examples of typical detection results on real test set for Case 1.

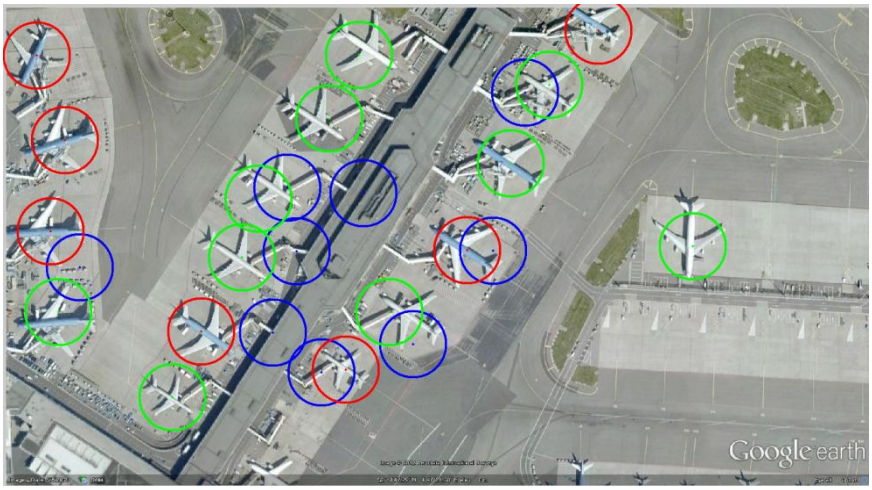


(a)

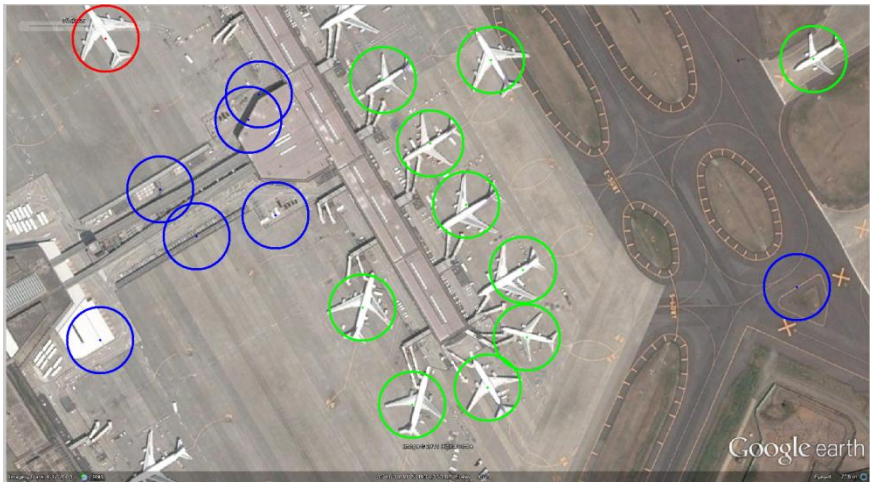


(b)

Figure 2.22: (a) and (b) Examples of typical detection results on real test set for Case 1.



(a)



(b)

Figure 2.23: (a) and (b) Examples of typical detection results on real test set for Case 1.

CHAPTER 3

PROPOSED METHOD

In this chapter, a novel method for geospatial object detection which does not require large training sets or initial segmentation for reaching a satisfactory performance is proposed. Inspired by the rectangle features, an operator exploiting the edge information via gray level differences between the geospatial object and its background is constructed with Haar-like polygon regions by using the shape information of the object as an invariant. The proposed method is evaluated for airplane class.

This chapter is organized as follows: A review of related work in matched filtering concept for object detection is given in Section 3.1. In Section 3.2, the global appearance based airplane operator is presented. In Section 3.3, the proposed detection algorithm for the airplane operator is explained. Finally, performance tests and results are presented in Section 3.4.

3.1 Matched Filtering

Spatial filtering can be utilized in object detection framework. A matched filter, in the context of image processing, is a spatial filter that can be used for providing a similarity measure between a test image and a reference image to be used to determine the presence of a known object. This similarity measure is simply the correlation between the test image and reference image. Assuming that a test

image, $i(x,y)$, is composed of the target image, $t(x,y)$, and an additive stationary noise, $N(x,y)$; the spatially filtered output by a matched filter with impulse response, $h(x,y)$ is given by:

$$i_o(x,y) = i(x,y) * h(x,y) \tag{3.1}$$

where $i_o(x, y)$ is the output image and input image is addition of two terms, as

$$i(x,y) = t(x,y) + N(x,y) \tag{3.2}$$

The matched filter should be designed such that the signal to noise energy ratio (SNR) at the filter output is maximized at a point (α, β).The signal energy at point (α, β) when no additive noise included can be represented by [49]:

$$|S(\alpha,\beta)\,|^2 = |t(x,y) * h(x,y)|^2 \tag{3.3}$$

The total noise energy at the matched filter output is [49]:

$$N = \iint_{-\infty}^{+\infty} W_N(\omega_x,\omega_y)\left|H(\omega_x,\omega_y)\right|^2 \mathrm{d}\omega_x d\omega_y \tag{3.4}$$

where $W_N(\omega_x,\omega_y)$ and $H(\omega_x,\omega_y)$ represent the power spectral density of stationary, image independent noise $N(x, y)$ and the transfer function of $h(x, y)$, respectively. Using the above definitions together with convolution theorem, signal-to-noise ratio, SNR is obtained as follows [49]:

$$SNR = \frac{\left|\iint_{-\infty}^{+\infty} T(\omega_x,\omega_y)\,H(\omega_x,\omega_y)\mathrm{e}^{\left\{\mathrm{i}(\omega_x\alpha,+\omega_y\beta)\right\}}\mathrm{d}\omega_x d\omega_y\right|^2}{\iint_{-\infty}^{+\infty} W_N(\omega_x,\omega_y)|H(\omega_x,\omega_y)|^2\,\mathrm{d}\omega_x d\omega_y} \tag{3.5}$$

where $T(\omega_x, \omega_y)$ is the Fourier transform of $t(x,y)$.

The optimal matched filter that maximizes the SNR given in (3.5) is obtained by a transfer function as in the following relation [49]:

$$H(\omega_x, \omega_y) = \frac{T^*(\omega_x,\omega_y)\mathrm{e}^{\{-\mathrm{i}(\omega_x\alpha,+\omega_y\beta)\}}}{W_N(\omega_x,\omega_y)} \tag{3.6}$$

If the power spectral density of the input noise is assumed to be uniform such that $W_N(\omega_x, \omega_y) = n$, the transfer function of the optimal matched filter becomes:

$$H(\omega_x, \omega_y) = \frac{T^*(\omega_x,\omega_y)\mathrm{e}^{\{-\mathrm{i}(\omega_x\alpha,+\omega_y\beta)\}}}{n} \tag{3.7}$$

The impulse response corresponding to the optimal matched filter is obtained as:

$$h(x, y) = \frac{t^*(\alpha-x,\beta-y)}{n} \tag{3.8}$$

The matched filter is the 180° rotated complex conjugate of the scaled version of the target image.

If the target image is real, then the matched filter output in case of an additive noise becomes:

$$i_o(x, y) = \frac{i(x,\ y) * t(\alpha-x,\beta-y)}{n} \tag{3.9}$$

The filter output is proportional to the correlation between the test image and the target image and the correlation peak is obtained when the matched filter totally overlaps the region of the test image containing the target image possibly together with additive noise [49].

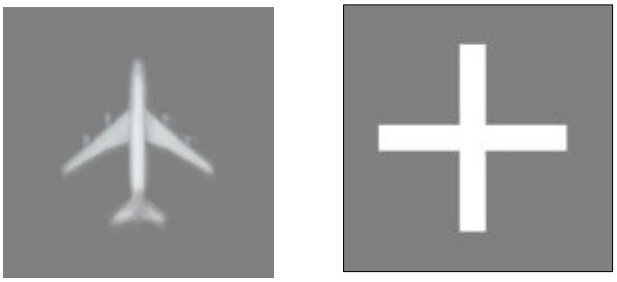
The matched filtering concept implies that the optimal spatial filter that correlates the target image to test image is the target image itself; i.e. the template with a scaling factor.

For the case of airplane detection, since there is not only a single airplane model available, matched filtering with a single airplane template would not yield the desired correlation outputs, considering the intra class variations of the airplanes and the appearance changes resulting from illumination changes and shadowing effects. An enormous amount of templates should be tested, which increases the computational cost, as well as false positives at the output dramatically. A rotation, scale and translation invariant target recognition from satellite imagery framework based on template matching is presented in [53]. However, the method, which employs normalized cross-correlation and phase correlation as similarity measures, suffers from high computational complexity. Due to these limitations, template matching is usually utilized by shape invariant objects, including edges joined in some special arrangements [49].

In this thesis, the idea of template matching exploited to construct an operator that can operate fast which is described in the following section.

3.2 Airplane Operator

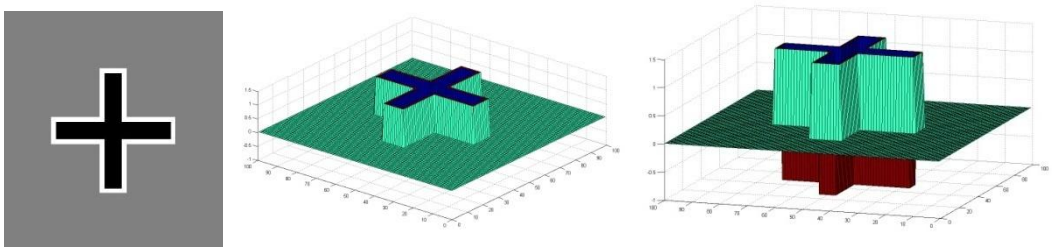
When observed from satellite imagery with a predefined resolution, airplanes appear to have a special polygonal shape (see Figure 3.1). In this thesis, that special appearance of airplane is modeled with a plus sign shaped Haar-like polygonal operator similar to [50] for fast detection purpose.



(a) (b)

Figure 3.1: (a) Average of 50 airplanes which are cropped and put on uniform background (b) Airplane approximation.

2D and 3D illustrations of the airplane operator are presented in Figure 3.2. Since it is hard to account for all possible colors and shape variations, a plus-shape (i.e. "+") is used in gray level to construct this airplane operator. Therefore, the objective is to utilize the intensity difference between the airplane and its surrounding background in gray level, assuming that airplanes create sufficient contrast with the background along their edge boundaries. The edge response is collected with the airplane operator similar to [33]. Instead of a point-by-point correlation, which is required for a conventional template matching method, integral image representation is utilized for fast evaluation of the airplane operator.



(a) (b)

Figure 3.2: (a) 2D and (b) 3D illustrations of aircraft operator.

If the black region and white region shown in Figure 3.1 (a) are denoted with R_b and R_w with their corresponding areas represented by a_b and a_w, respectively; the airplane operator $f_a(x,y)$, can be formulated mathematically as follows:

$$f_a(x,y) = \begin{cases} -1 & for \quad x,y \in R_b \\ \dfrac{a_b}{a_w} & for \quad x,y \in R_w \end{cases} \tag{3.1}$$

The response image, $R(x,y)$ to the airplane operator is calculated by convolving the gray level input image $I(x,y)$ with the airplane operator followed by a magnitude operator in order to be invariant against different intensity transitions across the edge, such as brighter background with relatively darker object or darker background with relatively brighter object.

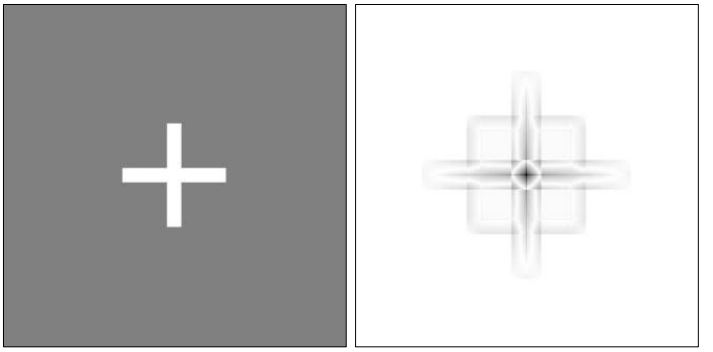
$$R(x,y) = \left| I(x,y) * f_a(x,y) \right| \tag{3.2}$$

Using (3.1), (3.2) can be equivalently represented as follows:

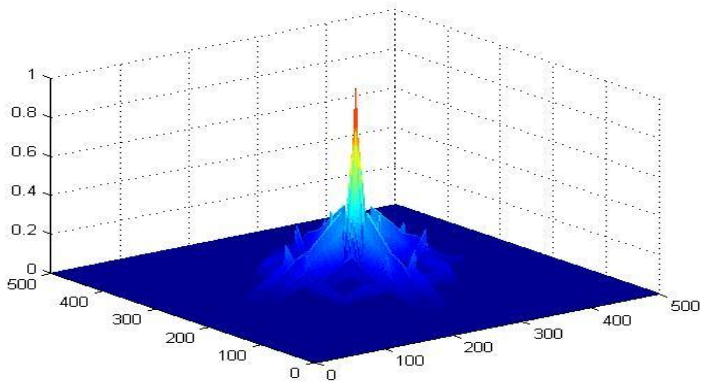
$$R(x,y) = \left| \left(\frac{a_b}{a_w} \right) \sum_{x',y' \in R_w} I(x',y') - \sum_{x',y' \in R_b} I(x',y') \right| \tag{3.3}$$

The response is simply the correlation of the input image with the operator. In the formulation of the airplane operator (see equation (3.1)), the coefficients -1 and $\frac{ab}{aw}$ are selected such that the operator would yield a zero response on a region with uniform pixel values, whereas it would yield a peak response when it is at an airplane center. In order to illustrate the response image for the ideal circumstances without noise, shadowing effects or textured background, some synthetic images are constructed. The first synthetic image shown in Figure 3.3 (a) is a plus sign shape on a uniform background that exactly matches airplane operator in shape, i.e. the operator is an exact template of the synthetic image.

Hence, the corresponding response (see Figures 3.3 (b) and (c)) is obtained with a peaky behavior around the center of the image as expected.



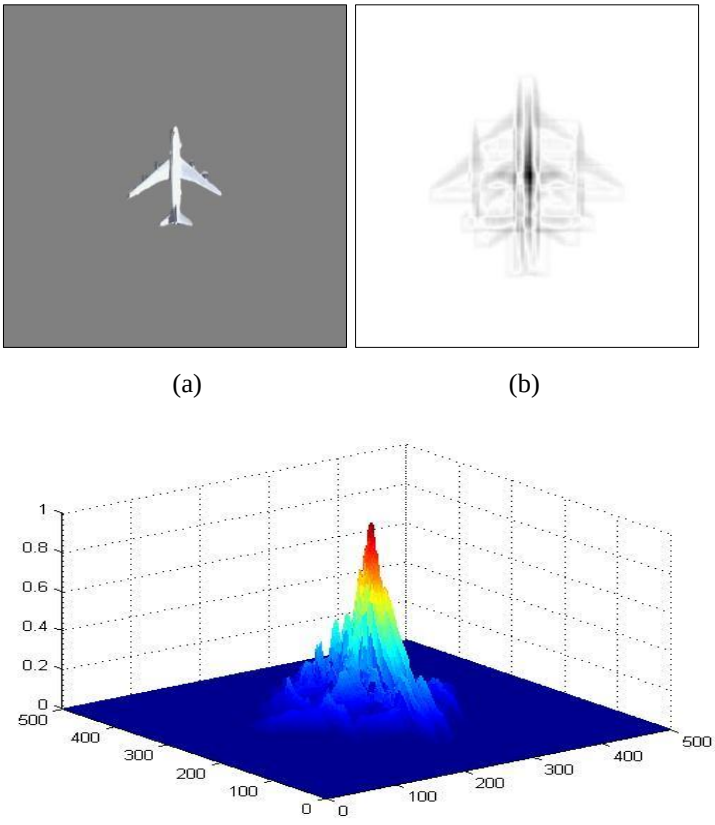
(a) (b)



(c)

Figure 3.3: (a) An ideal plus sign shaped object matching the airplane operator, (b) the corresponding response image and (c) 3D visualization of the response image.

The second simulated image is constructed by putting a real cropped airplane image obtained from Google Earth onto a uniform background as shown in Figure 3.4 (a). Since the airplane operator is not an exact match to the simulated image in this case, the response deviates from the ideal case with an exact template matching. But the peaky behavior around the center of the airplane in the response image (see Figures 3.4 (b) and (c)) indicates that the operator still yields a peak correlation, when the operator is on top of an airplane, similar to optimal matched filtering.



(a) (b)

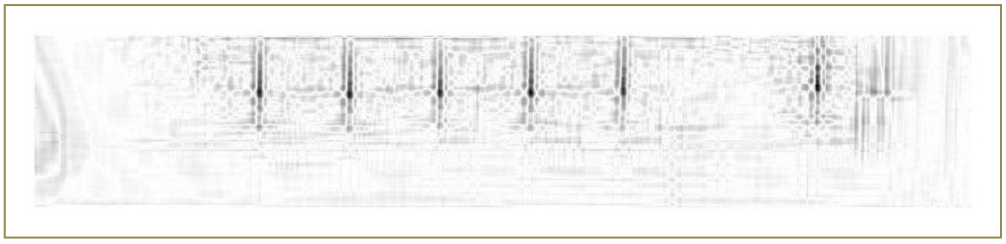
(c)

Figure 3.4: (a) A real cropped airplane image on a uniform background, (b) the corresponding response image and (c) 3D visualization of the response image.

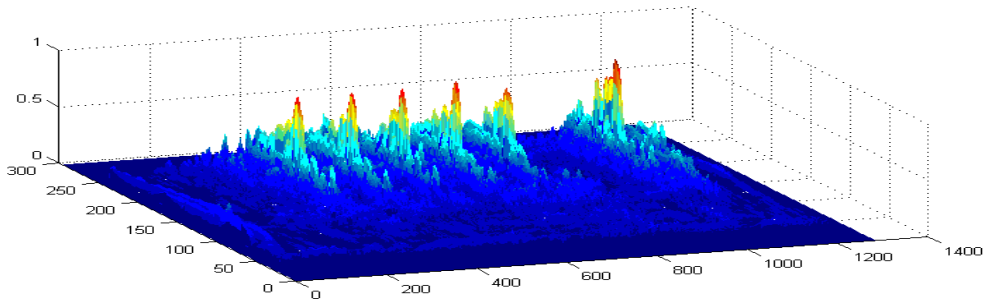
3.3 Airplane Detection Algorithm

As explained in the previous section, although being suboptimal, the airplane operator response has still a peaky behavior around the center of an airplane that matches the airplane operator. Therefore, after being compared to a threshold, this peaky response around the center of an airplane could be utilized for the detection purpose. To this end, given a test image, the airplane operator is applied at each pixel of the image and corresponding responses are obtained and stored. A typical response image to the airplane operator for an image containing upright airplanes taken from Google Earth is presented in Figure 3.5.

(a)



(b)



(c)

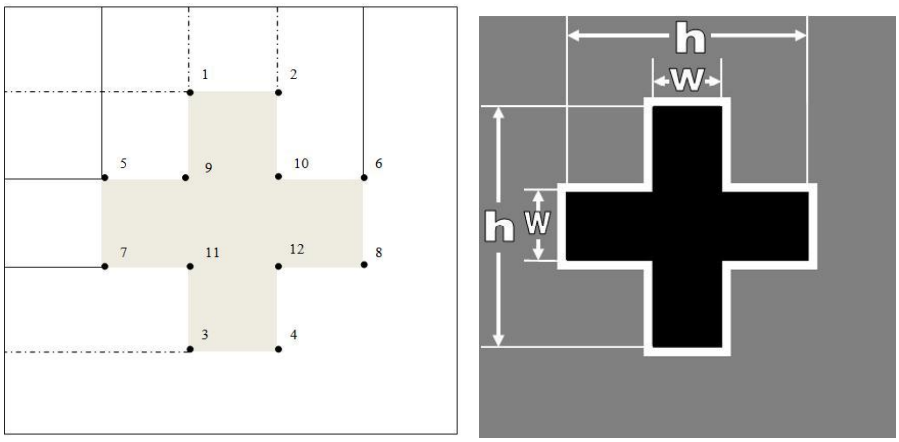
Figure 3.5: (a) An airport image, (b) corresponding response image and (c) 3D visualization of the response image.

In this thesis, detection is aimed to be performed on 0.5 m resolution images. As aforementioned, typical passenger planes occupy 150x150 pixels on the average. In order to be invariant against small variations in the size, two different scales of airplane operator are utilized in this thesis.

The complexity of a traditional cross correlation based template matching approach is directly proportional to the size of the operator and the input image. For an input image $I(x, y)$ of size $R \times C$ pixels and an airplane operator with size S, the complexity of calculating the response image for a single orientation of airplane operator using cross correlation at each pixel is $O(RCS)$. Considering the

average airplane size on 0.5 m resolution satellite images, the airplane operator size reaches to an area of 5600 pixels, which increases the computational complexity excessively, considering the possible different orientations this operator should be evaluated at. Therefore, using the properties of integral image as described in Section 2.1.2, the response to airplane operator is calculated with only a few array operations in constant time at any location of the test image as illustrated in Figure 3.5. The complexity using the integral image representation reduces to $O(RC)$.

The airplane operator is applied at different input image orientations for rotation invariant detection. Since the airplane operator is symmetric with respect to the origin as shown in Figure 3.6 (b), five angles that are 15º apart (15, 30, 45, 60, 75º) are selected from the quarter range of 0-360º. After evaluating the airplane operator at each orientation of the input image by using two scales, the responses are collected and stored. For each orientation of the input image, the points in the response image that are above a certain threshold, while being a local maximum around their surrounding neighborhood defined by the size of the operator, are collected as candidate airplane centroids. Those centroids in the rotated input image are then mapped back to the corresponding points in the regular input image. Since the threshold selection directly affects the detection performance, eight samples of the threshold, t, ($t/1.4$, $t/1.7$, $t/2$, $t/2.2$, $t/2.5$ $t/3.2$, $t/4$, $t/5$) initially obtained by using Otsu's thresholding method [51], are employed to construct the performance curve. Due to the high resolution angle sampling of 15º, multiple detections are obtained around a single airplane center. Therefore, a merging procedure by using mean shift clustering algorithm [52] is applied to the candidate airport centroids that are forming sets with a maximum distance of 60 pixels apart to achieve final detection. The pseudo algorithm for airplane detection utilizing the airplane operator is given in Figure 3.7.



(a) (b)

Figure 3.6: (a) The sum of pixel values under gray colored cross area is computed in 12 array references: [1+4 – (2+3)] + [5+8 – (6+7)] - [9+12 – (10+11)] the value of integral images at the black dots are denoted with 1,2,…12 (The value of integral image at a point is calculated by summing the pixels above and to the left of that point as described in Section 2.1.2). (b) Aircraft operator sizes.

- For the five different orientations, α, (0, 15, 30, 45, 60, 75º) of the input image
    1. Rotate the input image by α.
    2. Compute the integral image.
    3. Using integral image collect the airplane operator responses.
        - Threshold the response image by *T*.
        - Find candidate points in the threshold response that is a local maximum.
        - Map back and store the candidate points to the regular input image for each orientation.
        - Cluster the points that are maximum 60 pixels apart from each other.
- Repeat step 3 for each of thresholds *T = (t/1.4, t/1.7, t/2, t/2.2, t/2.5 t/3.2, t/4, t/5)* to obtain the points on performance curve. (*t* is the threshold found using Otsu's method.)

Figure 3.7: The airplane detection algorithm utilizing airplane operator.

3.4 Performance Tests

The experiments are performed on the same real data set of 20 Google Earth images described in Section 2.4.3. The recall vs. precision curve for the airplane operator is shown in Figure 3.8. The best performance of the algorithm with precision and recall rates of 68% and 88%, respectively, is obtained for the threshold value *t/2.2*. Sample detection results for the corresponding point are presented in Figures 3.9, 3.10, 3.11 and 3.12. As described earlier; correct detections, misses and false positives are represented with green, red and blue circles, respectively.

The airplane operator can perform detection task for a 1920x1032 pixels image within 1.7 seconds on a 2.80 GHz Intel Core i7 processor PC with 8 GB RAM.

It can be observed form the test results that the proposed airplane operator based detection algorithm provides slightly better performance, compared to the Haar feature based cascaded detector system. Since the airplane operator is not dependent on a training set, it is more generic compared to the trained set of Haar-like features. This property of the airplane operator makes the detection procedure more robust to illumination changes and shadows. For most of the cases, cast shadowing even help increase the detection performance by creating more contrast between the object and the background. Further analysis of the performance tests and future work will be discussed in the final chapter.

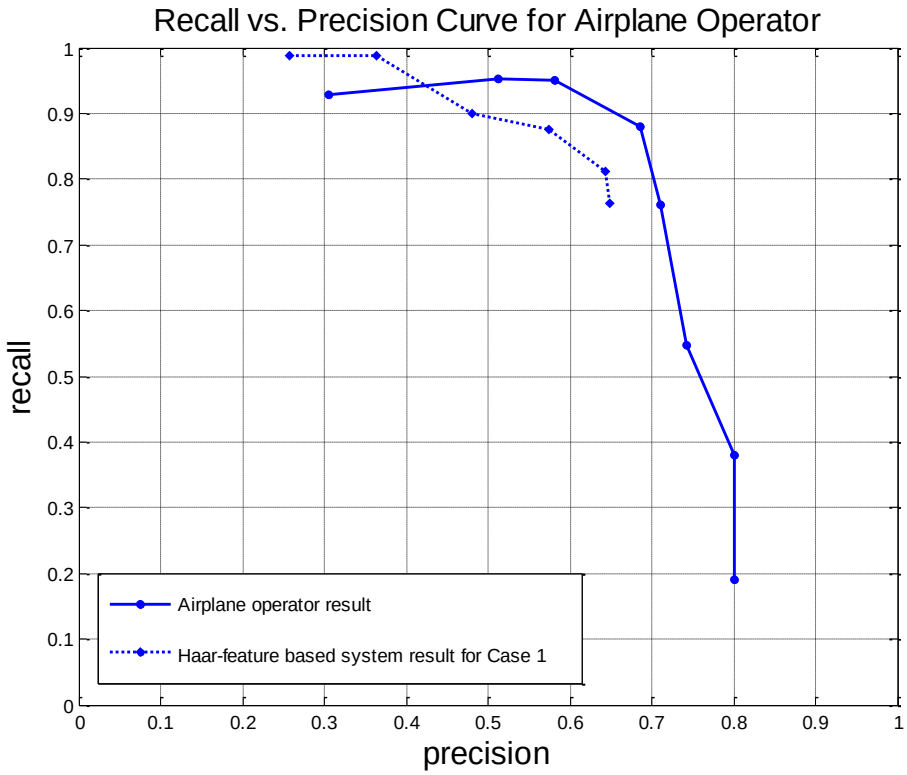
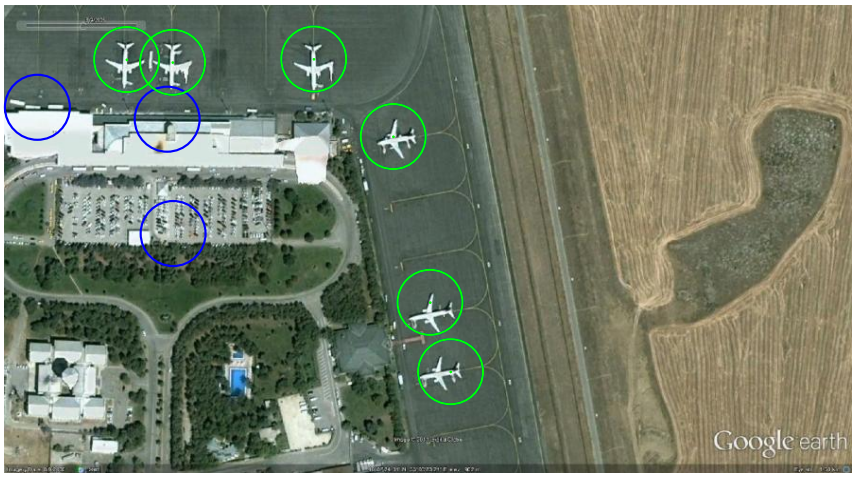


Figure 3.8: Precision vs. recall curve for our airplane operator on the real data set.

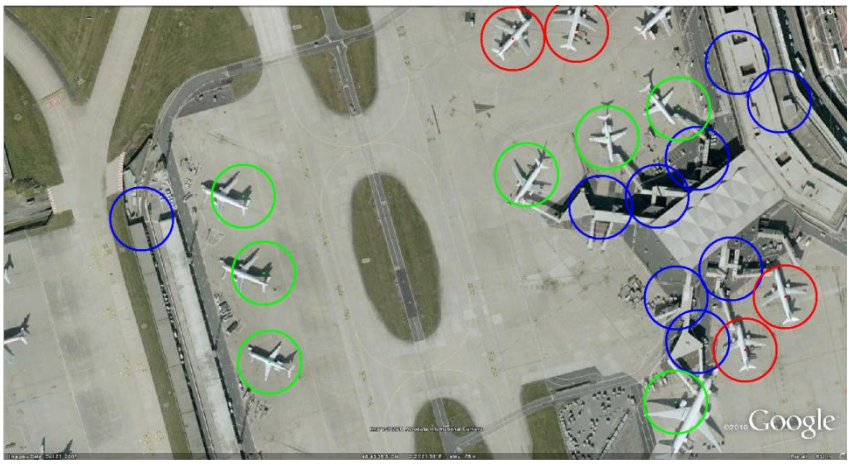
(a)



(b)

(c)

Figure 3.9: (a), (b) and (c) Examples of typical detection results on real test set.

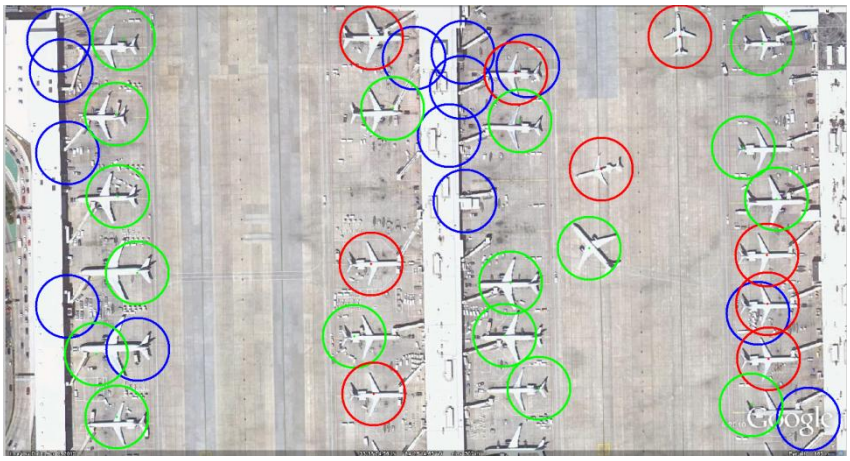


(a)

(b)

(c)

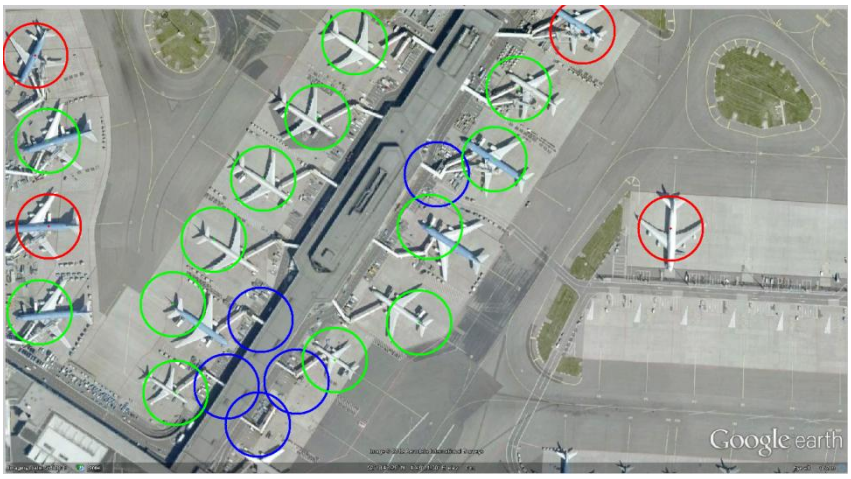
Figure 3.10: (a), (b) and (c) Examples of typical detection results on real test set.



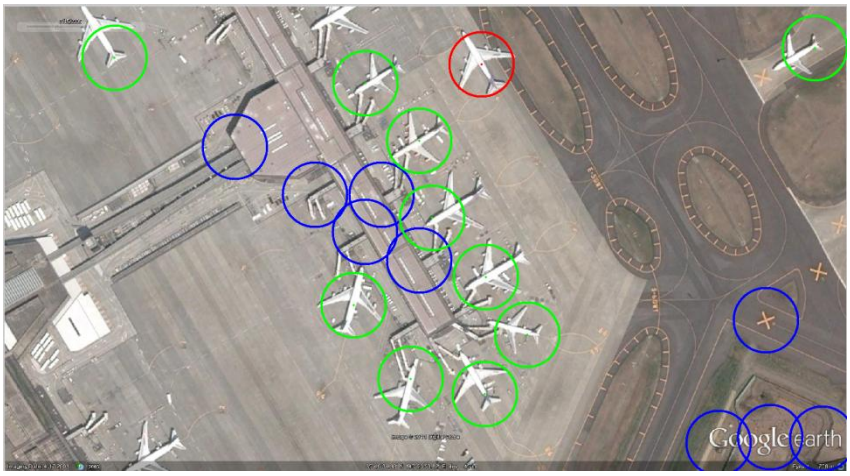
(a)

(b)

Figure 3.11: (a) and (b) Examples of typical detection results on real test set.



(a)



(b)

Figure 3.12: (a) and (b) Examples of typical detection results on real test set.

CHAPTER 4

CONCLUSIONS

4.1 Summary of the Thesis

This thesis is devoted to the problem of airplane detection from satellite imagery exploiting global appearance characteristics. The proposed method employs the computational power of integral image representation to evaluate a generic operator that is able to detect and localize objects.

Haar-like rectangle feature based system introduced in [12], is initially proposed for single view face detection problem. In this thesis, this approach is extended to multi-view (in-plane rotated) airplane detection problem. Since collecting a real data set with huge amounts of data that is required for such a statistical training method is inconvenient, a synthetically generated set is utilized for training. Extensive set of experiments using this set are performed under a wide range of operating environments with synthetic and real data sets.

The proposed algorithm utilizing invariant global appearance property is tested on the same real data set with the Haar-like rectangle feature based system in order to compare the performances. Recall versus precision curve representations are exploited for performance evaluation.

4.2 Conclusions and Future Work

The experiments verified that the Haar-like rectangle feature based system trained on synthetically generated data set is able to distinguish the airplanes from background on the synthetic test set quite efficiently. However, tests on a real case scenario resulted with a decrease in the performance as expected. This is a natural outcome of the disturbances implied by intra-class variations, cast and self shadowing effects and background clutter. In order to be adequately representative, various backgrounds are used in generation of synthetic data set. However, the explicit shape and appearance characteristic of the airplane are always not preserved under these conditions. Examples of different appearances in the training set are shown in Figure 4.1.

The proposed method that utilizes the explicit global appearance characteristic of airplanes, benefits from invariance. This framework could also be generalized for detecting other objects with rectangular shapes or targets that could be represented by unions of rectangular sub blocks.

Although the proposed method has promising results performing better than the Haar-like rectangle feature based system, there are still some problems. The additional challenges arise from the man-made objects such as buildings, vehicles, lane lines, etc. The man-made objects having sharp edges and corners, give high responses to the operator yielding false alarms.

A pre-classification step could be employed for classifying the man-made objects, vegetation, etc., to focus attention of the operator on region of interests that are more likely to contain airplanes, as a future work. Furthermore, a more sophisticated model for airplanes could contribute to the performance to eliminate false alarms.

Figure 4.1: Examples of various airplane appearances from synthetic training set.

REFERENCES

[1] P. M Roth and M. Winter, "Survey of Appearance-based Methods for Object Recognition" Technical Report, Inst. for Computer Graphics and Vision Graz University of Technology, Austria, 2008.

[2] D. G. Lowe. "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*, vol. 60, No. 2, pp. 91-110, 2004.

[3] Y. Freund and R. E. Shapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting", *Journal of Computer and System Sciences*, vol. 55, No. 1, pp. 119-139, 1997.

[4] R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification*, John Wiley & Sons, 2000.

[5] C. Tao, Y. Tan, H. Cai and J. Tian, "Airport Detection From Large IKONOS Images Using Clustered SIFT Keypoints and Region Information", *Geoscience and Remote Sensing Letters, IEEE*. Vol. 8 pp. 23-27, 2011.

[6] S. Sahli, Y. Ouyang, Y. Sheng and D. A. Lavigne, "Robust Vehicle Detection in Low-Resolution Aerial Imagery", *Proceedings of SPIE*, 2010.

[7] X. Sun, H. Wang and K. Fu, "Automatic Detection of Geospatial Objects Using Taxonomic Semantics", *Geoscience and Remote Sensing Letters, IEEE*. Vol. 7, pp. 23-27, 2010.

[8] M. Fischler and R. Elschlager, "The Representation and Matching of Pictorial Structures", *IEEE Trans. Computers*, Vol. 22, No. 1, pp. 67-92, 1973.

[9] M. Turk and A. Pentland, "Face Recognition Using Eigenfaces", *Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 586-591, 1991.

[10] H. A. Rowley , S. Baluja and T. Kanade, "Neural Network-Based Face Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.20, No. 1, pp. 23-38, 1998.

[11] F. Fleuret and D. Geman, "Graded Learning for Object Detection", *Proceedings of IEEE Workshop Statistical and Computational Theories in Vision*, 1999.

[12] P. Viola and M. J. Jones, "Robust Real–Time Face Detection", *International Journal of Computer Vision*, vol. 57, No. 2, pp. 137–154, 2004.

[13] P. Viola and M. J. Jones, "Robust Real-Time Object Detection", *Proceedings of IEEE Workshop on Statistical and Computational Theories of Vision*, 2001.

[14] B. Heisele, T. Serre, S. Mukherjee and T. Poggio, "Feature Reduction and Hierarchy of Classifiers for Fast Object Detection in Video Images", *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 18–24, 2001.

[15] A. Blum and P. Langley, "Selection of Relevant Features and Examples in Machine Learning", Artificial *Intelligence*, vol. 10, pp. 245–271, 1997.

[16] R. Kohavi, "Wrappers for Feature Subset Selection", *ArtificialIntelligence*, vol. 97, pp. 273–324, 1995.

[17] C. Papageorgiou and T. Poggio, "A Trainable System for Object Detection", *International Journal of Computer Vision*, vol. 38, No. 1, pp. 15–33, 2000.

[18] X. Perrotton, M. Sturzel and M. Roux, "Automatic Object Detection On Aerial Images Using Local Descriptors And Image Synthesis", *In Computer Vision Systems, Lecture Notes in Computer Science*, pp. 302-311, 2008.

[19] D. Gavrila and V. Philomin, "Real-Time Object Detection For "Smart" Vehicles", *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 1, pp. 87-93, 1999.

[20] A. Opelt, A. Pinz and A. Zisserman, "A Boundary-Fragment-Model For Object Detection", *Proceedings of European Conference on Computer Vision*, vol.2, pp. 575–588, 2006.

[21] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection”, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 886–893, 2005.

[22] R. Polikar, “Ensemble Based Systems in Decision Making”, *IEEE Circuits and Systems Magazine*, vol. 6, No. 3, pp. 21-45, 2006.

[23] Y. Freund and R.E. Schapire, “Decision-Theoretic Generalization Of On-Line Learning And An Application To Boosting,” *Journal of Computer and System Sciences*, vol. 55, No. 1, pp. 119–139, 1997.

[24] A. Torralba, K. P. Murphy and W. T. Freeman, “Sharing Visual Features for Multiclass and Multiview Object Detection”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, No. 5, pp. 854-869, 2007.

[25] T. Serre, L. Wolf and T. Poggio, “Object Recognition with Features Inspired by Visual Cortex”, *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp.994-1000, 2005.

[26] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, “Gradient-Based Learning Applied to Document Recognition”, *Proceedings of the IEEE*, vol. 86, No. 11, pp. 2278–2324, 1998.

[27] K. Fukushima, “Neocognitron: A Self-Organizing Neural Network Model for A Mechanism of Pattern Recognition Unaffected by Shift in Position”, *Biological Cybernetics*, 1980.

[28] B. Heisele, T. Serre, S. Mukherjee and T. Poggio, "Feature Reduction and Hierarchy of Classifiers for Fast Object Detection in Video Images", *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 18–24, 2001.

[29] F. Han, Y. Shan, R. Cekander, H. Sawhney and R. Kumar, "A Two-Stage Approach to People and Vehicle Detection with Hog-Based Svm", *Performance Metrics for Intelligent Systems Workshop in conjunction with the IEEE Safety, Security, and Rescue Robotics Conference*, pp. 133-140, 2006.

[30] X. Perrotton, M. Sturzel and M. Roux, "Automatic Object Detection on Aerial Images Using Local Descriptors and Image Synthesis", *Proceedings of the 6th International Conference on Computer Vision Systems*, pp. 302-311, 2008.

[31] H. Cai and Y. Su. "Airplane Detection in Remote Sensing Image with a Circle-frequency Filter", *Proceedings of the SPIE*, vol. 5985, pp. 529-534, 2005.

[32] T. T. Nguyen, H. Grabner, B. Gruber and H. Bischof, "On-line Boosting for Car Detection from Aerial Images", 2007 IEEE International Conference on Research, Innovation and Vision for the Future, pp. 87-95, 2007.

[33] H. Moon, R. Chellappa and A. Rosenfeld, "Performance Analysis of a Simple Vehicle Detection Algorithm", *Image and Vision Computing*, vol. 20, No. 1, pp. 1-13, 2002.

[34] H. Ming-Kuei, “Visual pattern recognition by moment invariants”, *IRE Transactions on Information Theory*, vol.8, No.2, pp.179-187, 1962.

[35] F. Zernike, *Physica*, vol. 1, pp. 689, 1934.

[36] J. Ricard, D. Coeurjolly and A. Baskurt, “Generalization of Angular Radial Transform”, *International Conference on Image Processing*, vol.4, pp. 2211- 2214, 2004.

[37] D. Zhang and G. Lu, “A Comparative Study of Curvature Scale Space and Fourier Descriptors for Shape-based Image Retrieval”, *Journal of Visual Communication and Image Representation*, vol. 14, No. 1, pp. 39-57, 2003.

[38] T. Zhao and R. Nevatia, “Car detection in low resolution aerial image”, *Proceedings of IEEE International Conference on Computer Vision*, vol. 1, pp. 710-717, 2001.

[39] J.Y. Choi and Y.K. Yang, “Vehicle Detection from Aerial Images Using Local Shape Information”, *Proceedings of the 3rd Pacific Rim Symposium on Advances in Image and Video Technology*, pp. 227-236, 2008.

[40] J. Iisaka and T.S. Amano, “A Shape-based Object Recognition for Remote Sensing”, *Geoscience and Remote Sensing Symposium, IGARSS*, 1995.

[41] L. Eikvil, L. Aurdal and H. Koren, “Classification-Based Vehicle Detection in High-Resolution Satellite Images”, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 64, No. 1, pp. 65-72, 2009.

[42] J. W. Hsieh, J. M. Chen, C. H. Chuang and K. C. Fan, "Aircraft Type Recognition in Satellite Images", *Processing of IEEE Vision, Image and Signal Processing*,vol.152, No. 3, pp. 307-315, 2005.

[43] I. T. Jolliffe, *Principal Component Analysis*, Springer, 2002.

[44] A.Hyvarinen, J. Karhunen and E. Oja, *Independent Component Analysis*, John Wiley & Sons, 2001.

[45] D. D. Lee and H. Seung, "Learning the Parts of Objects by Non-Negative Matrix Factorization", *Nature*, vol. 40, pp. 788-791, 1999.

[46] R. Lienhart and J. Maydt, "An Extended Set of Haar-Like Features for Rapid Object Detection", *Proceedings of International Conference on Image Processing*, vol.1, pp. 900- 903, 2002.

[47] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting", *Journal of Computer and System Sciences*, vol. 55, No. 1, pp. 119–139, 1997.

[48] R. Polikar, "Ensemble Based Systems in Decision Making", *IEEE Circuits and Systems Magazine*, vol.6, No.3, pp. 21-45, 2006.

[49] W. K. Pratt, *Digital Image Processing*, John Wiley & Sons, 2007.

[50] D. Arslan and A. A. Alatan, "A Computationally Efficient Appearance-based Algorithm for Geospatial Object Detection", *Proceedings of SPIE*, vol. 8398, 2012.

[51] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, No. 1, pp. 62- 66, 1979.

[52] C. Yizong, “Mean Shift, Mode Seeking, And Clustering”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, No. 8, pp. 790-799, 1995.

[53] A. Ertürk, “Rotation, Scale and Translation Invariant Automatic Target Recognition Using Template Matching For Satellite Imagery”, M. Sc. Thesis, Middle East Technical University, Ankara, 2010.