OBJECT DETECTION USING SHARED LOCAL FEATURES

A DISSERTATION
SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
SIGNAL PROCESSING AND DIGITAL DESIGN

Submitted by

GUJJULA LINGA REDDY

2K13/SPD/05

Under the supervision of

Mr. DINESH K.VISHWAKARMA
(Assistant Professor)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042
2015

Certificate

This is to certify that the dissertation title ***"Object detection using shared local features"***submitted by **Mr G.Linga Reddy**, Roll. No. ***2K13/SPD/05***, in partial fulfilment for the award of degree of Master of Technology in Signal Processing & Digital Design at **Delhi Technological University, Delhi**, is a bonafide record of student's own work carried out by him under my supervision and guidance in the academic session 2013-15. To the best of my belief and knowledge the matter embodied in dissertation has not been submitted for the award of any other degree or certificate in this or any other university or institute.

Dinesh K.Vishwakarma
Supervisor
Assistant Professor
Dept. of ECE
Delhi Technological University

Acknowledgement

I am indebted to my thesis supervisor **Mr Dinesh K.Vishwakarma, Assistant Professor** Department of Electronics and Communication, for his gracious encouragement and very valued constructive criticism that has driven me to carry out the project successfully.

I am greatly thankful to **Prof. Prem R. Chadda,** Head of Department (Electronics & Communication Engineering), entire faculty and staff of Electronics & Communication Engineering and friends for their continuous support, encouragement and inspiration in the execution of this "**thesis"** work.

Finally I express my deep sense of gratitude to my parents who bestowed upon me their grace and were source of my inspiration and encouragement.

G.Linga Reddy
M.Tech (SPDD)
2K13/SPD/05

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

LBP	LOCAL BINARY OATTERN
CS-LBP	CENTRE SYMMETRIC LOCAL BINARY PATTERN
OCS-LBP	ORIENTED CENTRE SYMMETRIC LOCAL BINARY PATTERN
HOG	HISTOGRAM OF ORIENTED GRADIENTS
AP	AVERAGE PRECISION

ABSTRACT

To detect object which is in different views from the cluster of images is still a challenging task. In our proposed method, we developed a new method to detect an object from different views by using shared local features in partially occluded images. Here shared features are the common features which are obtained from different classes and these common features are trained jointly in order to reduce the no of classifiers to detect an object. First we select some random samples (rectangular boxes) with different sizes which cover entire image. Each sample is represented by centre point of rectangular box, length and width of box. After selecting random samples, we apply oriented centre symmetric local binary pattern(OCS-LBP) & HOG for each sample and trained by random forest classifier. Like wise apply the same procedure for different views and generate shred features from all different views. To detect an object view point and its location we use probabilistic method for all views and which has the highest probability that view becomes the view point of detected object. Our proposed method was successfully performed on PASCAL VOC 2012 dataset and obtained better results compared to other methods.

CHAPTER-1

INTRODUCTION

1.1 Introduction

Object detection is one of the challenging and interesting tasks in the fields of image processing and computer vision. Face detection and human detection have achieved successful results as of now in detection field but object detection from video, still images remains a difficult task. Since in the still images, videos there are so many objects are present which are not important . in order to recognise object from clutter of images we need to use specific features which are present in the particular object. These features distinguish object from clutter of objects as well as from background. For example in the image there are so many objects like car, human, background etc. our aim is to detect car from the image and now we need to extract features which are being different from other objects present in image. In this example we can extract the features from the car are windows, wheels etc..which are different from other objects.

To extract features for detecting object generally carried either from entire the object as a single feature vector or dividing the object into parts and then extract features from different parts. If we extract features from entire the object it is called 'global feature extraction' and if we extract features from different parts of the object it is called 'local feature extraction'. In global feature extraction method the classification is simple because feature extraction is not complex but in local feature extraction method the classification is difficult because the feature extraction is complex. While classifying objects when there are different backgrounds present in different images then global features is not able to classify efficiently but in this case local features are very robust in classifying the object . Local feature extraction is more robust for occluded objects compared to global feature extraction.

Now a days object detection is using in many fields such as video surveillance, object tracking and computer vision. In these fields the first step is to detect object from the occluded objects and if we are not able to detect object properly then remaining steps in these fields may not be able to track object properly. So object detection plays a major role in the field of image processing.

To extract features for detecting object generally carried either from entire the object as a single feature vector or dividing the object into parts and then extract features from different parts. If we extract features from entire the object it is called ‘global feature extraction’ and if we extract features from different parts of the object it is called ‘local feature extraction’. In global feature extraction method the classification is simple because feature extraction is not complex but in local feature extraction method the classification is difficult because the feature extraction is complex. While classifying objects when there are different backgrounds present in different images then global features is not able to classify efficiently but in this case local features are very robust in classifying the object . Local feature extraction is more robust for occluded objects compared to global feature extraction.

Generally representation of local features and global features is different that means the local features represent the textures and global features represent the contour parts. Local features are like coloured local features, texture local features, shape local features. Here one point is to be noted that we can use shape features as local features as well as global features . if we consider entire image for feature extraction by using shape features then it is considered global feature extraction and if we consider different parts of the image for feature extraction by using shape features then it is called local feature extraction. From this we can say that global features can commit more errors compared to local features.

Mainly either global features or local features are calculated by extracting three types of features. Those are coloured features, shape features, texture features . histogram feature extraction and local binary pattern features come under colour features. Gabor filtering comes under texture feature extraction where as histogram of oriented gradients(hog), SIFT features come under shape features. But these methods have many limitations when they are occluded. So it is needed to use part based models and local features for efficient extraction. part based models are models collected from different parts of image in deformable configuration. There are many advantages of using part based models like these give good results when object is partially occluded. And part models mostly exclude background from image so it is easy for detection window to detect required object.

1.2. Motivation And Objective :

After reading literature it is found that object detection is really a challenging task when the object is mainly varied by view point, occlusion and lighting. And also computational time is more for detecting objects. So in this proposed method in order to increase precision accuracy we used oriented centre symmetric local binary pattern (OCS-LBP) which reduces computational time considerably. And in order to detect object in any view we train images which are in different direction by using random forest classifier. Here we used shared local features which reduces the no of local features to reduce the computational time.

This aim of this thesis is to detect object with high precision accuracy and less computational time. The first step of this project is to select random seed locations with random sizes that means dividing the whole image into no of patches in randomly with different sizes. Then for each random patch find OCS-LBP descriptors as features. Like that find same features in different view point images covering total 0-360 orientation. Then by using random forest classifier train each input image which is divided into no of patches. In order to determine the object window more efficiently we train using random forest classifier instead of SVM or Adaboost.

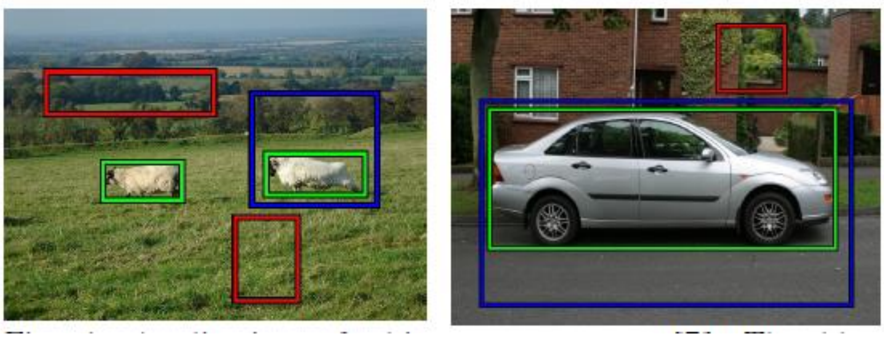

Fig.1.1. object detection windows

CHAPTER-2

LITERATURE REVIEW

The object detection research started in the late 1950s and in the early 1960s, The first object detection was carried by using auto correlation function and matching the templates. At that time the main classification was done in the field of recognising the character, analysing fingerprint and also in the field of classification for microscopic cell division. In these early decades the features selection was based on statistical pattern analysis derived from geometric description and classification was based on parametric learning method. At that time for extracting features moments are used in geometric description. Again geometric field played a major role in 1990s.

In a similar manner, local feature-based methods build class-specific clusters of local features with a similar appearance, which are then treated as object parts and combined spatially in a probabilistic manner [A.Thomas et al 2006] . in this paper it is presented a novel system for generic object class detection. In contrast to most existing systems which focus on a single viewpoint or aspect, our approach can detect object instances from arbitrary viewpoints. This is achieved by combining the Implicit Shape Model for object class detection proposed by Leibe and Schiele with the multi-view specific object recognition system of Ferrari et al. After learning single-view codebooks, these are interconnected by so-called activation links, obtained through multi-view region tracks across different training views of individual object instances. During recognition, these integrated codebooks work together to determine the location and pose of the object. Experimental results demonstrate the viability of the approach and compare it to a bank of independent single-view detectors.

[Leibe et al 2004] presented a two-stage object detection approach. In the first stage, a codebook of local appearances is learned, which contains information about the local structures that may appear in objects in the target category. Next, an implicit shape model is learned to specify where the codebook entries could occur on the object.

[Laptev,2009] introduced an object detection method for single views by combining AdaBoost learning with a local histogram feature. In this method, an exhaustive set of rectangular regions in the normalized object window is selected, and then, AdaBoost is used to select the histogram features and learn the object classifier. Although these methods produce good detection results for single-viewpoint objects, they still have a few limitations such as detecting view- independent objects and computational runtime. In this paper it is addressed address the problem of visual object class recognition and localization in natural images. Building upon recent progress in the field we show how histogram-based image descriptors can be combined with a boosting classifier to provide a state of the art object detector. Among the improvements we introduce a weak learner for multi-valued histogram features and show how to overcome problems of limited training sets.

One problem affecting object detection is how to handle view independent of a class rather than through category-level object detection, which is concerned only with finding single views of an object, such as frontal and profile views of faces and cars. To detect view-independent objects, [Torralba et al, 2007] presented a multi-task learning procedure based on boosted decision stumps, which reduces the computational and sample complexity by finding common features. These common features can be shared across classes, and the detectors used for each class are trained jointly rather than independently using common features. It is considered the problem of detecting a large number of different classes of objects in cluttered scenes. Traditional approaches require applying a battery of different classifiers to the image, at multiple locations and scales. This can be slow and can require a lot of training data since each classifier requires the computation of many different image features. In particular, for independently trained detectors, the (runtime) computational complexity and the (training-time) sample complexity scale linearly with the number of classes to be detected. We present a multitask learning procedure, based on boosted decision stumps, that reduces the computational and sample complexity by finding common features that can be shared across the classes (and/or views). The detectors for each class are trained jointly, rather than independently. For a given performance level, the total number of features required and, therefore, the runtime cost of the classifier, is observed to scale approximately logarithmically with the number of classes.

[Leibe et al,2008] introduced an object detection method that learns the appearance and spatial structure of a visual object category to recognize view-independent objects in that category, localizes them in cluttered real-world scenes, and segments them automatically from the background. However, these methods do not consider the deformation of objects. This paper presents a novel method for detecting and localizing objects of a visual category in cluttered real-world scenes. Our approach considers object categorization and figure ground segmentation as two interleaved processes that closely collaborate towards a common goal. the tight coupling between those two processes allows them to benefit from each other and improve the combined performance.

[Uijlings et al 2013] designed a three-step object detector that first selects candidate bounding-box size and viewpoint, and then rely on a view-specific classifier to validate these hypotheses and decide whether an object is present. This paper addresses the problem of generating possible object locations for use in object recognition. Selective search which combines the strength of both an exhaustive search and segmentation. Like segmentation, we use the image structure to guide our sampling process. Like exhaustive search, we aim to capture all possible object locations. Instead of a single technique to generate possible object locations, we diversify our search and use a variety of complementary image partitioning to deal with as many image conditions as possible. A mixture of multi-scale deformable part models (DPMs) for view-independent object detection was pro- posed by Felzenszwalb et al. [4]. This method uses discriminative training with partially labeled data on individual side-views of the objects.

[Gu and Ren 2010] use a mixture of holistic templates and discriminative learning for object viewpoint classification and category detection. Their research discriminatively incorporates a large mixture of templates inspired by the previous study using HOGs, and shows that the templates, which are directly used for viewpoint classification, correspond well to the canonical views of an object. object detection system that represents highly variable objects using mixtures of multi scale deformable part models. These models are trained using a discriminative procedure that only requires bounding boxes for the objects in a set of images. The resulting system is both efficient and accurate, achieving state-of-the-art results. Pictorial structures represent objects by a collection of parts arranged in a deformable configuration. Each part captures local appearance properties of an object while the deformable configuration is characterized by spring-like connections between certain pairs of parts.

Further, [Lopez- Sastre et al 2011] revisited the DPMs and improved the accuracy of object category pose estimation by designing different training strategies using a semi-latent support vector machine (SVM) learning methodology. Deformable Part Models (DPMs) as introduced by Felzenszwalb et al. have shown remarkably good results for category-level object detection. In this paper, it is explored whether they are also well suited for the related problem of category-level object pose estimation. To this end, paper extended the original DPM so as to improve its accuracy in object category pose estimation and design novel and more effective learning strategies. [Gu et al 2012] produced visual clusters by considering multi-view components of the data, which are similar in their appearance and configuration spaces. The authors trained individual classifiers for each component and learned a second classifier, which operated at the category level by aggregating the responses from multiple components.

[Wang and Lin 2011] presented a discriminative part-based model to represent and recognize object shapes using an And - Or graph. The And - Or graph model can handle large intra- class variances and background clutter during object shape detection in images. However, object detection based on DPMs requires additional computational time whenever part models are added to each viewpoint class. Similar to DPMs, [Brox et al 2011] used a poselet-based detector that characterizes object parts rather than global objects. In addition, their detector overcomes the shift and deformation issues that have affected view-independent object detection by non-rigidly aligning each poselet activation to the corresponding edge structures in the image.

[Russakovsky et al 2012] proposed an object-centric spatial pooling approach (OCP) for determining the location of an object of interest, which can be useful for image classification. OCP is used to infer the locations of objects, and this location information is then used to pool the foreground and background features. However, these methods need to extend their models to deal with the continuous viewpoint estimation problem. Recent studies have attempted to detect view independent objects using 3D object models. [Glasner et al 2011] incorporated a category-level detection and viewpoint estimation method for rigid 3D objects from single 2D images. This research uses the voting method for efficient accumulation of evidence, and combines a re- scoring and refinement mechanism using an ensemble of view-specific SVMs. However, this method considerably has only been used for experiments on rigid car data.

[Zia et al 2011] proposed a method for representing object models with more geometric detail than that provided by previous object class detectors, such as local shape features, discriminative part detectors, and efficient techniques of approximate probabilistic inference. Geometric 3D reasoning has received renewed attention recently, in the context of visual scene understanding. The level of geometric detail, however, is typically limited to qualitative or coarse-grained quantitative representations. This is linked to the fact that today's object class detectors are tuned towards robust 2D matching rather than accurate 3D pose estimation, encouraged by 2D bounding box-based benchmarks such as Pascal VOC. In this paper, we therefore revisit ideas from the early days of computer vision, namely, 3D geometric object class representations for recognition. These representations can recover geometrically far more accurate object hypotheses than just 2D bounding boxes, including relative 3D positions of object parts. In combination with recent robust techniques for shape description and inference, our approach outperforms state-of-the-art results in 3D pose estimation, while at the same time improving 2D localization. In a series of experiments, we analyze our approach in detail, and demonstrate novel applications enabled by our geometric object class representation, such as fine-grained categorization of cars according to their 3D geometry and ultra-wide baseline matching. The authors then proved that this geometric richness is a meaningful ingredient for accurate geometric scene-level reasoning.

[Pepik et al 2012] extended discriminatively trained deformable part models to include estimates of both the viewpoints, and the 3D parts that are consistent across the viewpoints. [Fidler et al 2012] introduced a method for localizing objects into three dimensions by enclosing them within tightly oriented 3D bounding boxes. This model represents an object class as a deformable 3D cuboid composed of faces and parts that are both allowed to deform with respect to their anchors on the 3D box. This paper addresses the problem of category-level 3D object detection. Given a monocular image, our aim is to localize the objects in 3D by enclosing them with tight oriented 3D bounding boxes. We propose a novel approach that extends the well-acclaimed deformable part-based model to reason in 3D. This model represents an object class as a deformable 3D cuboid composed of faces and parts, which are both allowed to deform with respect to their anchors on the 3D box. It is modeled the appearance of each face in front parallel coordinates, thus effectively factoring out the appearance variation induced by viewpoint.

[Hejrati and Ramanan 2012] presented a two-stage model focusing on an application for finding and analyzing cars. In the first stage, their research describes a compositional representation that models a large number of effective views and shapes using a small number of local view- based templates. These estimates are then refined in the second stage using an explicit 3D model of the shape and viewpoint. Even though these methods efficiently detect view-independent objects with regular patterns of independent 2D views, i.e., a car, they may provide a false detection when objects have irregular patterns in independent 2D views, i.e., an animal, sofa, or boat.

In the above approach to detecting and analyzing the 3D configuration of objects in real-world images with heavy occlusion and clutter. It is focused on the application of finding and analyzing cars. The first stage reasons about 2D shape and appearance variation due to within-class variation (station wagons look different than sedans) and changes in viewpoint. Rather than using a view-based model, it is described a compositional representation that models a large number of effective views and shapes using a small number of local view-based templates. It is used this model to propose candidate detections and 2D estimates of shape. These estimates are then refined by our second stage, using an explicit 3D model of shape and viewpoint. We use a morphable model to capture 3D within-class variation, and use a weak-perspective camera model to capture viewpoint.

Another issue is the computational runtime required for view-independent object detection. Traditional approaches require the application of a large number of different classifiers to an image at multiple locations and scales. Therefore, the computational complexity, as well as the training and testing time, increases linearly with the number of classes requiring detection because separate classifiers are trained and applied independently . To reduce the runtime complexity, [Chang et al2009] used shared features to reduce the number of classifiers. This method is used to detect cars in different views and sizes with a wide variety of backgrounds. In addition, the authors used an integral histogram to speed up the calculation of the HOGs.

[Razavi et al 2010] used an extension of Hough-based object detection to handle multiple viewpoints, and built a shared codebook by jointly considering different viewpoints.

Sharing features across views is a better use of training data, increasing the efficiency of training and detection. Hough transform based object detectors learn a mapping from the image domain to a Hough voting space. Within this space, object hypotheses are formed by local maxima. The votes contributing to a hypothesis are called support. In this work, we investigate the use of the support and its back projection to the image domain for multi-view object detection. To this end, we create a shared codebook with training and matching complexities independent of the number of quantized views. It is shown that since back projection encodes enough information about the viewpoint all views can be handled together. In these experiments, It is demonstrated that superior accuracy and efficiency can be achieved in comparison to the popular one versus the rest detectors by treating views jointly especially with few training examples and no view annotations.

[Torralba et al.2007] presented a multi-task learning procedure based on boosted decision stumps, which reduces the computational and sample complexity by finding common features that can be shared across classes (and/or views). The detectors used for each class are trained jointly, rather than independently. [Tosato et al 2010] modeled a human as a hierarchy of fixed overlapping parts, and each part was trained using a boosted classifier learned using Log boost on Riemannian manifolds. new algorithm to detect humans in still

images utilizing covariance matrices as object descriptors. Since these descriptors do not lie on a vector space, well known machine learning techniques are not adequate to learn the classifiers. The space of d-dimensional nonsingular covariance matrices can be represented as a connected Riemannian manifold. A novel approach for classifying points lying on a Riemannian manifold by incorporating the a priori information about the geometry of the space.

[Velaldi et al 2009] proposed a three-stage classifier, which combines linear, quasi-linear, and non-linear kernel SVMs for object detection. The objective is to obtain a state-of-the art object category detector by employing a state-of-the-art image classifier to search for the object in all possible image sub windows. Multiple kernel learning of Varma and Ray (ICCV 2007) is to learn an optimal combination of exponential $\chi 2$ kernels, each of which captures a different feature channel. these features include the distribution of edges, dense and sparse visual words, and feature descriptors at different levels of spatial organization. Such a powerful classifier cannot be tested on all image sub-windows in a reasonable amount of time. Thus it is proposed a novel three-stage classifier, which combines linear, quasi-linear, and non-linear kernel SVMs. It is shown that increasing the non-linearity of the kernels increases their discriminative power, at the cost of an increased computational complexity. Our contributions include

(i) showing that a linear classifier can be evaluated with a complexity proportional to the number of sub-windows (independent of the sub-window area and descriptor dimension);

(ii) a comparison of three efficient methods of proposing candidate regions (including the jumping window classifier of Chum and Zisserman (CVPR 2007) based on proposing windows from scale invariant features); and

(iii) introducing overlap-recall curves as a mean to compare and optimize the performance of the intermediate pipeline stages.

[Leibe et al 2008] introduced an object detection method that learns the appearance and spatial structure of a visual object category to recognize view-independent objects in that category, localizes them in cluttered real-world scenes, and segments them automatically from the background. However, these methods do not consider the deformation of objects. This paper presents a novel method for detecting and localizing objects of a visual category in cluttered real-world scenes. Our approach considers object categorization and figure ground segmentation as two interleaved processes that closely collaborate towards a common goal. the tight coupling between those two processes allows them to benefit from each other and improve the combined performance.

Conventional object detection methods recognize objects by considering full body and single degree viewpoints using colour histograms , histograms of oriented gradients (HOGs), and shape features. However, these methods have several limitations in a cluttered background with partial occlusions; therefore, part-based models and based approaches have been proposed. Part- based models deliver a better object detection. performance than a single model. In this type of model, objects are represented as a collection of parts arranged in a deformable configuration, and the part-based models can capture significant variations in appearance. In research on part-based models, different parts are used to capture the local-appearance properties of an object, whereas the deformable configuration of the object is characterized by spring-like connections between specific pairs of parts. In contrast, a single (global) model is often insufficiently expressive to represent a rich object category.

Object detection methods based on part-based models have the following advantages

(i). Part models are intrinsically robust against partial inter-object occlusions.

(ii).Part models exclude most of the background in the detection window, which avoids any confusion caused by changes in the background.

(iii). Although the offline learning of part models requires a specific amount of computational time and training samples to extract discriminative information, object detection can be performed in real time using the trained part-based models.

Hough transform based object detectors learn a mapping from the image domain to a Hough voting space. Within this space, object hypotheses are formed by local maxima. The votes contributing to a hypothesis are called support. In this work, we investigate the use of the support and its back projection to the image domain for multi-view object detection. To this end, we create a shared codebook with training and matching complexities independent of the number of quantized views. It is shown that since back projection encodes enough information about the viewpoint all views can be handled together.

[Torralba et al 2010] presented a multi-task learning procedure based on boosted decision stumps, which reduces the computational and sample complexity by finding common features that can be shared across classes (and/or views). The detectors used for each class are trained jointly, rather than independently.

[Tosato et al 2010] modeled a human as a hierarchy of fixed overlapping parts, and each part was trained using a boosted classifier learned using Log boost on Riemannian manifolds. new algorithm to detect humans in still images utilizing covariance matrices as object descriptors. Since these descriptors do not lie on a vector space, well known machine learning techniques are not adequate to learn the classifiers. The space of d-dimensional nonsingular covariance matrices can be represented as a connected Riemannian manifold. A novel approach for classifying points lying on a Riemannian manifold by incorporating the a priori information about the geometry of the space.

[Velaldi et al 2009] proposed a three-stage classifier, which combines linear, quasi-linear, and non-linear kernel SVMs for object detection. The objective is to obtain a state-of-the art object category detector by employing a state-of-the-art image classifier to search for the object in all possible image sub windows. Multiple kernel learning of Varma and Ray (ICCV 2007) is to learn an optimal combination of exponential $\chi 2$ kernels, each of which captures a different feature channel. these features include the distribution of edges, dense and sparse visual words, and feature descriptors at different levels of spatial organization. Such a powerful classifier cannot be tested on all image sub-windows in a reasonable amount of time. Thus we propose a novel three-stage classifier, which combines linear, quasi-linear, and non-linear kernel SVMs.

CHAPTER-3

View Independent Object Detection Using Shared Local Features

3.1.Introduction:

The main aim of this thesis is to detect the location of view independent object. Till now different methods have been proposed for object detection but the main problems found are view point, occlusion and computational time. By using the proposed method we can reduce computational time in a greater extent. And we train the classifier by taking different types of images in order to classify view independent object detection. Finally we achieve view independent object detection with less computational time.

The first step is to find shared local features by using exhaustive greedy selection method. The total training images are divided into learning group and testing group. First select one image and divide the whole image into some random patches. Then extract local features by using OCS-LBP descriptors. Apply these features to the random forest classifier in order to classify these features. Like wise apply same procedure for different view images and train using random forest classifier. Now find OCS-LBP descriptors for each test image and test by using random forest classifier. After that select top features from each view. Then compare each view features with other view features and select those features that overlap with other view features. These are called shared local features.

After finding shared local features from different view point categorise the features in such a way that patches having same location and size and train these patches by using random forest classifier . Like wise we get different categories and trained using random forest classifier. Finally apply a test image to these shared local classifiers and it will detect view and location of object. The purpose of using random forest classifier is for high dimensionality and more no of images. In these cases either SVM or AdaBoost classifiers are not able to provide good results. Finally we achieve object detection in different view point and less computational time.

3.2. Extracting local features

Local features are generally extracted from interesting points which are similar points in object in different images and these enables reduction in data. The interesting points are selected from highly textured parts and sudden changes in the object. There are so many methods to extract interesting points. The common methods used are SIFT [D.Lowe 2004], Harris interesting points [B.Leibe et al 2008] ,Local shape context [S.Belongie et al 2002].

Fig.3.2.1. interest points extracted from highly textured parts and those from abrupt changes.

Here local parts are not selected uniformly from overall image because the local parts are extracted from highly textured parts and sudden abrupt changes. From the above figure we can say that local parts are extracted from window frames and wheel so these are not that much distinguished parts from bikes and other objects.

Interest points are selected by selecting overlapping points [4, 5] which covers even smaller parts of the object. These interest points are fixed to significant locations and covers smaller parts of the object causes difficult to select interest points. Interest points can be extracted from the randomly generated sub windows which are normalised [33].

Fig.3.2.2. interest points are selected randomly from overall image

After selecting interest points randomly using selection window we have to find out local features by using Oriented centre symmetric local binary pattern (OCS-LBP) for each window which has been selected randomly. Here we can use either local binary pattern (LBP) or Centre symmetric local binary pattern (CS-LBP) . OCS-LBP provides better performance detection compared other methods.

3.2.1.Local binary pattern(LBP) :

Local binary pattern is type of feature extraction used in computer vision. It is texture based feature extraction proposed in 1990s. If we combine both HOG and local binary pattern then we get good performance results. Steps to find LBP

(i). Divide the window into cells (for example 8*8 cell dimension)

(ii). The pixels in each cell are compared with its neighbouring 8 pixels .

(iii). If the centre pixel has higher value than neighbouring pixel then assign ‘1’ otherwise ‘0’

(iv). Like wise compute histogram over the entire cell

(v). Normalise the window after calculating features

(vi). Finally combine all the features of cells and this gives the feature vector using LBP.

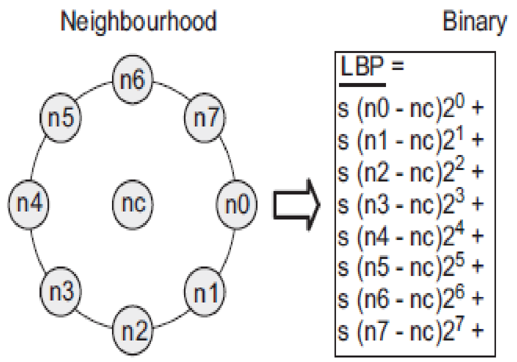

Fig. 3.2.1.1. Local binary pattern

3.2.2.Centre symmetric local binary pattern (CS-LBP) :

In LBP descriptors the histogram produced is large and becomes difficult to use in classification. So here we changed LBP[M.Heikkila et al 2009] descriptors slightly means instead of taking every pixel into account here we compare the pixels in the neighbourhood. The steps can follow

(i). Divide the window into cells (for example 8*8 cell dimension).

(ii). The pixel in each cell is has 8 neighbouring pixels .then compare the diagonally opposite pixels with respect to centre pixel

(iii). If the difference between these pixel values exceeds threshold then assign '1' otherwise '0'.

(iv). Like wise compute histogram over the entire cell.

(v). Normalise the window after calculating features

(vi). Finally combine all the features of cells and this gives the feature vector using LBP.

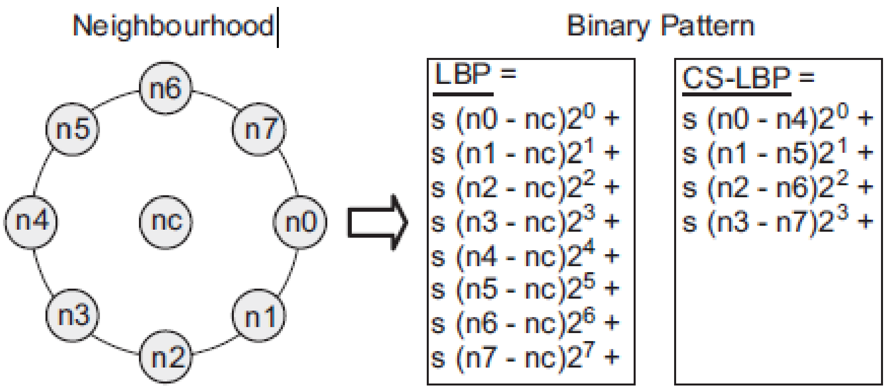

Fig.3.2.2.1. Centre symmetric local binary pattern.

In LBP pattern we can produce total 256 binary patterns where as in CS-LBP we can produce 16 binary pattern which has greater advantage reducing feature vector.

3.2.3.Oriented centre symmetric local binary pattern (OCS-LBP) :

In the CS-LBP the performance of object detection is less it does not magnitude information. So we use OCS-LBP in order to performance of object detection because it uses both orientation and magnitude information. The steps are as follows

(i). First divide the window into cells (for example 2*2 cell dimension).

(ii). The pixel in each cell is has 8 neighbouring pixels .then compare the diagonally opposite pixels with respect to centre pixel.

(iii). If the difference between these diagonally opposite pixels is above threshold then between those two pixels find out which one has highest pixel value.

(iv). Assign the difference value to the pixel which has highest magnitude and assign zero to the pixel which has lowest magnitude.

(v). Repeat the same process at each and every pixel and integrate from pixel to pixel by using integral histogram.

(vi). Repeat same process for all cells and find features.

(vii). Normalise the features by using min-max normalization.

(viii). Combine all the features from all cells and fprm a feature vector.

For example if there are two diagonally opposite pixels with respect to the centre pixel having intensities 80,50 and fix threshold value at 10. First calculate the difference in intensities between the pixels. So the difference is 30 which is greater than threshold then assign the difference value to the pixel value which has highest value in intensity and zero the pixel which has lower magnitude value. Here 80 is replaced by 30 and 50 is replaced by 0.

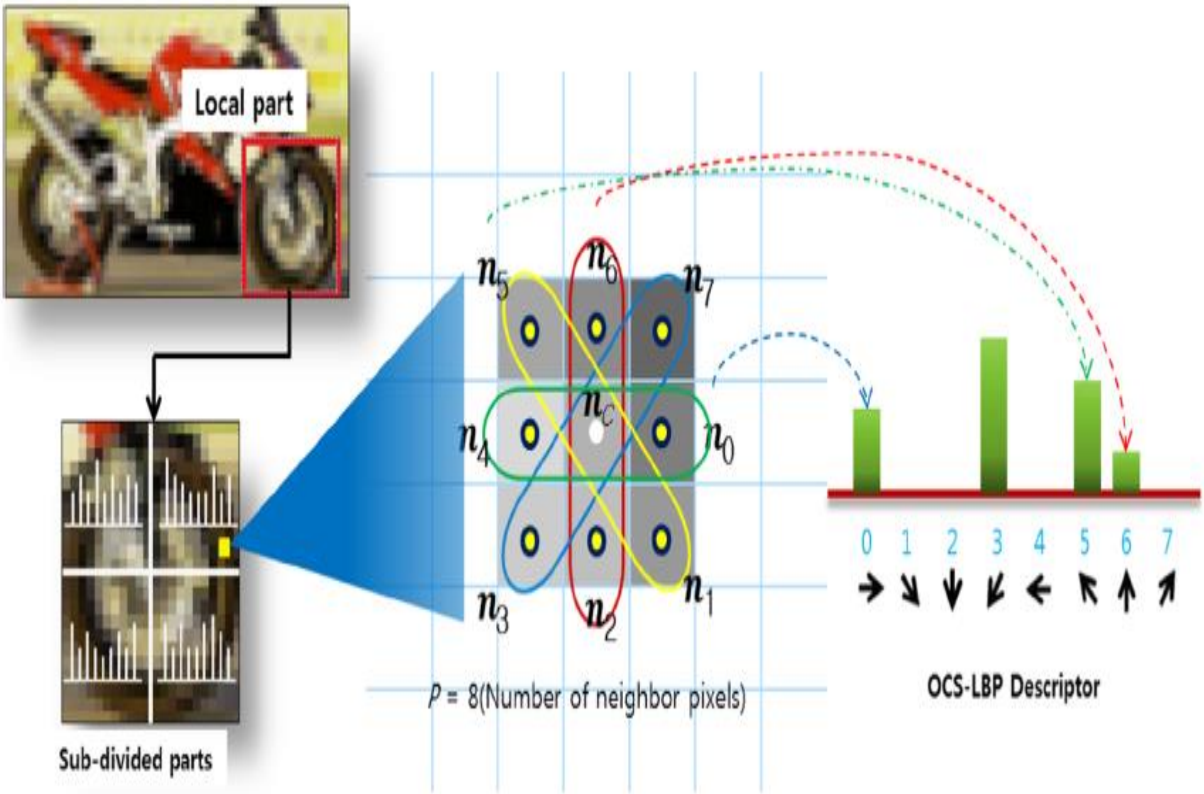

Fig. 3.2.3.1. Oriented centre symmetric local binary pattern (OCS-LBP) descriptors.

By using oriented centre symmetric local binary pattern (OCS-LBP) the performance of detection is increased compared to the CS-LBP. By using OCS-LBP descriptors the performance is increased by 7% compared to the CS-LBP. Because in OCS-LBP we consider both orientation as well as magnitude.

3.2.4. Histogram of oriented gradients:

Histogram of oriented gradients(HOG) [N.Dalal,B.Triggs, 2005] are features which are helpful in finding human detection. Now a days we use HOG’s for object detection also. In this project we used HOG features for object detection and used HOG’s as shared common features in different views.

The first step of calculating histogram is Gamma normalization . the purpose of normalization is to increase of performance and to become modest in performance. It applies to all colour models and improves the performance. The next step is calculating gradients. The purpose of calculation of gradients is segmentation . Here we used simple segmentation method and computes gradients. We know so many masking techniques and out of these masking techniques we used 1 dimensional techniques like [-1 0 1] are used in both x and y directions. We calculate magnitude and gradients at each pixel and divide angles in the ranges from 0 to 360 and store the sum of total magnitudes and store in bins and finally these become the features.

In order to get good classification the features are to be selected properly. For that we divide image into parts here we divide the image 3*3 blocks. For each block calculate the HOG features and store in bins and finally store all bins in one feature vector. In one block first we do gradients and then calculation of magnitude and angle at each pixel and finally the calculate the orientation of histograms at each pixel. Like wise apply same procedure for each block. Here we apply the normalization after finding the feature vector.

3.3.Random Forest classifier:

Before explaining random forest classifier any classifier has to follow certain conditions to decide which algorithm gives better results.

(i). How many no of training examples.

(ii). Feature dimensionality.

(iii). Does the classifier separate linearly?

(iv). Are features independent?

(v). Are features are linearly dependent on the target variable?

(vi). What are the requirements of the system in terms of speed/performance/memory storage?

(vii). Over fitting problem.

Before considering the classifier to classify we the problems are to be considered. In the thesis random forest classifier is used because it has some advantages compared to other classifier. Suppose we are going to train 100000 features for classification SVM classification will be difficult and computational time increases. So, in this case Random Forest classifier is best suitable in both saving computational time and better classification.

SVMs are not best suited for features having high dimensionality and if no of images are more. AdaBooost classifier is also not best suited because its performance depends on the weak classifier. Considering all the above problems we decided to use Random Forest classifier instead of using Adaboost and SVM classifier if there are more no of images to classify.

Random Forest classifier [B.C.Ko et al 2013] is the combination of prediction of trees such that each tree is randomly sampled but with same distribution. It is also defined as the ensemble classifier used either for classification or for regression or for any other purpose by constructing decision trees in the training part and output is either for mode of classification or prediction of regression or for any other purpose.

This algorithm was developed by Leo Breiman and Adele cutler. Breiman introduced "Bag of words "concept and random selection of features. Ho and Amit and Gemen introduced construction of decision trees with controlled variance. Selection of random selection trees is just like random selection window [B.C.Ko,2009]. The steps to find Random Forest classifier

(i). Decision tree learning.

(ii). Tree bagging.

(iii). Tree bagging to Random forest.

(iv). Extensions.

3.3.1.Decision tree learning:

Decision tree is a popular method used in machine learning techniques. Each tree grows in a randomised manner and just like the binary tree which many nodes and two branches at each node. Here, tree is formed by measuring the information gain which is obtained from entropy . Given training data is divided randomly by the splitting process which is calculated from information gain at each splitting process. The split which has highest information gain can be considered and divide into two parts and assign to each node. Like wise the process continues.

Each tree grows as follows

(i). If there are N no of training set then sample the training set at random with replacement from the original data. For each tree the training set is sample itself.

(ii). Select m no of input variables out of M input variables where m <<M and used for best split and m held constant during forest growing.

(iii). Each tree is grown to its largest possible extent without pruning.

For example there are three attributes like outlook, humidity and windy. The mode of boy to play at every attribute is given in table then tree formation will be like below figure

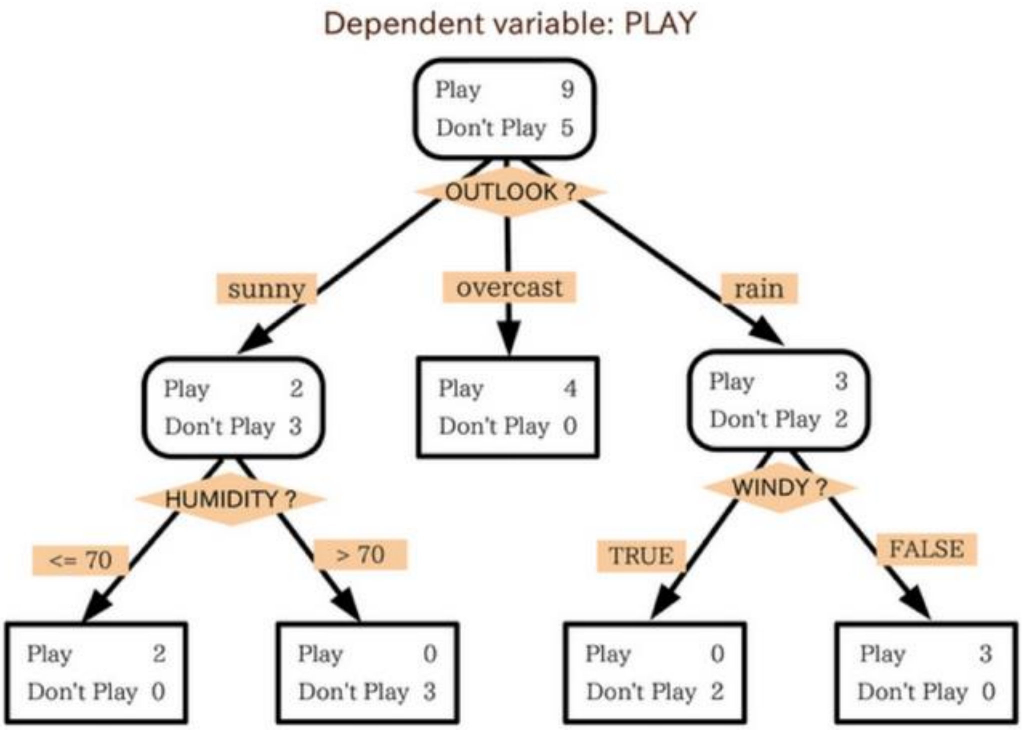

Fig. 3.3.1.1 . Tree formation

3.3.2.Tree bagging:

In the tree bagging the training set is given to the bootstrap aggregating. Let the training set be X= x_1,.,.,.,.,.x_n, and with responses Y= $y_{1,}$.,.,.,.,.,.,y_n bagging repeatedly and training set is sampled with replacement and assigns these samples to the trees such that they are fit.

For b = 1,.,., B

1. Sample n training examples from X,Y with replacement. These are denoted as X_b, Y_b.
2. Train decision or regression tree on X_b, Y_b.

After training prediction of samples can be done by averaging the prediction from all individual trees .

Without increasing the bias the variance of model can be decreased by using bootstrapping. Here noise is also reduced considerably until the trees are uncorrelated.

3.3.3.Bagging to Random Forest:

Bagging the Random Forest differ from trees only in one way that they use modified learning algorithm which is used for splitting. This process is called " Feature Bagging." Generally there is no need of doing bootstrapping if the trees are uncorrelated.

Suppose p are the features then $\sqrt{p}$ are used for splitting.

3.3.4.Extensions:

Random trees are generally randomised to yield random trees. These are trained like general procedure random samples and bootstrapping fro top-down splitting. Here there some terms to be mentioned. Those are entropy, information gain. These terms are mainly helpful while forming tree. In tree formation we have to use these terms for splitting.

3.3.5.Features of Random forest classifier:

1. It runs more efficiently when large no of data set is present.
2. It gives more accuracy among all classifiers.
3. It can classify any no of input variables without deletion.
4. It gives posterior probability by which we can estimate which variable is important.
5. It gives very less generalisation error compared to other classifiers.
6. When data is missing it gives missing data as well as accuracy of classification.
7. It has methods for balancing errors in class population unbalanced data sets.
8. Forest which are saved for future purpose on other data set.
9. In this algorithm we use prototypes which gives information between variables and classifiers.
10. It provides closeness between cases which are used in clustering, and gives interesting views of data.
11. It offers experimental results for providing variable interaction.

3.4. Shared features:

Generally local features are extracted from interest point and these local features are high which require computational time. So here we use shared local features which reduces the features considerably and decreases computational time. Shared local features[A.Torrlba et al 2007] means the features which are common in multi class objects. For example, if we want to extract shared local features from bike in different orientation. Here the bike is in any orientation the features like wheels front part are common in all orientations. So these features are used as shared features which is used to classify view independent object detection with reduction in computational time.

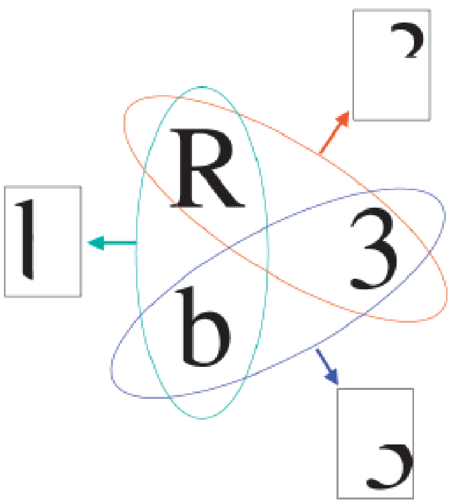

Fig.3.4.1. different objects having shared features

From the above figure we can say that objects R and b have common features shown in figure and objects R and 3 have common features and objects b,3 have common features. By using shared features we can reduce the complexity and computational time.

In this project the shared features can be extracted by using exhaustive greedy selection method. it can be explained as follows

1. First find random samples with random size and random locations in the image.
2. For each random sample find out OCS-LBP descriptors.
3. Train by using random forest classifier to train all the local features.
4. For the test image find out OCS-LBP descriptors.
5. Test these features by using random forest classifier.
6. Select top features which have high posterior probability.
7. Repeat the same process for the other images which are in different orientation.
8. Finally compare one local feature with other local feature and select features as shared features when overlapping between the features crosses 80% of total.

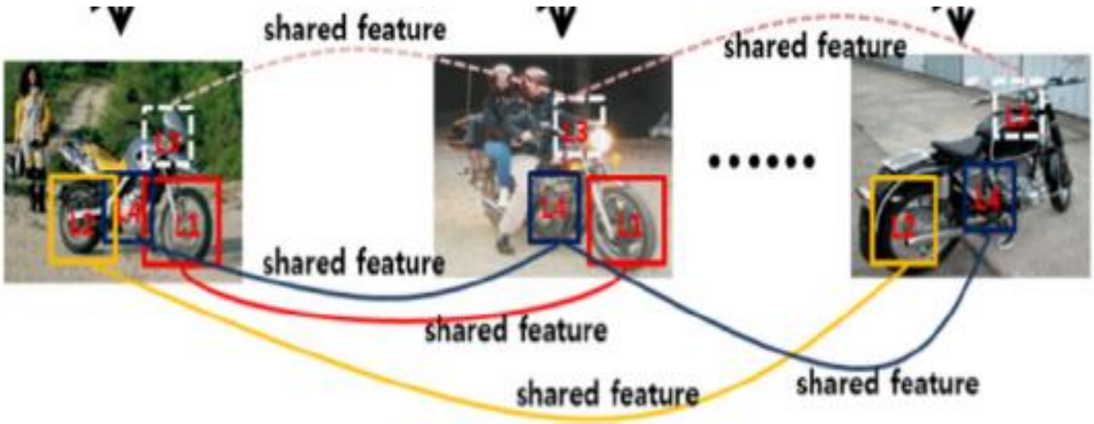

Fig .3.4.2. Extraction of shared local features

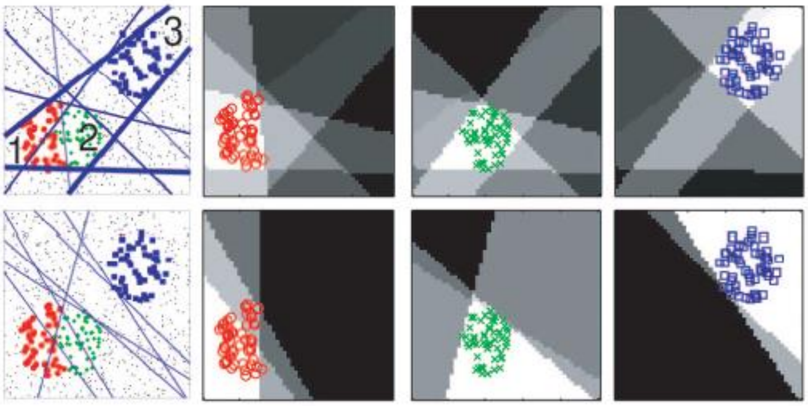

Fig 3.4.3. Illustration of feature sharing and independent features in which there are 3 object classes and one background class.

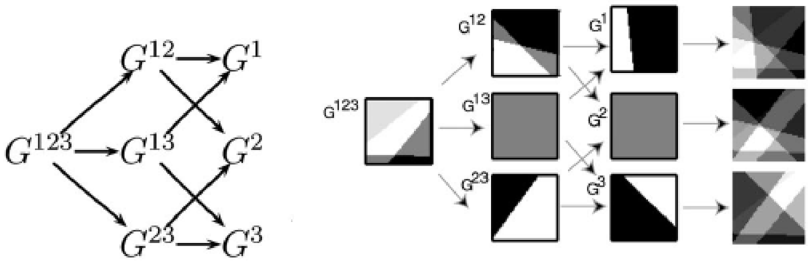

Fig.3.4.4. all possible ways to share features among all classifiers.

3.5.Algorithm description:

In order to find view independent object detection using shared local features we use total two methods.

1. Algorithm for extracting shared local features.
2. Object detection using shared classifiers

3.5.1.Algorithm for Extracting shared local features:

Exhaustive greedy selection method is used for finding shared local features. The steps are as follows . The training set is divided into learning group and testing group.

Step1:

Select k random seed samples with random size and random location for local features.

Step2:

For p=1 to P(no of views) do

// for all view classes repeat 3-8

Step3:

For i=1 to k do

// for all random samples repeat step 4

Step4:

Find out a set for each local feature L_i^p with same location and size from each random sample in the learning group for p-th view class.

Here L_i^p consists of centre position $(cx_{i,}cy_i)$, width and height (w_i, h_i), and view. It is represented as

$$L_i^p \quad = \ [cx_{i,}cy_i, w_i, h_i, p_i]^T$$

Determine OCS-LBP & HOG descriptors for each L_i^p

Step5:

Train each local set to classify k local features by using Random Forest classifier.

step6:

determine local feature L_i^p with same location and size in the learning group and find OCS-LBP descriptors for each local set.

Step7:

Test each local feature by giving as input to the trained Random Forest and compute posterior probability for k local features from output of Random Forest classifier.

Step8:

Select top M local features which have high posterior probability among k local features.

Step9:

From l=1 to M*P

Here local feature becomes shared feature when overlapping among P view classes exceeds threshold T

Spatial Overlapping is defined as overlap from one local feature to the other local feature is above 80%

$$\begin{cases} L_{m \ is \ SF} & if \ (\sum_{n=0}^{M*P} F(L_m \cap \ L_n)) > T \\ reject & else \end{cases}$$

3.5.2.Algorithm for object detection using shared classifiers:

Step1:

After selecting shared local features from different view points, some additional local features are also selected to separate required object from background, and different objects.

Step2:

Select local features which shares same location and size from different view points. And also select view images which shares same features. These are denoted as SLF-1,SLF-2,.....,

SLF-n.

Step3:

Train each SLFs by using Random Forest classifier and classify the shared features in each SLFs. These are called shared classifiers

Step4:

Given the test image, with same location and size by using selection window tested by using Random Forest classifiers. Each shared classifier generates posterior probability after testing.

Step5:

Find arithmetic average of each distribution p(c_i/s_i) for all shared classifiers S=(s1,s2,........,sn)

$$p_1(c_i/s_i) = 1/\mathrm{N} * \sum_{i=1}^{N} p(\frac{c_i}{s_i})$$

Where N is no of shared classifiers.

The view which has highest average posterior probability considers that view as determined view and detected can be found easily.

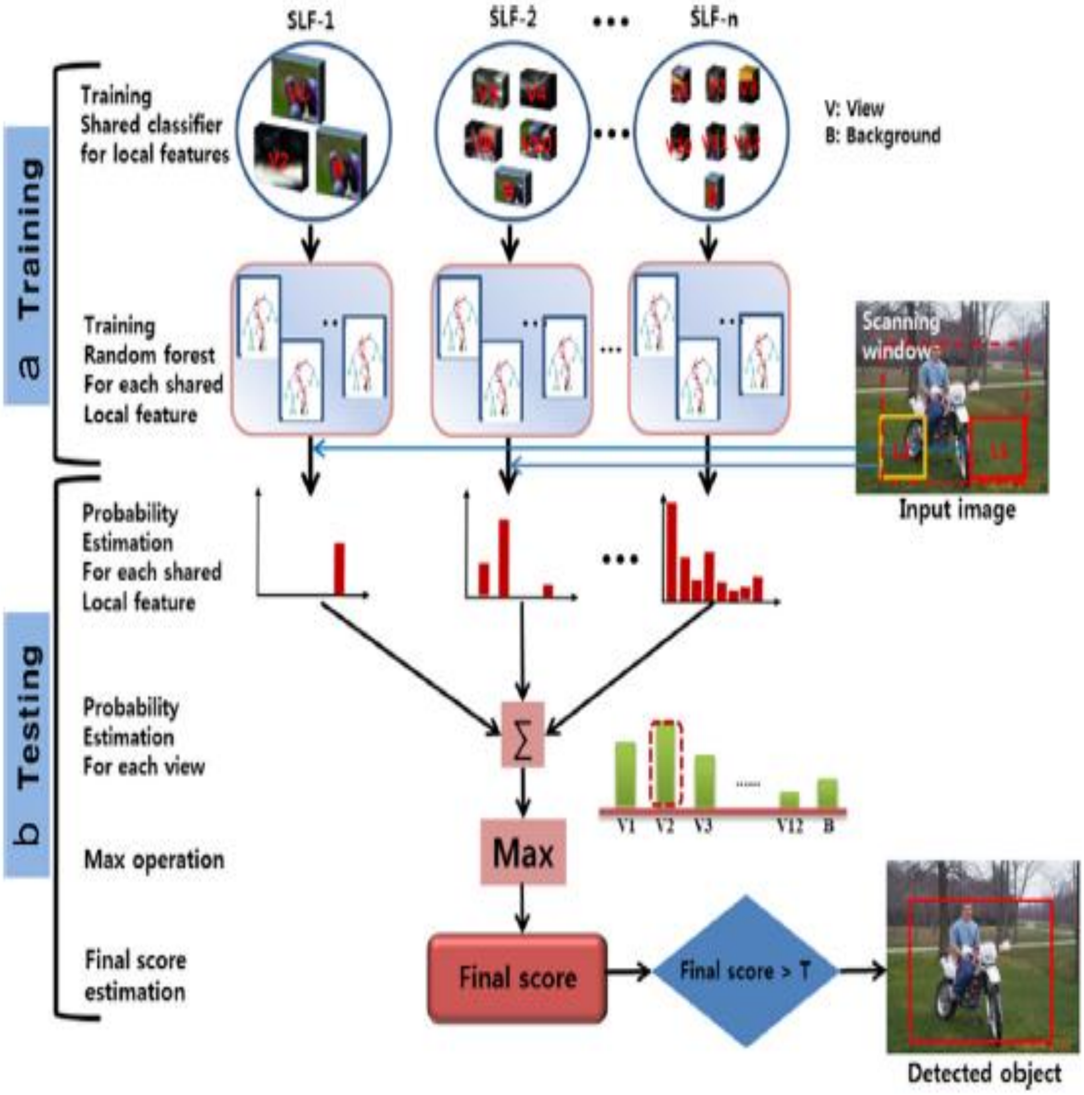

Fig. 3.5.2.1. Object detection using shared classifiers

CHAPTER-4

EXPERIMENTALWORK AND DISCUSSION:

We used PASCAL VOC 2012 dataset for object detection by using our proposed method. PASCAL VOC 2012 data set consists of 20 classes, the training set consists of so many images of different class in different views. Some of the classes have insufficient data and in some classes some images are missing. So we have finally taken 10 classes in different view points. Those are aeroplane, bicycle, boat, bus, car, dog, horse, motorbike, sofa, and train. These classes have multi view points which are helpful to us in detecting view independent object detection by using shared local features. During training we used 1200 objects in different views and also some negative images from background for each category. In this dataset the images are in random so we have to arrange images manually in all views.

During testing we don’t need to arrange manually. we resize the image into 80*40 and window size minimum is 10*10 and maximum is 40 *20. Here sliding window method is for object detection and with appropriate window is given to shared classifiers. Here we do segmentation to obtain dense samples and made a bounding box consist of smallest rectangular boxes.

we compared PASCAL VOC 2012 dataset with EPFL dataset which consists of car images in multiple views. We calculated precision and recall for estimation of object detection and also we compared our methods with other methods like DPM, colour, S-pooling, And-OR, M-kernel, S-search . our method has got good result compared to the other methods. And also we tested the performance with CS-LBP descriptors and with OCS-LBP descriptors. We got good performance by using OCS-LBP descriptors.

The object detection performance is measured by precision. Here precision is defined as ratio of true positive rectangles to the total no of rectangles used in the bounding box.

Precision for four different views:

Object	0-90	90-180	180-270	270-360	AP
Motor bike	66.82	63.73	73.50	63.3	66.83
Car	70.02	68.32	72.45	67.43	69.6
aeroplane	70.2	68.5	72.8	71.6	70.77
Bus	66.82	59.9	69.12	61.2	64.26
Horse	59.9	57.60	62.21	55.20	58.72

Table.4.1. precision for four different views by proposed method

Object	DPM	colour	s-pooling	And-OR	M-kernel	s-search	Proposed
Aeroplane	47.5	45.4	65.1	50.2	59.6	61.8	70.77
Car	44.2	44.8	44.9	42.8	59.3	65.5	69.6
motorbike	50.9	49.6	46.4	50	59.4	63	66.83
Bus	51.8	54.6	51	47.2	52.5	57.1	64.26
Horse	46.9	45.6	39.7	43.5	50.9	48.8	58.72

Table.4.2. Comparison of the average precision results using the proposed object detection method without considering viewpoint classification

Fig.4.1. Experimental detection windows of objects

CONCLUSION AND FUTURE WORK

In this project we have developed view independent object detection using shared local features which reduces the computational time considerably. And also it is robust to detect objects in the cluttered background, partially occluded objects. Here we used HOG descriptors which increases the performance than CS-LBP descriptors. The performance of HOG is more because it feature complexity is less.

Here we used exhaustive greedy selection method for extracting shared features which are common in different views. These shared features are trained as well as tested by using Random Forest classifier. Random Forest classifier is an ensemble classifier which ensembles decision trees and Random Forest increases training and testing speeds. This method is tested on PASCAL VOC 2012 which has different images of aeroplane, car, bike, motorbike etc. In the training the training images are divided into learning group/ testing group. The performance is better than other methods.

For future work, we plan to improve our algorithm to reduce the false detection when an object is occluded by other objects or deformed its shape. In particular, because pattern classifier is also important factor for increasing detection performance, we focus my research on finding suitable classifier in a number of latest classifiers such as random ferns and deep learning

REFERENCES

[1]. M. Swain, D.Ballard, Colour indexing ,Int.J.Comput.Vis.7(1)(1991) 11–32.

[2]. N.Dalal, B.Triggs, Histograms of oriented gradients for human detection, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition(CVPR), 2005,pp.886–893.

[3].S.Belongie, J.Malik, J.Puzicha, Shape matching and object recognition using shape context , IEEE Trans. Pattern Anal. Mach. Intell. 24(4)(2002)509–522.

[4].P.F.Felzenszwalb, R.B.Girshick, D.McAllester, D.Ramanan ,Object detection with discriminatively trained part based models, IEEE Trans. Pattern Anal .Mach. Intell. 32(9)(2010)1627–1645.

[5].D.Tosato, M.Farenzena, M.Cristani, V.Murino, Part-based human detection on Riemannian manifolds, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), 2010, pp. 3469–3472.

[6].G.Shu, A.Dehghan, O.Oreifej, E.Hand, M.Shah, Part-based multiple person tracking with partial occlusion handling, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012,pp.1815–1821.

[7].A.Thomas, V.Ferrari, B.Leibe, T.Tuytelaars ,B.Schiele, L.V.Gool, Towards multi-view object class detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006,pp.1589–1596.

[8]. B. Leibe, A. Leonardis, B. Schiele, Combined object categorization and segmentation with an implicit shape model, in: Proceedings ECCV Workshop on Statistical Learning in Computer Vision (ECCV Work- shops), 2004, pp.17–32.

[9]. I.Laptev , Improving object detection with boosted histograms, Image Vis. Computer. 27(5) (2009) 535–544.

[10]. A. Torralba, K. P. Murphy, W .T. Freeman, Sharing visual features for multi class and multi view object detection, IEEE Trans. Pattern Anal. Mach. Intell. 29(5) (2007)854–869.

[11]. B. Leibe, A.Leonardis, B.Schiele, Robust object detection with interleaved categorization and segmentation, Int. J. Comput. Vis. 7 (1-3) (2008)259–289.

[12].J.R.R.Uijlings, K. E. A. Sande, T. Gevers, A. W. M. Smeulders, Selective search for object recognition, Int. J. Comput.Vis.104(2) (2013) 154–171.

[13]. C. Gu, X. Ren, Discriminative Mixture of templates for view point classification, in :Proceedings European Conference on Computer vision (ECCV),2010,pp.408–421.

[14].R.J.López-Sastre, T.Tuytelaars, S.Savarese, Deformable part models revisited: a performance evaluation for object category pose estimation, in: Proceedings IEEE Conference on Computer Vision Workshops (ICCV Workshops),2011,pp.1–8.

[15].C.Gu, P.Arbeláez, Y.Lin, K.Yu, J.Malik, Multi component models for object detection, in: Proceedings European Conference on Computer Vision (ECCV), 2012,pp.445–458.

[16]. X. Wang, L. Lin, Dynamic and-or graph learning for object shape modeling and detection, in: Proceedings Annual Conference on Neural Information Processing Systems (NIPS), 2012, pp.1–9.

[17]. T. Brox, L. Bourdev, S. Maji, J. Malik, Object segmentation by alignment of poselet activations to image contours, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, pp. 2225–2232.

[18].F.S.Khan, R.M.Anwer, J.Weijer, A.D.Bagdanov, M.Vanrell, A.M.Lopez, Color attributes for object detection, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012,pp.3306–3313.

[19].O.Russakovsky, Y.Lin, K.Yu, L.Fei-Fei, Object centric spatial pooling for image classification, in: Proceedings European Conference on Computer Vision (ECCV), 2012,pp.1–15.

[20].D.Glasner, M.Galun, S.Alpert, R.Basri, G.Shakhnarovich, View point aware object detection and pose estimation, in: Proceedings IEEE Conference on Computer Vision (ICCV),2011,pp.1275– 1282.

[21].M.Z.Zia, M.Stark, B.Schiele, K.Schindler, Revisiting 3D geometric models for accurate object shape and pose, in: Proceedings IEEE Conference on Computer Vision workshops (ICCV Workshops), 2011, pp.569–576.

[22].B.Pepik, P.Gehler, M.Stark, B.Schiele, 3D\widehat2PM-3D deformable part models, in: Proceedings European Conference on Computer Vision (ECCV),2012,pp.356–370.

[23].S.Fidler, S.Dickinson, R.Urtasun, 3D object detection and view point estimation with a deformable 3D cuboid model, in: Proceedings Annual Conference on Neural Information Processing Systems (NIPS), 2012,pp.620–628.

[24].M.Hejrati, D.Ramanan, Analyzing 3D objects in cluttered images, in: Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS), 2012, pp.602–610

[25]. C.H.Chang, C.C.Wang, J.J.Lien, Multi-view vehicle detection using gentle boost with sharing HOG feature, in: Proceedings 22th IPPR Conference on Computer Vision, Graphics and Image Processing (IPPR), 2009,pp.1688–1694.

[26].N.Razavi, J.Gall, L.V.Gool, Back projection revisited: scalable multi view object detection and similarity metrics for detections, in: Proceedings European Conference on Computer Vision (ECCV), 2010,pp.620–633.

[27].B.C.Ko, J.Y.Kwak, J.Y.Nam, Human tracking in thermal images using adaptive particle filters with online random forest learning, Opt. Eng. 52(11) (2013)1–14.

[28].B.C.Ko, D.Y.Kim, J.H.Jung, J.Y.Nam, Three level cascade of random forests for rapid human detection, Opt. Eng. 52(2)(2013)1–11

[29].M.Heikkilä, M.Pietikäinen, C.Schmid, Description of interest regions with center symmetric local binary patterns, Pattern Recogition. 42(3) (2009)425–436.

[30].Y.Ma, L.Deng, X.Chen, N.Guo, Integrating orientation cue with EOH-OLBP-based multi level features for human detection, IEEE Trans. Circuit Syst. Video Technol. 23(10)(2013)1755–1766.

[31].F.Porikli, Integral histogram: a fast way to extract histograms in Catesian spaces, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005, pp.829–836.

[32].D.Lowe, Distinctive image features from scale invariant key points Int. J. Computer. Vis.60 (2) (2004)91–110.

[33].R.Marée, P.Geurts, J.Piater, L.Wehenkel, Random sub windows for robust image classification, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR),2005, pp.34–40.

[34]. B.C.Ko, S.H.Kim, J.Y.Nam, X-ray image classification using random forests with local wavelet based CS-Local Binary Patterns, J. Digital Imaging 24(6)(2001)1141–1151.

[35]. L. Breiman, Random forests, Mach. Learn. 45(1)(2001)5–32.

[36]. H.Byun, B.C.Ko, Robust face detection and tracking for real life applications, Int. J. Pattern Recognition. Artif.Intell.17(6)(2003).

[37]M.Everingham, L.V.Gool, C.K.I.Willams, J.Winn, A.Zisserman, the PASCAL Visual Object Classes (VOC) Challenge, Int. J. Computer.Vis. 88 (2) (2010)303–338.

[38]M.Ozyysal, V.Lepetit, P.Fua, Pose estimation for category specific multi view object localization, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, p p. 778–785.

[39]A.Velaldi, V.Gulshan, M.Varma, A.Zisserman, Multiple kernels for object detection ,in: Proceedings of IEEE Conference on Computer Vision (ICCV) 2009,pp.606–613.