Commun.Fac.Sci.Univ.Ank.Series A2-A3
Volume 58, Number 2, Pages 28-49 (2016)
DOI: 10.1501/commua1-2_0000000095
ISSN 1303-6009

ROTATION, SCALING AND TRANSLATION INVARIANT OBJECT RECOGNITION IN SATELLITE IMAGES

Yusuf SOYMAN[1], Hakkı Alparslan ILGIN[2]

[1]Siemens, İstanbul, TURKEY, E-mail: yusufsoyman@gmail.com,
[2]Corresponding author,
Ankara University, Faculty of Engineering, Electrical and Electronics Eng. Dept., 06830, Gölbaşı, Ankara, TURKEY
E-mail: ilgin@eng.ankara.edu.tr

(Received: October 31, 2016; Accepted: December 20, 2016)

ABSTRACT
In satellite imagery, different shots may be taken according to viewing angle, rotation and scale. Algorithms used to recognize objects in satellite images should find them without being affected from these variations. In this paper, rotation, scaling and translation invariant object recognition in satellite imagery is performed. The criterions affecting the performance of such a system are investigated in a strictly controlled environment. As a result of experimental studies with the usage of different parameters, successful performances have been attained. In consequence of contrasting studies the effects of these parameters upon object recognition have been emphasized.

KEYWORDS: Object recognition, RST invariance, feature extraction, classification, clustering, Bag of Visual Words (BOVW), rigid body, satellite imagery.

1. INTRODUCTION

Object recognition is a critical task in computer vision systems, which finds a specific object in an image or video. Rigid and non-rigid object recognition methods are proposed in the literature [1], [2], [3]. While some of them recognize the object using segmentation, other groups of methods accomplish it without segmentation process [4], [5].

In this paper, feature based rigid body object recognition method is examined. Firstly, training and evaluation data sets, which belong to target and non-target classes, are constituted with great care. The proposed method and experimental results are then discussed. The organization of the paper is as follows. Properties of training and evaluation data set are mentioned in Section 2. Feature extraction techniques are explained in Section 3. Section 4 elaborates on Bag of Visual Words (BOVW) approach to form a visual dictionary, and mapping the features of training data set to the visual dictionary is described. Training of two different classifiers is performed in Section 5. Prediction of the evaluation data set exploiting the trained classifiers is performed and experimental results are given in Section 6. Finally, conclusion is given in Section 7.

2. TRAINING AND EVALUATION DATA SET

Since the object recognition method should be rotation, scale and translation invariant, training data set should have wide variety of properties to represent compelling circumstances. Training and evaluation data sets, which approximately have 200 images with different properties, are constituted with this regard.

Data set should cover not only the rotated and scaled images but also low-contrast and cluttered images. All the images are taken in visible region. Evaluation data set possesses images with smaller than 45 degree of viewpoint change. Data set is prepared for two-class classification in terms of target or non-target decision. Targets are selected as airplane images. On the other hand, non-target images are selected as car images. Some of training data, which constitutes target and non-target classes, are shown Fig. 1 and Fig. 2, respectively.

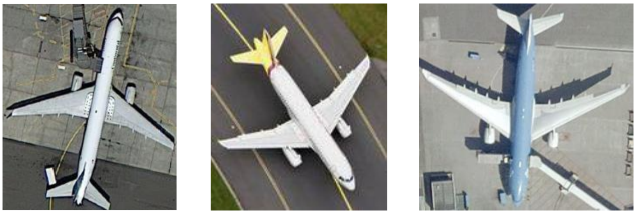

Fig. 1: Training data samples for target class

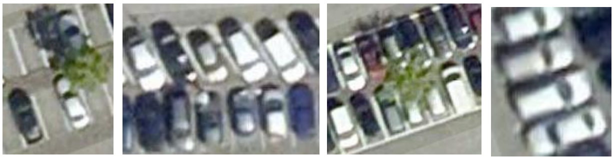

Fig. 2: Training data samples for non-target class

3. FEATURE EXTRACTION

Feature extraction is performed to acquire salient parts utilizing training data set. Scale Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF) are chosen as scale- and rotation-invariant interest point detector and descriptors. They provide robust object representation. Detailed information is given in the following section concerning with the features.

3.1. SCALE INVARIANT FEATURE TRANSFORM (SIFT)

SIFT is invariant not only the scale and rotation but also viewpoint and illumination changes. In order to get the SIFT features of objects the algorithm proposed by Lowe [6] is applied. Flow of the algorithms is summarized as follows:

1) Constructing a Scale Space

Initial preparation step needs to be applied on the image to ensure scale invariance. This process is carried out by investigating for the stable points across different scales of a scale space. The objective is accomplished by generating four octaves of the original image. Each octave’s image is down-sampled by factor of 2 of the previous one. Images are progressively convolved with a variable scale Gaussian filter within an octave;

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y)\ , \qquad (3.1)$$

where *L* is the blurred image, *G* is the Gaussian operator which is given below, *I* is the image, * is the convolution operator, *x-y* are location coordinates and *σ* is the scale factor

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \qquad (3.2)$$

2) Laplacian of Gaussian (LoG) Approximation

Laplacian of Gaussian (LoG) is abundant for finding keypoints in an image. Due to the fact that it is computationally expensive, an efficient way is proposed to approximate it by Lowe [6]. The idea is to use scale-space extreme in the Difference-of-Gaussian (DoG) function convolved with the image, *D(x,y,σ),* which can be calculated from the difference of two nearby scales separated by a constant multiplicative factor *k*:

$$\begin{aligned} D(x,y,\sigma) &= (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) \\ &= L(x,y,k\sigma) - L(x,y,\sigma) \end{aligned} \tag{3.3}$$

DoG images are obtained by differentiating two consecutive images on the same octave. Acquired images are approximately identical with the Laplacian of Gaussian images with the advantages of both low computational complexity and scale invariance.

3) Accurate Keypoint Localization

Keypoints are the maxima and minima points in the DoG image. Detecting keypoints are accomplished first by locating maxima and minima in DoG images following by detecting subpixel maxima and minima.

Maxima and minima of the DoG images are detected by comparing a pixel to its 26 neighbors in 3x3 regions at current and adjacent scales [6]. The pixel is selected as a “keypoint” if it is the maximum or minimum of all 26 neighbors. Since they do not have sufficient neighbors to perform the comparison, keypoints are not detected in the lowermost and topmost scales.

Due to the fact that the maxima and minima almost never lie exactly on the previously found pixel, they need to be interpolated to the approximate pixels. Therefore, approximation is done by locating the subpixel location. This process is performed by generating subpixels from the available ones [7]. Taylor expansion of the image around the keypoint is used to obtain the extreme points using following formula

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x . \tag{3.4}$$

Subpixel keypoint location is obtained by solving Taylor expansion of the image. These subpixels increase chances of matching and stability of the algorithm.

4) Eliminating edge and low-contrast responses

Keypoints, which are detected in the previous step, generally result with lots of occurrences. While some of them do not have enough contrast, some of them lie along the edge. In both cases, they are not useful as features. Therefore, keypoints are rejected if they have low-contrast or if they lie along an edge. Eliminating these keypoints makes the algorithm more efficient and robust [6].

Two gradients, perpendicular to each other at the keypoints, are calculated around the keypoints. If the corner is encountered around the keypoint instead of flat region or edge, point is selected as the keypoint. On the other hand, if the magnitude of the intensity at the current pixel in the DoG image is less than a threshold value, it is rejected. Finally, more robust keypoints are acquired with less number of them to deal with.

5) Identifying Keypoint Orientation

The keypoints are still sensitive to rotation changes even if they already have scale invariance. Satellite images are affected from these changes resulting in wrong recognition output. In this step, keypoints are made invariant to rotation by defining an orientation for each keypoint. Firstly, gradient directions and magnitudes for all pixels around the keypoint are calculated as follows [8]

$$\theta(x,y)=\tan^{-1}((L(x,y+1)-L(x,y-1))\,/\,(L(x+1,y)-L(x-1,y))) \qquad (3.5)$$

$$m(x,y)=\sqrt{(L(x+1,y)-L(x-1,y))^2+(L(x,y+1)-L(x,y-1))^2} \qquad (3.6)$$

where θ is gradient orientation and m is gradient magnitude. Then, an orientation histogram is created with these parameters. In the orientation histogram, the 360 degrees of orientation are broken into 36 bins and magnitude of gradient at that keypoint is proportional quantity that is accumulated inside the bin. After these calculations are performed for all the pixels around the keypoint, the orientation histogram has a peak at some point. That bin is defined as orientation of keypoint. However, if there is any peak above the 80% of the highest peak, it is selected as new keypoint. Although this new keypoint has the same location and scale as the original one, its orientation is equal to its own bin. Finally, the most prominent orientations are identified.

6) Descriptor Representation

Keypoint description should be performed to make the keypoints very unique. First of all, a 16x16 window fits around the keypoint. This 16x16 window is broken into sixteen 4x4 windows. Gradient magnitudes and orientations are calculated within each 4x4 window. The extracted orientations are put into an 8 bin orientation histogram. The magnitude of gradient is strongly related with the quantity accumulated inside the bin. Unlike the previous one, the quantity accumulated also strongly related with the distance from the keypoint. Therefore, the gradients that are far away from the keypoint have less effect while the near gradients have more effect to the orientation histogram. This process is performed with Gaussian weighting function. The weighted magnitude of orientation is obtained by multiplying magnitude of orientation with the Guassian weighting function. This procedure is applied for all sixteen 4x4 regions as shown in Fig 3.

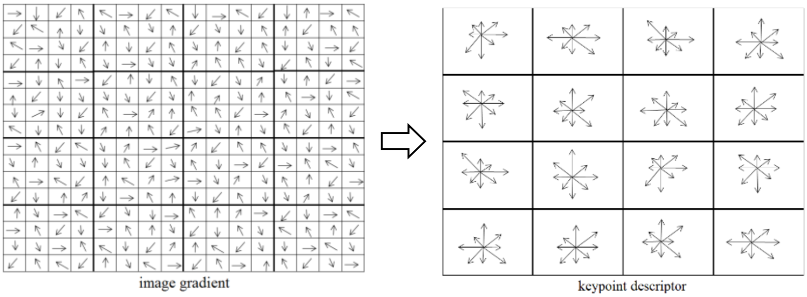

Fig. 3: Descriptor Representation

Once 128 numbers are acquired, normalization is then carried out. These 128 numbers form the feature vector [6]. From now on, the keypoint is uniquely identified by this feature vector. However, this feature vector has a few complications such as rotation and illumination dependence. Keypoint's rotation is subtracted from each orientation to achieve rotation invariance. Illumination invariance is achieved by thresholding the values. Any value of the feature vector that is greater than 0.2 is redefined as 0.2. Final step for illumination invariance is to normalize the resultant feature vector.

3.2. SPEEDED UP ROBUST FEATURES (SURF)

SURF is a robust and fast algorithm providing similarity invariant representation [16]. SURF detects interest points of an image from the salient features. Initial image with box filters at several scales are convolved. A series of images similar to SIFT approach are acquired. The histograms of gradient-like local operators defined as feature descriptor.

1) Feature Detection

Herbert et. al. [16] proposed Hessian-matrix based detector since it provides not only computational efficiency, but also good accuracy [16]. Hessian matrix *H(x, y, σ)* at a point *(x,y)* in an image *I* at scale σ is defined as follows;

$$\mathrm{H}(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix}. \tag{3.7}$$

where $L_{xx}(x,y,\sigma)$ is the convolution of the image *I* in point (x, y) with second order derivative of Gaussian which is defined as below:

$$\begin{aligned} L_{xx}(x,y,\sigma) &= I(x,y) * \nabla^2 G(x,y,\sigma) \\ \nabla^2 G(x,y,\sigma) &= \frac{\partial^2}{\partial x^2} G(\sigma) \end{aligned} \quad . \tag{3.8}$$

$L_{xy}(x,y,\sigma)$ and $L_{yy}(x,y,\sigma)$ are calculated similar to $L_{xx}(x,y,\sigma)$. Koenderink shows that Gaussians are optimal in scale space analysis [18]. Drawbacks of using Gaussian in practice are that they need to be discretized and cropped, and aliasing still occurs on the images [16]. Therefore, approximated version of it is exploited rather than using Gaussian. Approximated convolutions are shown as D_{xx}, D_{xy} and D_{yy}. Therefore, determinant is calculated with

$$\det(\mathrm{H}_{approximate}) = D_{xx}D_{yy} - (0.9D_{xy})^2 \ . \tag{3.9}$$

Normalization is then carried out with respect to filter size. Implementation of box filter is performed using integral images to decrease computation time. $I_{\square}$ *(x,y)* integral images are acquired by adding all the pixels of input image *I* of a box filter formed by pixel positions (*x*,*y*) [16]. Integral image is defined as follows

$$I_{\Sigma}(x,y) = \sum_{p=1}^{p \le x} \sum_{q=1}^{q \le y} I(p,q) \, . \tag{3.10}$$

Scale space constructing procedure is different from the SIFT. Filters with various sized are applied to input image to obtain the cascaded Gaussian convoluted images [16]. In the SIFT, these Gaussian smoothed images are acquired by applying Gaussian filtering to the previously smoothed one. However, SURF does not need to wait for any images to process with. Moreover, all the varying sized filters can be applied parallel to reduce computation time.

Scale spaces are usually implemented by sub-sampling the images resulting with image pyramid. Image pyramid in SURF is obtained by using up-scaling the filter size rather than sub-sampling the image. Sizes of the filter are 9x9, 15x15, 21x21, 27x27, etc. [16]. Variances of Gaussian on different levels are increased accordingly, starting from σ = 1.2 and continues with the scales of it.

Non-maximum suppression in 26 neighbors is performed to localize feature points in the image. Interpolation is carried out in scale and image space to obtain maxima of the determinant of the Hessian matrix [19].

2) Feature Description

Descriptors are acquired using distribution of Haar wavelet responses around the features that are attained in previous step. In order to provide rotation invariance, Haar wavelet responses in *x* and *y* directions are calculated. Integral images are used for quick filtering. Sum of all responses within a sliding orientation window covering 60° degree are calculated for dominant orientation [16]. The vertical and horizontal responses are summed to obtain a vector. Orientation of interest point is defined as the longest vector.

Squared region centered on the interest point is divided into 4x4 sub-regions. Haar wavelet responses in horizontal and vertical directions are summed up over each sub-region [16]. These entities constitute first part of feature vector. The sum of absolute values of responses constitutes remaining part of feature vector. Therefore, each sub-region has four-tuple vector and resulting in 64-dimensional descriptor vector for all 4x4 sub-regions.

The reason of selecting SURF is due to its concise descriptor length. Whereas SIFT approach uses a descriptor consisting of 128 floating point values, SURF condenses this descriptor length to 64 floating point values. The success of two feature extraction algorithm is compared in Section 6.

3.3. MODIFIED FEATURE EXTRACTION

Feature detection is also performed on modified images instead of raw images. Images are partitioned into a grid. Feature point in each cell are extracted using SIFT and SURF feature detectors. These methods are called GridSIFT and GridSURF, respectively. Other modification also performed during feature detection to detect feature points over multiple levels of Gaussian pyramids which are called PyramidSIFT and PyramidSURF.

SIFT and SURF descriptors only use information in intensity channel. However, it is known that rare color transitions could be very discriminative from information theory. Therefore, Sande et. al. [27] propose the usage of color information for feature description. In this manner, raw RGB image is converted into opponent color space as follows [21]:

$$\begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} = \begin{pmatrix} \dfrac{R-G}{\sqrt{2}} \\ \dfrac{R+G-2B}{\sqrt{6}} \\ \dfrac{R+G+B}{\sqrt{3}} \end{pmatrix} \tag{3.11}$$

where O_3 refers to intensity information. O_1 and O_2 channels are comprised of red – green and green – blue color information. However, the channels comprising color information also comprise intensity information. Therefore, they are not invariant to illumination changes. Finally, descriptors are extracted using SIFT and SURF descriptors on each of three channels. All of them are concatenated into a single color descriptor [27]. These descriptor methods are called OpponentSIFT and OpponentSURF. Effects of modified feature detectors and descriptors on object recognition are emphasized in Section 6.

3.4. MAXIMALLY STABLE EXTREMAL REGION (MSER)

MSER provides multi-scale detection result in detection of both small and large structure without smoothing. Matas et. al. [22] propose affine invariant method which is called Maximally Stable Extremal Regions (MSER) to cope with viewpoint changes. The regions in MSER are defined by extremal feature of intensity function in region and on its outer boundary [22].

1) Maximally Stable Extremal Region (MSER) Detection

Firstly, pixels are sorted using intensity values which are between 0 and 255. Pixels are then replaced in the image either in decreasing or increasing order. Connected components are listed using 4-neighbourhoods or adjacency [22]. Their areas are attained. Finally, thresholds are selected as intensity levels that are local minima of rate of change of area function. Each extremal region is represented a threshold and position of local maxima or minima [22].

2) Measurement Regions

Measurement regions are attained from invariant construction of extremal regions [22]. Each region may be associated with a measurement region. Although smaller regions are both satisfy the planarity condition and not to cross discontinuity in deep and orientation they are less discriminative. On the other hand, large regions have the risk of including background and occlusion. Obviously, scene content affects to optimal size of measurement regions. It is different for each region.

Measurement regions are selected at multiple scales such as itself of extremal region, 1.5, 2 and 3 times scaled convex hull of region. Measurement region matching techniques is chosen as Mahalanobis distance.

3) Invariant Description

In order to provide affine invariance, transformation is performed to diagonalise the covariance matrix of extremal region. Complex moments based rotational invariants [26] are then used. This cascaded procedure ensures affine invariance.

4) Robust Matching

Voting mechanism provides robust matching [22]. The regions possessing the largest number of votes are the candidates for tentative correspondences.

5) Tentative Correspondence using Correlation

Transformations that diagonalize the covariance matrix of regions are performed. Resulting circular regions are correlated in polar coordinates for different sizes of circles.

In [25], performance of some of region detectors such as Hessian-affine, Harris-affine, intensity extrema, edge based regions, salient regions and MSER is compared through many tests. MSER consistently gives highest score. Thus MSER is ensured as reliable region detector. In terms of view point change, MSER outperforms other region descriptors. MSER gives second best results under scale change and in-plane rotation following the Hessian-affine [25]. MSER suffers from blur changes. However, it is robust to illumination changes.

4 BAG OF VISUAL WORDS (BOVW)

The Bag-of-Word (BOW) model provides simplifying representation in information science. In BOW model, a document is represented as an unordered collection of words, disregarding grammar and word order [9]. That is sparse histogram of vocabulary. Moreover, studies in document classification reveal that grammar and word order are not as discriminative as frequency of each word.

BOW model can be adapted to image classification problem. Image patches (codewords) are the visual equivalents of words, and the image is treated as codebook (bag) of these words [10], [15]. In computer vision, a bag of visual words is vocabulary constitutes sparse vector of frequency of image features. After the features (codewords) are extracted in Section 3, codebook is generated using k-means++ clustering because of its simplicity, speed and accuracy [11], [12]. In this paper, codewords are chosen as 50, 100 and 250 to investigate effect of it on classification. Features of training data are extracted according to this codebook.

5 CLASSIFICATION

In this section, classifiers are trained with different parameters in terms of feature extraction, variable sized vocabulary and training and evaluation data sets. Bayes classifier and Support Vector Machines (SVM) are used as classifier for target and non-target decision.

5.1. BAYES CLASSIFIER

Bayes classifier minimizes the probability of misclassification [13]. Distribution of feature vectors of each class is estimated as normal. Therefore, the whole data distribution function is estimated to be Gaussian mixtures of classes which of each represent a model. Mean vectors and covariance matrices for every class are estimated using training data. Predictions are performed using them [14].

5.2 SUPPORT VECTOR MACHINES (SVM)

SVM fits a separating hyperplane to discriminate target class from non-target class. SVM maps feature vectors into a higher-dimensional space using a kernel function and builds an optimal linear discrimating function in this space or an optimal hyperplane that fits into the training data [17]. A hyperplane, which is shown in Fig. 4, is defined as follows:

$$f(x) = \beta_0 + \beta^T x \,, \tag{5.1}$$

where β is known as weight factor and β_0 as the bias.

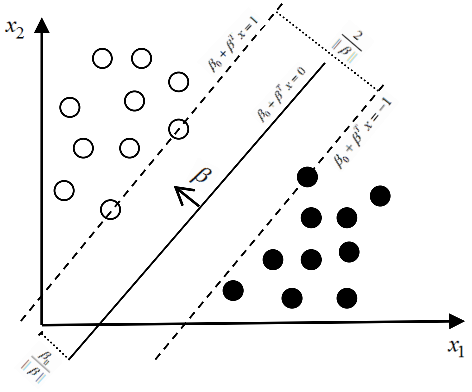

Fig. 4: Maximum-margin hyperplane

The optimal hyperplane can be represented in an infinite number of different ways by scaling of *β* and *β*0. The representation is chosen as [17]:

$$\beta_0+\beta^T x=1$$
$$\beta_0+\beta^T x=-1, \qquad (5.2)$$
$$|\beta_0+\beta^T x|=1$$

where *x* is training data closest to the hyperplane. In general, the training examples that are closest to the hyperplane are called support vectors [17]. Distance between a point *x* and the hyperplane *(β, β0)* is calculated as:

$$\frac{|\beta_0+\beta^T x|}{\|\beta\|}=\frac{1}{\|\beta\|}. \qquad (5.3)$$

$$M=\frac{2}{\|\beta\|}. \qquad (5.4)$$

where *M* is distance between the support vectors each of which belongs to different class and 1/||β|| is margin. Margin is the distance between the support vectors and separating hyperplane. Finally, maximizing the margin is obtained by minimizing the *L(β)* function [17].

$$\min_{\beta,\beta_0} L(\beta)=\frac{1}{2}\|\beta\|^2 \qquad (5.5)$$

$$y_i(\beta^T x_i + \beta_0) \geq 1 \ \forall i, \tag{5.6}$$

where y_i represents each of the labels of the training data. Using the Lagrange multipliers α_i, previous constraint problem can be expressed as an unconstrained optimization problem [23]

$$J(\beta, \beta_0, \alpha) = \frac{1}{2} \| \beta \|^2 + \sum_{i=1}^{n} \alpha_i \left\{ 1 - y_i(\beta^T x_i + \beta_0) \right\}, \tag{5.7}$$
$$\alpha_i \geq 0$$

which leads to primal form of objective function. J is minimized with respect to β and β_0 and maximized with respect to α_i. The optimization of $J(\beta, \beta_0 ,\alpha)$ is converted to a dual-quadratic problem by using Karush – Kuhn – Tucker (KKT) conditions [24], differentiating J with respect to β and β_0 and equating to zero yields

$$\sum_{i=1}^{n} a_i y_i = 0, \tag{5.8}$$

$$\beta = \sum_{i=1}^{n} a_i y_i x_i \ . \tag{5.9}$$

Substituting into (5.7) gives the dual form of Lagrangian

$$W(a) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j, \tag{5.10}$$

which is maximized with respect to α_i the subject to eq. (5.8). As shown in eq. (5.10), optimization criterion can be expressed as inner product of patterns x_i. The number of pattern is shown with n, and α_i is the Lagrange multipliers. The optimal hyperplane $f(x)$ is given by

$$f(x) = \sum_{i \in SV} \alpha_i y_i (x_i^T x_i) + \beta_0, \tag{5.11}$$

$$\beta_0 = y_i - \beta^T x_i, \tag{5.12}$$

where SV is a set of support vector, i is support vector index. For nonlinearly separable patterns, eq. (5.5) is extended with a positive slack variable ξ as follows

$$y_i(\beta^T x_i + \beta_0) \geq 1 - \xi_i \ \forall i \\ \xi_i \geq 0 \quad , \tag{5.13}$$

For a point to be misclassified by separating hyperplane as shown in Fig. 5, ξ must be bigger than 1.

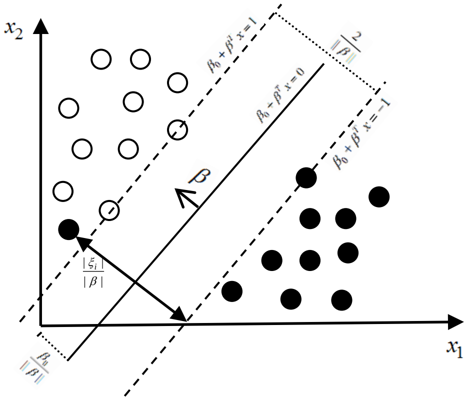
x_2
$\beta_0 + \beta^T x = 1$
$\frac{2}{\|\beta\|}$
$\beta_0 + \beta^T x = 0$
$\beta_0 + \beta^T x = -1$
β
$\frac{|\xi_i|}{|\beta|}$
x_1
$\frac{\beta_0}{\|\beta\|}$

Fig. 5: Linear separating hyperplane for non-separable data

Extra cost term C is included in the cost function as follows

$$J(\beta, \beta_0, \xi) = \frac{1}{2}\|\beta\|^2 + C\sum_{i=1}^{n}\xi_i \ , \tag{5.14}$$

where C is regularization parameter, which specifies the balance between the margin maximization and the misclassification minimization [23]. The lower the value of C results in the smaller the penalty of outliers. Optimization problem is acquired by using the same procedure with linearly separable data except that α_i is bounded by the regularization factor [24].

Feature space is nonlinearly transformed to higher-dimensional space in which linear methods could be accomplished if set of data in feature space is nonlinearly distributed. A kernel function, $K(x_i, x_j)$, is used instead of direct computations of $g(x)$ to increase dimensionality of data as shown below

$$K(x_i \, x_j) = g^T(x_i)g(x_j) \ . \tag{5.15}$$

Decision function using kernel is defined as follows

$$f(x) = \sum_{i \in SV} \alpha_i y_i K(x_i \, x_j) + \beta_0 \tag{5.16}$$

$$\beta_0 = y_i - \sum_{i \in SV} \alpha_i y_i K(x_i \, x_j) \tag{5.17}$$

Radial Basis Function (RBF) is chosen as kernel function as shown below [23]:

$$\begin{aligned} K(x_i \, x_j) &= e^{-\gamma \|x_i - x_j\|^2} \\ \gamma &= \frac{1}{2\sigma^2} \end{aligned} \tag{5.18}$$

Using radial basis function as kernel generally results in good separating hiperplane for nonlinearly separable data. Once separating hyperplane is acquired according to training data, predictions are performed using evaluation data.

6 EXPERIMENTAL RESULTS

In this section, recognition performances of the combined methods, which are obtained using various parameters, are given. False Alarm Rate (FAR) is used as evaluation metric. Type of SVM kernel selected as Radial Basis Function (RBF). Only first two results in Table 1 use linear kernel. Table 1 shows the results while both training and evaluation data sets are at same resolution which is 10 meters.

Tab.1: Experimental results at the same scale

Feature Extraction	#Training Data (Target vs Non-target)	#Evaluation Data (Non-target vs Target)	#Vocabulary	SVM FAR (%)	Bayes FAR (%)
SIFT	50 – 50	25 – 17	250	4.76	4.76
SIFT	50 – 50	25 – 17	100	2.38	2.38
SIFT	50 – 50	25 – 17	100	0	2.38
SURF	50 – 50	25 – 17	100	7.14	2.38
SIFT	50 – 50	25 – 17	50	2.38	2.38
SURF	50 – 50	25 – 17	50	4.76	14.29
SIFT	25 – 25	25 – 17	100	0	2.38
SIFT	25 – 25	50 – 42	100	1.09	4.35
SIFT	25 – 25	50 – 42	250	0	2.17
SURF	25 – 25	50 – 42	250	10.87	10.87

In Table 1, best detection and lowest FAR is attained with SIFT feature extraction method using SVM classification technique. Experiments show us that selection of kernel function as radial basis function provides better discrimination capability. Moreover, Bayes results in worse FAR compared to SVM when we decrease training data and increase evaluation data. Besides, there is no unique vocabulary number. This parameter depends on not only variation of training data but also feature extraction and classification methods. Fig. 6 shows the graphical representation of Table 1 [20]. Vertical axis represents FAR. Horizontal axis indicates properties of implemented experiments. Results obtained using SVM methods are shown with blue color, which is at the left side. There is no blue bar for some of the results due to zero false alarm rates. On the other hand, red color represents Bayes results.

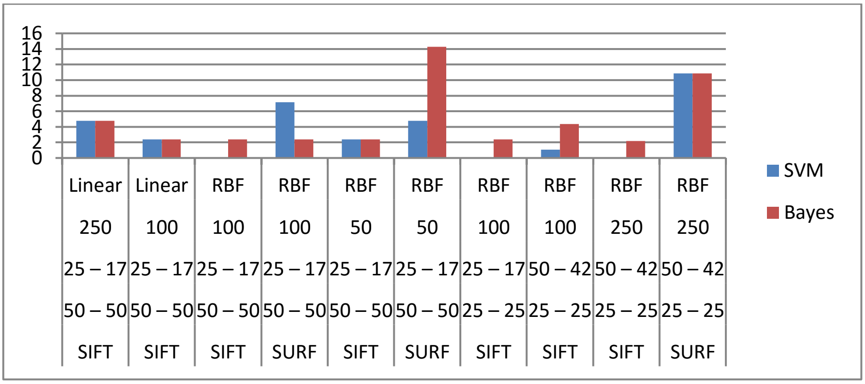

Fig. 6: False alarm rate at the same scale

Experiments are performed to test recognition performance of affine-invariant detector MSER with various descriptors in Table 2. SVM generally provides better FAR compared to Bayes as in the previous experiment. MSER provides relatively better feature detection capability over SIFT and SURF feature detection methods. In this experiment, we also study on alternative feature description technique which is Opponent SIFT and Opponent SURF. Since training data sets include color images, we use these methods in order to test the effects of color on describing features. Although same FAR is achieved with Opponent SIFT compared to SIFT, SURF has better feature description capability over Opponent SURF. Therefore usage of these techniques does not give any significant advantages over SIFT and SURF themselves.

Tab.2: False alarm rates obtained using affine-invariant detector at the same scale

Feature Detection / Description	#Training Data (Target vs Non-target)	#Evaluation Data (Non-target vs Target)	#Vocabulary	SVM FAR (%)	Bayes FAR (%)
MSER / SIFT	50 – 50	25 – 17	100	2.38	0
MSER / SIFT	50 – 50	25 – 17	250	2.38	0
MSER / Opponent SIFT	50 – 50	25 – 17	100	0	4.76
MSER / SURF	50 – 50	25 – 17	100	0	2.38
MSER / SURF	50 – 50	25 – 17	250	2.38	4.76
MSER / SURF	50 – 50	25 – 17	50	0	14.29
MSER / Opponent SURF	50 – 50	25 – 17	50	7.14	9.52

Table 3 shows the FAR obtained at different scales. Evaluation is performed with 20 meter resolution data sets while training of classifiers performed with 10 meter resolution data sets. Type of SVM kernel is also selected as radial basis function (RBF). Training data set contains 67 and 58 images in terms of target and non-target images. On the other hand, evaluation data set contains 17 and 53 in terms of non-target and target images.

Tab.3: False alarm rates at different scales

Feature Detector	Feature Descriptor	#Vocabulary	SVM FAR (%)	Bayes FAR (%)
SIFT	SIFT	50	**4.29**	42.86
SIFT	SIFT	100	**4.29**	7.14
SURF	SURF	50	**18.57**	55.71
SURF	SURF	100	10.00	**5.71**
SIFT	Opponent SIFT	100	**2.86**	7.14
SIFT	Opponent SIFT	50	**7.14**	44.29
Pyramid SIFT	SIFT	50	**4.29**	47.14
Pyramid SIFT	SIFT	100	**1.43**	4.29
GridSIFT	SIFT	100	8.57	**4.29**
GridSIFT	SIFT	50	**7.14**	61.43
Pyramid SIFT	Opponent SIFT	50	**4.29**	8.57
Pyramid SIFT	Opponent SIFT	100	**0**	7.14
Pyramid SURF	Opponent SURF	100	**10.00**	12.86
SURF	Opponent SURF	100	8.57	**4.29**
Pyramid SURF	SURF	100	7.14	7.14
Grid SURF	SURF	100	16.13	**3.23**
MSER	SIFT	100	**55.71**	58.57
MSER	Opponent SIFT	100	**54.29**	64.29
MSER	Opponent SURF	100	**24.29**	31.43
MSER	SURF	100	**20.00**	37.14

Experiments on Table 3 show that alternative feature detection and description techniques are worth to study on. Although Pyramid SIFT as a feature detection method results in better FAR in terms of both SVM and Bayes, usage of Opponent SIFT as a feature description method has better FAR only for SVM technique. On the other hand, while MSER provides lower FAR with experiments on the same scale, it gives worse results on the images taken at different scales.

7 CONCLUSION

In this paper, an extensive study for airplanes from satellite image recognition is performed. Experimental results reveal that number of visual words in the dictionary, feature descriptor and classification type affects the object recognition performance.

SIFT-based descriptors generally outperform SURF-based descriptors. However, SURF has similar recognition performance with SIFT in some situations, while at the same time being much faster.

MSER detector provides the same or better discrimination at the same scale. Conversely, the experiments at different scales reveal that MSER suffers from scale changes [22].

Number of visual words included in the dictionary is directly proportional with the recognition performance. Too small-sized vocabulary causes underfitting (high bias) problem owing to inadequate representation of all patches. Moreover, vocabulary with large size causes overfitting (high variance) and quantization artifacts. Therefore, selecting the optimum number of visual word is important.

Support Vector Machines (SVM) generally outperforms Bayes classifier. Moreover, SVM performance with restricted training data gives better and prominent results. Nevertheless, there are rare cases that Bayes classifier gives better classification performance. Results attained on the tables demonstrates that usage of better feature description techniques can result in better recognition performance than usage of sophisticated classification techniques. Therefore usage of better feature description techniques has more impact on the results than the sophisticated classification algorithms. If the training and evaluation data are at the same scale, MSER is preferable as feature detection technique. If they are at different scales, Pyramid SIFT is preferable. There is not exactly better method in terms of feature description. Number of vocabulary also should be adjusted for application. As a classification method, SVM is especially preferable for applications with restricted data. While Bayes gives better results with wider data sets owing to big number of samples since there is at least one sample from each data set, SVM, being deterministic, practically obtains lower false alarm rates since the possibility of acquiring data during training stage at each scenario is very low in real-life.

RERERENCES

[1] NGUYEN, Duc, OGUNBONA, Philip. *A novel shape-based non-redundant local binary pattern descriptor for object detection.* Pattern Recognition, May 2013. ISSN 0031-3203

[2] BESL, Paul, MCKAY, Neil. *A Method for Registration Of 3-D Shapes*. IEEE Transactions On Pattern Analysis And Machine Intelligence, FEB 1992, ISSN 0162-8828

[3] CHUI, Haili, RANGARAJAN, Anand. *A new point matching algorithm for non-rigid registration.* Computer Vision and Image Understanding. Mar 2003, ISSN 1077-3142

[4] LEPETIT, Vincent, FUA, Pascal. *Keypoint recognition using*

randomized trees. IEEE Transactions on Pattern Analysis and Machine Intelligence. SEP 2006. ISSN 0162-8828

[5] SOYMAN, Yusuf. *Robust Automatic Target Recognition in FLIR imagery*. Automatic Target Recognition XXII, Proceedings of SPIE, 2012. ISBN 978-0-8194-9069-8.

[6] LOWE, David. *Distinctive Image Features from Scale-Invariant Keypoints*. International Journal of Computer Vision, 2004. ISSN 0920-5691.

[7] LOWE, David. *Fit a quadratic to surrounding values for sub-pixel and sub-scale interpolation*. 2002.

[8] LOWE, David. *Towards a computational model for object recognition in IT cortex*. Proc. Biologically Motivated Computer Vision, pages 2031, 2000.

[9] TOLDO, Roberto, CASTELLANI, Umberto, FUSIELLO, Andrea. *Visual vocabulary signature for 3D object retrieval and partial matching*. Proceedings of the 2nd Eurographics conference on 3D Object Retrieval. March 29, 2009, Munich, Germany. ISBN 978-3-905674-16-3

[10] NOWAK, Eric, JURIE, Frederic and TRIGGS, Bill. *Sampling Strategies for Bag-of-Features Image Classification*. 9th European Conference on Computer Vision, Graz, Austria. 2006. ISBN 978-3-540-33838-3.

[11] LEUNG, Thomas and MALIK, Jitendra. *Representing and recognizing the visual appearance of materials using three-dimensional textons*. International Journal of Computer Vision. 2001. ISSN 0920-5691

[12] ARTHUR, David, VASSILVITSKII, Sergei. "*k-means plus plus: the advantages of careful seeding*". Proceedings of the eighteenth annual ACM-SIAM symposium on discrete algorithms. Society for Industrial and Applied Mathematics Philadelphia, PA, USA. pp. 1027–1035. 2007. ISBN 978-0-898716-24-5

[13] DEVROYE, Luc, GYORFI, Laszio, LUGOSI, Gabor, *A*

probabilistic theory of pattern recognition. 1996. Springer. ISBN 0-3879-4618-7.

[14] FUKUNAGA, Keinosuke. *Introduction to Statistical Pattern Recognition*. Second edition, New York: Academic Press, 1990. ISBN 0-12-269851-7

[15] YOUNGJOONG, Ko. *A study of term weighting schemes using class information for text classification*. SIGIR'12. ACM. 2012. ISBN 978-1-4503-1472-5

[16] BAY, Herbert, ESS, Andreas, TUYTELAARS, Tinne, VAN GOOL, Luc. *Speeded-Up Robust Features (SURF)*. Computer Vision and Image Understanding. 2008. ISSN 1077-3142

[17] BURGES, Christopher. *A tutorial on Support Vector Machines for pattern recognition*. Data Mining and Knowledge Discovery. 1998 ISSN 1384-5810

[18] KOENDERINK, Jan. *The structure of images*. Biological Cybernetics 50. 363 – 370. 1984. ISSN 0340-1200.

[19] BROWN, Matthew, LOWE, David. *Invariant features from interest point groups*. BMVC. 2002.

[20] SOYMAN, Yusuf, ILGIN, Hakkı, Alparslan, *Rotation, Scaling and Translation Invariant Object Recognition*. KTTO, 2013.

[21] VAN DE WEIJER, Joost and GEVERS, Theo. *Boosting saliency in color image features*. In IEEE Conference on Computer Vision and Pattern Recognition, volume 1, pages 365–372, San Diego, USA, 2005. ISSN 0162-8828

[22] MATAS, Jiri, CHUM, Ondrej, URBAN, Martin and PAJDLA, Tomas. *Robust wide baseline stereo from maximally stable extremal regions*. Proc. of British Machine Vision Conference, pages 384-396, 2002. ISSN 0262-8856

[23] FLETCHER, Robert. *Practical Methods of Optimization*. John Wiley & Sons, Ltd, 1988.

[24] WEBB, Andrew, COPSEY, Keith. *Statistical Pattern Recognition*. 2011. ISBN 978-0-470-68227-2

[25] MIKOLAJCZYK, Krystian, TUYTELAARS, Tinne, SCHMID, Christine, ZISSERMAN, Andrew, KADIR, Timor and VAN GOOL Luc. *A Comparison of Affine Region Detectors*. International Journal of Computer Vision, Volume 65, Numbers 1-2 / November, 2005, pp 43-72 ISSN 0920-5691

[26] MATAS, Jiri, BILEK, Petr, CHUM, Ondrej, *Rotational invariants for wide-baseline stereo*. Proceedings of CVWW'02, February 2002, pp.296–305.

[27] VAN DE SANDE, Koen E.A. GEVERS, Theo, and SNOEK, Cees. *Color descriptors for object category recognition*. CGIV 2008.