UTILIZING MULTIPLE INSTANCE LEARNING FOR COMPUTER VISION TASKS

A THESIS

SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING

AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Fadime Şener

July, 2013

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Pınar Duygulu Şahin(Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Nazlı İkizler Cinbiş(Co-Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. İ. Aykut Erdem

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Öznur Taştan Okan

Approved for the Graduate School of Engineering and Science:

Prof. Dr. Levent Onural
Director of the Graduate School

ABSTRACT

UTILIZING MULTIPLE INSTANCE LEARNING FOR COMPUTER VISION TASKS

Fadime Şener
M.S. in Computer Engineering
Supervisor: Assist. Prof. Dr. Pınar Duygulu Şahin
Co-Supervisor: Assist. Prof. Dr. Nazlı İkizler Cinbiş
July, 2013

The Multiple Instance Learning (MIL) paradigm arises to be useful in many application domains, whereas it is particularly suitable for computer vision problems due to the difficulty of obtaining manual labeling. Multiple Instance Learning methods have large applicability to a variety of challenging learning problems in computer vision, including object recognition and detection, tracking, image classification, scene classification and more.

As opposed to working with single instances as in standard supervised learning, Multiple Instance Learning operates over bags of instances. A bag is labeled as positive if it is known to contain at least one positive instance; otherwise it is labeled as negative. The overall learning task is to learn a model for some concept using a training set that is formed of bags. A vital component of using Multiple Instance Learning in computer vision is its design for abstracting the visual problem to multi-instance representation, which involves determining what the bag is and what are the instances in the bag.

In this context, we consider three different computer vision problems and propose solutions for each of them via novel representations. The first problem is image retrieval and re-ranking; we propose a method that automatically constructs multiple candidate Multi-instance bags, which are likely to contain relevant images. The second problem we look into is recognizing actions from still images, where we extract several candidate object regions and approach the problem of identifying related objects from a weakly supervised point of view. Finally, we address the recognition of human interactions in videos within a MIL framework. In human interaction recognition, videos may be composed of frames

of different activities, and the task is to identify the interaction in spite of irrelevant activities that are scattered through the video. To overcome this problem, we use the idea of Multiple Instance Learning to tackle irrelevant actions in the whole video sequence classification. Each of the outlined problems are tested on benchmark datasets of the problems and compared with the state-of-the-art. The experimental results verify the advantages of the proposed MIL approaches to these vision problems.

Keywords: Computer vision, Multiple instance learning, Image retrieval, Image re-ranking, Action recognition in images, Multiple features, Interaction recognition .

ÖZET

BİLGİSAYARLI GÖRÜ PROBLEMLERİNİN ÇOKLU ÖRNEKLE ÖĞRENME İLE DEĞERLENDİRİLMESİ

Fadime Şener
Bilgisayar Mühendisliği, Yüksek Lisans
Tez Yöneticisi: Assist. Prof. Dr. Pınar Duygulu Şahin
Ortak Tez Yöneticisi: Assist. Prof. Dr. Nazlı İkizler Cinbiş
Temmuz, 2013

Çoklu örnekle öğrenme paradigmasının birçok uygulama alanında yararları görülmekte beraber bu öğrenme yöntemi etiketlemenin zor olduğu bilgisayarlı görü problemlerine özellikle uygundur. Çoklu örnekle öğrenmenin bilgisayarlı görüde nesne tanıma ve bulma, izleme sahne sınıflandırma, resim sınıflandırma vb. gibi birçok zorlu öğrenme problemlerine uygulamaları bulunmaktadır.

Geleneksel gözetimli öğrenmede teksel etiketlerin kullanılmasından farklı olarak, çoklu örnekle öğrenme örnek torbaları üzerinden çalışır. Bir torba eğer en az bir pozitif örnek içeriyorsa pozitif olarak etiketlenir diğer türlü torba pozitif örnek içermiyorsa negatif olarak etiketlenir. Çoklu örnekle öğrenmenin amacı torba olarak organize edilmiş eğitim verisini kullanarak bazı konseptler için bir model öğrenmektir. Bilgisayarlı görüye çoklu örnekle öğrenmeyi uygulayabilmenin önemli bir aşaması da görsel problemler için torba tanımının yapılması ve torbaların içindeki örneklerin ne olacağının belirlenmesidir.

Bu bağlamda üç farklı bilgisayarlı görü problemi ile çalışmakta ve özgün çözümlerimizi sunmaktayız. İlk olarak resim geri getirme ve sıralama problemine çalıştık ve ilgili resimleri içeren aday çoklu örnek torbalarını otomatik olarak oluşturduğumuz yöntemimizi sunduk. İkinci olarak resimlerden hareket tanıma problemine çalıştık. Resimlerden nesne içeren aday pencerelerin otomatik olarak çıkararak ile zayıf gözetimli bir yaklaşımla nesnelerin tanınması problemine araştırdık. Son olarak videolardan insan etkileşimlerini tanımayı bir çoklu örnekle öğrenme çatısı içerisinde amaçladık. İnsan etkileşimi tanımada videolar farklı aktiviteleri içeren hareketlerden oluşurlar ve amacımız video içerisine dağılmış olan bu ilgisiz aktivitelere rağmen etkileşimi tanımaya çalışmaktır. Bu problemi çözmek için, videolarda bulunan bu ilgisiz hareketleri ele alacak şekilde

çoklu örnekle öğrenme yöntemini kullandık. Bahsettiğimiz çalışmalarımızı veri kümeleri üzerinde test ettik ve en iyi çözümlerin sonuçları ile karşılatırdık. Veri kümeleri üzerindeki deneysel sonuçlarımız sunduğumuz algoritmaların performansını doğrulamaktadır.

Anahtar sözcükler: Bilgisayarlı görü, Çoklu örnekle öğrenme, Görüntü geri getirme, Görüntü sıralama, Görüntülerden hareket tanıma, Çoklu özniteliler, Etkileşim tanıma.

Acknowledgement

First of all, I would like to thank my advisors Assist. Prof. Dr. Pınar Duygulu Şahin and Assist. Prof. Dr. Nazlı İkizler Cinbiş for giving me a chance to work with them. Thanks for supporting and guiding me in this thesis. I owe a very important debt to Nazlı İkizler Cinbiş who made the biggest difference in my life. She has motivated me towards new possibilities in life for last two and a half years. I'm so lucky to have her be my advisor.

I want to thank the members of my thesis committee Assist. Prof. Dr. İ. Aykut Erdem and Assist. Prof. Dr. Öznur Taştan for accepting to review my thesis and their valuable comments. I would like to offer my special thanks to Dr. Erkut Erdem who introduced me with Vision World.

I would like to thank members of Bilkent RETINA research group especially Hande for always being kind and for beautiful dinner organizations, mostly shared times at Quick China. Thanks to Yiğit, Gökhan, Sermetcan, Eren, Caner, Ahmet, Anıl, İlker. Also I would like to thank members of Hacettepe University Computer Vision Laboratory especially Çağdaş, Aysun and Levent for their friendship.

I would also like to thank my dear cousin Burcu, we shared several years together and I have learnt much from her. Thanks to my friend Medine and her husband Şahin and I wish them a happy life together. I want to thank Seher for her support from İstanbul. I want to thank my childhood friend Adalet and my doctor at the same time. And thanks to Uğur who colored my last year with his lifestyle.

I deeply thank my parents Ali and Zehra Şener and my grandfather Cuma Şener for always believing in me, for their continuous love and support to my decisions. Thanks to my dear brothers Ender and Eren who I love the most.

Finally I would like to thank Nermin, for always being together. The good and bad times of last six years have been shared with her.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis introduces novel solutions for three computer vision problems using Multiple Instance Learning (MIL), which is a semi-supervised learning methodology proposed as an extension of the standard supervised learning by Dietterich et al. [1]. The MIL framework has been receiving much attention recently and has a large applicability to learning problems in computer vision such as object recognition and detection, tracking, image classification, scene classification and more.

Multiple instance learning algorithms operate in the case of incomplete knowledge about training instances and their corresponding labels and this is why they are called *semi-supervised*. In standard supervised learning, every instance in training set is represented by a feature vector and associated with a label, in binary case positive or negative. However, multiple instance learning operates over bags of instances, where each bag is composed of one or more instances, hence one or more feature vectors. In MIL framework, the labels are assigned to bags, instead of the instances, and for the associated labels there is an assumption; in binary classification, a bag is labeled as positive, if at least one of the instances within the bag is known to be positive, whereas it is labeled as negative, if all the instances are known to be negative. The aim is to predict the label of an unseen bag. This form of learning is referred as weakly supervised, since the labels for

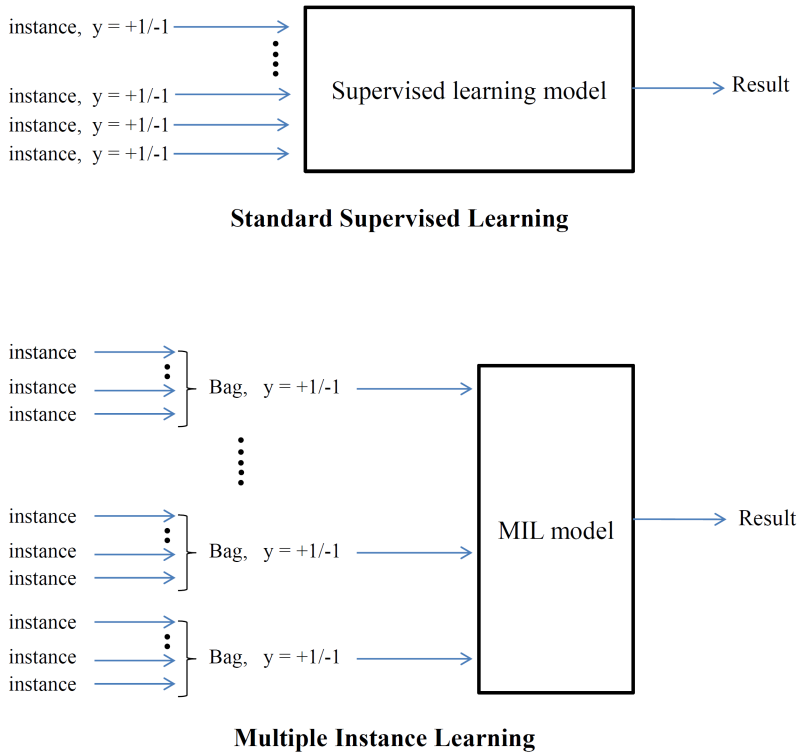


Figure 1.1: Standard supervised learning and multiple instance learning is illustrated. A representation based on a similar diagram by Dietterich et al. [1]

the individual instances are not available, and only the labels of the bags are provided. This difference between standard supervised learning and multi-instance learning is illustrated in Figure 1.1.

Although multiple instance learning structure is very suitable for many real world problems, it is not straightforward to adapt the solution to the visual problems in a multi-instance representation. There are two key issues to deal with; first is determining what the bag is and second, what will be the corresponding instances in the bags. In this context, we study three different computer vision problems, which are image retrieval and re-ranking, action recognition in still images and interaction recognition from videos. We discuss how we can apply multi-instance structure such that these problems benefit from MIL framework as much as possible. We describe successful procedures that effectively formulate the bag and instance formation for each of the aforementioned computer vision problems and evaluate them in detail.

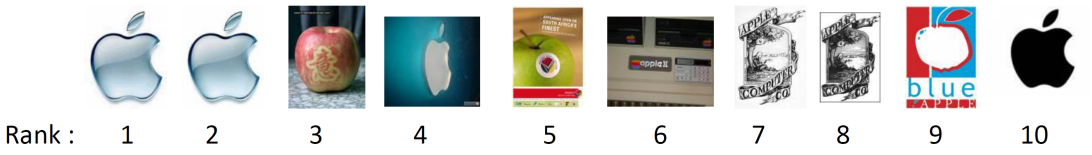


Figure 1.2: Ranked top 10 images results for "logo apple" query from "web-queries" dataset [2] .

1.1 Image Retrieval

The problem of re-ranking of images returned by text based search engines is the first problem that we tackle in the context of Multiple instance learning. In recent years, there has been an enormous increase in the amount of data stored on the Web and an important part of this data is images. Retrieving relevant images according to text-based queries has therefore been an important need. However, text-based image search may perform poorly since the returned results are seriously affected by different factors, such as irrelevant or incomplete text surrounding the images, polysemy or synonymy of textual descriptions, and so on. Since most of the current search engines (such as Google or Yahoo Image Search) make use of such surrounding textual data, the performance of image retrieval can be relatively lower than expected. In Figure 1.2, it is observed that for "logo apple" query, which should return "Apple Company Logo" images, results are affected by images of fruit "apple".

In order to increase the performance of such text-based image retrieval systems, approaches on visual re-ranking have been proposed in recent years. In visual re-ranking approaches, the idea is to explore the initial list of returned images by visual content analysis and propose a new ranking in which more relevant images are ranked higher. Such methods are also referred as relevance-based re-ranking methods [7].

We propose an approach to re-rank images returned by text-based search engines and improve image retrieval results, by building candidate bags that are utilized by multiple instance classifiers. Our proposed system is unsupervised, in the sense that, it does not need any explicit manual labeling of the images nor any

user feedback. The only input is a text query, and by evaluating the image data content retrieved by this query, our approach first automatically builds classifiers and then re-ranks the images based on the outputs of these classifiers.

The main idea of the proposed method is to automatically create "bags" that will be used with Multiple Instance Learning. In MIL, the classification is built upon bags as opposed to single instances. In this respect, Multiple Instance Learning framework is inherently suitable for retrieval problems, since in retrieval, the relevancy of the retrieved images is unknown. We claim that, by using the retrieved order of images, we can intelligently build the candidate bags for MIL framework, and then, the classifiers can learn the hidden patterns that are common to those images in these candidate bags. Consequently, based on these classifiers, the images can be re-ranked so that query-relevant images are ranked higher.

The bag generation step is the key point of the approach. We propose three different ways for building candidate bags, namely fixed-size, dynamic size and sliding window. We also evaluate the combinations of these three schemes. Based on the generated candidate bags, the algorithm consequently builds classifiers. Our algorithm operates on multiple sized candidate bags, and train classifiers using each of the constructed set of bags. We then use the ensemble of these classifiers and re-rank the images based on the responses from each of these classifiers.

We test our algorithm in Google [5] and Inria [2] datasets. The results show that by simply using multiple candidate bags and multiple instance learning in conjunction, our algorithm can perform on par with or better than the state-of-the-art.

1.2 Recognizing Actions From Still Images

Secondly we approach the problem of identifying related objects from a weakly supervised point of view and explore the effect of using Multiple Instance Learning

Figure 1.3: Three images from Stanford 40 Actions dataset [3] for “walking dog”, “brushing teeth” and “playing guitar” actions respectively.

for finding the candidate object regions and their corresponding effect in recognition. Recognizing actions in still images has recently gained attention in the vision community due to its large applicability to various domains. In news photographs, for example, it is especially important to understand what the people are doing from a retrieval point of view. Our approach does not use any explicit object detector, or part/attribute annotation during training. Instead, multiple object hypotheses are generated via Objectness Measure [6]. We then utilize a MIL classifier for learning the related object(s) amongst the noisy set of object region candidates

As opposed to motion and appearance in videos, still images convey the action information via the pose of the person and the surrounding object/scene context. Objects are especially important cues for identifying the type of the action. Previous studies verify this observation [8, 9, 10] and show that identification of objects play an important role in action recognition. Figure 1.3 shows example still images for actions ”walking dog”, “brushing teeth” and “playing guitar”. As it can be seen, “walking dog” action can be described with dog, an open scene and a standing human, “brushing teeth” action can be described with a bathroom background, teeth, and toothbrush, “playing guitar” action can be described with guitar and spatial pose of human body parts for playing guitar.

Besides the features extracted from candidate object regions, we evaluate various features that can be utilized for effective recognition of actions in still images. In our evaluation, we consider facial features in addition to features extracted within person regions and also features that describe the global image characteristics. We evaluate how much each proposed representation contribute

to the recognition of particular actions.

We test our method on the extensive Stanford 40 Actions dataset [3]. Our results show that the MIL framework over the candidate object hypotheses is quite successful and achieves better recognition performance compared to the state-of-the-art part and attributes based model of [3].

1.3 Interaction Recognition From Videos

Finally, we approach the problem of recognition of interactions between two-person from daily videos from a weakly supervised point of view. With the increase of cell phones and advanced camera hardware, there has been an enormous increase in the amount of videos. This rapid growth in the amount of data increases the need for video processing. The problem of recognition of human actions in videos is a major problem in computer vision community. It is especially important to understand recognition of human actions in many areas such as; security, robotics, video search, human-computer interaction, smart homes, child care, etc. However, action recognition is a highly difficult problem due to many problems such as; large number of actions, variability, inter-classes variability, intra-classes variability etc.

Recognition of individual actions is heavily studied in the literature of computer vision. Recently recognition complex non-periodic actions especially interactions has gained attention. Two-person interaction recognition has potential to create complex system applications from surveillance to human computer interfaces for content-based video retrieval. There has been little prior research in recognizing interaction between individuals when compared to single human action recognition. As a prior work, study of the Patron-Perez et al. [4] aims to recognize two-person interactions for four interactions in their newly proposed datasets; “hand shake”, “high five”, “hug” and “kiss” from video sequences of TV shows. Their work relies on detection of upper-body and estimation of head orientation. Figure 1.4 shows sample frames of a “hug” interaction video from

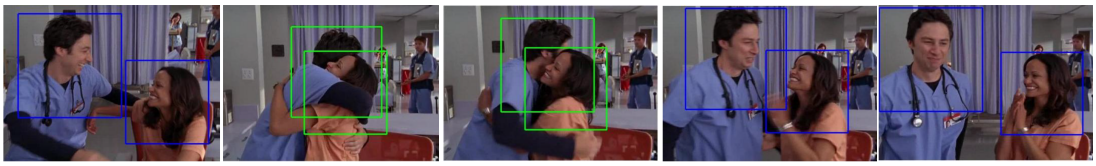


Figure 1.4: Several temporally aligned frames for a "Hug" interaction video from "TV Interactions" dataset [4]. Bounding boxes for face regions, blue for no-interaction, green for hug interaction

"TV Interactions" [4] dataset, for each frame upper-body detection of two-person is presented.

In our work, we aim to recognize two-person interaction in videos based on multiple instance learning in a weakly supervised way. With this purpose, in our proposed framework, we form each frame as a bag instance and each video as a bag. Our proposed system is unsupervised, in the sense that, we know the interaction class label of each video, which is a bag; however we do not have the information in which frame interaction is being processed. From sample frames shown in Figure 1.4, it can be observed that a video may start with a starting atomic action such as people walking toward to or a standing pose etc., then the interaction is processed, finally two-person move away from each other. Interaction and non-interaction frames in videos have not a certain order of processing such as non-interaction, interaction, non-interaction respectively. A video may start directly with the interaction and then two person may move away etc., In any case the frames do not include any interaction may effect recognition of original interaction in negative way. By using multiple instance learning, we aim to remove the negative contribution of these unimportant starting and ending atomic actions.

Besides, calculating descriptors step is another new point of our approach. Firstly the person regions are detected in each frame thus we reduce dimension by focusing on only person regions. We use the detected person regions in two types; first is face regions of two-person and the other is body regions of two-person which is an extended version of face region and it also covers person body parts. From these two type of regions we extract several visual feature descriptors to get shape information for example two-person stretch their arms to each other

for “handshake” interaction, to get motion information for example “highfive” is a relatively quicker interaction than “handshake” and to get information of spatial relation between two-person for example in “kiss” interaction two-person come more closer to each other than “handshake” interaction. Based on our definition of bag and instance, where each video frame is an instance and each video is a multi-instance bag, we define a frame as a relation between two-person. So our descriptor for an instance is a combination of two-person face and body region features. However this idea is not so easy to implement, since “TV Interactions” dataset [4] is a realistic one, so it may contain many people in a frame like in Figure 1.4 or has only one person and it may include different viewpoints of interactions. With the help of MIL framework we successfully overcome these problems of realistic videos.

We test our algorithm in “TV Interactions” dataset [4] dataset. Our results show that the MIL framework is quite successful and achieves better recognition performance compared to the state-of-the-art part [4].

The rest of this Thesis is organized as follows.

Chapter 2 consists of four parts. We first start with a brief introduction to Multiple Instance Learning together with overview of approaches in the literature. Then we review the related literature over three computer vision problems; image retrieval, recognizing actions from still images and interaction recognition from videos.

Chapter 3 describes our image retrieval and re-ranking approach. Firstly, the proposed approach of constructing bags for multiple instance classifiers is introduced, and then experimental evaluation is provided.

Chapter 4 describes our recognizing actions from still images approach. We start by presenting the various features utilized for recognizing actions in still images, especially the MIL approach for objects, and then we present the extensive evaluation of the features.

Chapter 5 describes our interaction recognition from videos approach. Firstly we start by presenting our features for interaction videos, and then we present the evaluation of the features and combinations of them.

Chapter 6 concludes the Thesis with a summary and discussions of the presented approaches with possible future directions.

Chapter 2

Background and Related Work

We study three different computer vision problems; image retrieval, recognizing actions from still images, interaction recognition from videos by using Multiple Instance Learning paradigm. In this chapter, we present a summary of the related studies over these subject.

2.1 Multiple Instance Learning (MIL)

Multiple instance learning(MIL) methods have large applicability to computer vision problems, especially to the cases where the annotation is expensive or difficult to obtain. This weakly supervised learning paradigm has been used in a wide range of applications, such as object recognition and detection [11, 12], tracking [13, 14], image classification [15, 16], scene classification [17] and more.

Multiple Instance Learning (MIL) is a variant of supervised learning which learn a concept given as bags of instances. As opposed to traditional supervised learning, where the learning procedure works over instances and their corresponding labels , multiple instance learning operates over bags of instances, where each bag is composed of multiple instances. This form of learning is referred as "semi-supervised" (or "weakly supervised"), since the labels for the individual instances

are not available, and only labels for the bags are given. The key assumption of MIL is a bag is labeled as positive, if at least one of the instances within the bag is known to be positive, whereas it is labeled as negative, if all the instances are known to be negative.

The multiple instance problem was introduced by Dietterich et al. [1] and used to solve the problem of drug activity prediction. A potency of a drug is determined by its binding degree with a target molecule and binding strength of a drug is determined by the shape of the drug molecules. However molecules may adopt many possible shapes by simply rotation of internal bonds. Binding degree of a drug may have many alternative low energy values with the change of molecular shape so it may adopt a set of alternative feature vectors and usually only one of the feature vectors represents the active molecular shape. A drug can be named as a bag and a bag with a set of feature vectors may have a label as "active" or "inactive". And the key assumption is valid through the solution where Dietterich et al. [1] named their algorithm as the axis-parallel rectangle (APR) method. In this algorithm they aim to solve the problem of finding an axis-parallel hyper-rectangle by expanding or shrinking a hyper-rectangle in the feature space which APR should contain maximum number of instances from all positive bags and minimum number of instances from negative bags. A drug is classified as "active" if at least one of its instances is inside of APR otherwise classified as "inactive".

The APR idea is extended to a probabilistic generative framework which is named as Diverse Density and was proposed by Maron and Perez [18]. They applied their solution for drug activity prediction problem, and also for two novel problems; stock market selection problem and learning a simple description of a person from images . The aim of the algorithm is to select a concept that is close to at least one instance to all positive bags and far from all negative bags which measure is named as Diverse Density. The desired concept is selected by maximizing Diverse Density measure. Zhang and Goldman [19] combined the expectation-maximization (EM) approach with Diverse Density and developed an algorithm called, EM-DD, to search for the optimal concept.

APR [1], DD [18] and EM-DD [19] are generative solutions for multiple instance learning and aim to identify the positive region in feature space which is close to all positive bags and far from all negative bags. APR [1] approach aims to find this space by using hyper-rectangles. DD [18] approach aims to define this space with the location of positive instances have high diverse density and the idea of DD [18] approach is improved by EM-DD [19] approach. Later Andrews et al. [20] described multiple intance problem in a discriminative way they proposed two novel algorithms called MI-SVM and mi-SVM where they aim to apply traditional supervised learning algorithms to multi-instance problems. For mi-SVM algorithm they modify Support Vector Machines for instance-level classification. In mi-SVM authors firstly convert the dataset of multi-instance problem, where a bag based description is present each bag has a label, to a traditional supervised learning dataset by assigning bag labels to instances of bags. Then a standard SVM is applied to this new dataset. All instances in positive bags are re-labeled using previously learned decision hyper plane and if the bag does not have any positive labeled instance, the instance that gives the maximum response to the decision function is labeled as positive. This re-labeling and training process continues until there is no label that changes. For MI-SVM they modify Support Vector Machines for bag-level classification. Firstly they initialize every bag by assigning the average of all instances' feature values in that bag, and then they learn a decision hyper plane with these single valued bags. This decision hyper plane is used to select a bag instance from each bag that gives the maximum response. The previous value of bag is replaced with the instance with maximum value. Finally with new valued bag dataset a new SVM is trained and re-labeling process continues until selected instances for bag representation do not represent any change.

Besides mi-SVM and MI-SVM algorithms mentioned above, many methods that use standard supervised learning techniques to solve multiple-instance problems have been proposed. Wang and Zucker [21] adopted the k-nearest neighbor algorithm for multi-instance problems by using modified version of Hausdorff distance between bags which is the maximum similarity of any two instances from

each bag. They presented two variant of the kNN algorithm named Bayesian-kNN and Citation-kNN. Andrews and Hofmann [22] developed a multiple instance learning method based on a generalization of linear programming boosting. [23], [24] applied neural networks and [25], [26] applied decision trees for multi-instance problems.

Graphical models also used in MIL problems. Deselaers and Ferrari [27] adopt the standard multi-instance assumption proposed by Dietterich et. al [1]. They define MIL with a conditional random field framework and name it MI-CRF. According to definition bags corresponds to nodes of their model and instances of a bag correspond to states of these nodes. Aim of the model is selection of the most positive instance in a bag same as done in MI-SVM. Leistner et. al [28] presents an algorithm for randomized trees called MIForests. Although the performance of these method outperform many MIL approach they have a stage of learning of complex graphical model as a result may be expensive algorithms.

As a different point of view to MIL problems Zhou et al. [29] shows that for some problems the relations among instances of a bag may convey important information. They proposed two new algorithms named as mi-Graph and MI-Graph where they model the relationships between the instances within a bag. They take every bag as a graph and each instance as a node in the graph. For MI-Graph they present a graph kernel between bags to distinguish the positive and negative bags and have a disadvantage which is the computational complexity with the growing number of edges. For mi-Graph they construct their graph by deriving affinity matrices and then present a graph kernel.

For standard MI assumption, if a bag contains at least one positive instance it is labeled as positive, otherwise if it does not contain any positive instance it is labeled as negative which is completely suitable for such as drug activity recognition problem. Because one of the molecules shapes is enough to say that drug has potency. However recent work on MIL focused on relaxed versions of this assumption because of the applicability of MIL problems to other domains and they may require alternative MIL assumptions. For some problems for example for image categorization negative bags may contain parts of positive category

instances. Foulds and Frank [30] examples this with the task of categorization "beach", "ocean" and "desert" images. If we define each image by segments we can define "ocean" with water segments, "deserts" with sand segments and "beaches" contain both water and sand segments. In binary case for this three class categorization task we use one-vs-all classification and it is clear that with the participation of "beach" category negative bags may contain some parts of positive instances. Chen and Wang [31] proposed a MIL framework called DD-SVM where a bag label is not determined by standard assumption, instead with some number of instances satisfies some properties. According to this firstly a collection of instances is determined by Diverse Density [18] function. These instances are more likely to appear in positive bags more than other bags. Then a nonlinear mapping is defined over these instances and every bag is embedded to a new feature space. Eventually DD-SVM converts multi-instance problem to a standard supervised learning problem. Additionally there are some other algorithms aim to convert multi-instance problems to standard supervised learning problems MILES [32], MILIS [33]. These algorithms' final aim is defining an embedding space for training stage however their difference comes from how they select the discriminative instances to construct embedding space. MILES has proposed by Chen et. al [32] and it does not make an instance selection in first stage. It gathers all instances in the bags as a vocabulary and defines a similarity between bags and instances in embedding space, then SVM is applied to new space and instance selection is done in this part. MILIS algorithm is proposed by Fu et al. [33] and propose efficient solution for selection of instances from positive and negative bags. They propose this selection over negative instances using kernel density estimator.

In this thesis we use Multiple Instance Learning with Instance Selection (MILES) algorithm proposed by Chen et. al [32]. MILES [32] algorithm works by embedding the original feature space x, to the instance domain $\mathbf{m}(B)$. Each bag is represented by its similarity to each of the instances in the dataset. The similarity between bag $\mathbf{B}_i$ and concept c_l is defined as

$$s(c_l, \mathbf{B}_i) = \max_j \exp\left(-\frac{D(x_{ij}, c_l)}{\sigma}\right), \tag{2.1}$$

where $D(x_{ij}, c_l)$ measures the distance between a concept instance c_l and a bag instance x_{ij} and σ is the bandwidth parameter. Any standard distance measure that is suitable for the feature space can be used for $D(x_{ij}, c_l)$. Then each bag can then be represented in terms of its similarities to each of these target concepts and this mapped representation $\mathbf{m}(B_i)$ can be written as

$$\mathbf{m}(B_i) = [s(c_1, B_i), s(c_2, B_i), \ldots, s(c_N, B_i)]^T. \tag{2.2}$$

This mapping step may bring many redundant or irrelevant features so authors apply 1-norm SVM and complete selecting of the most important features step while constructing classifiers simultaneously. Since in some cases classification of instances may be important, Chen et. al [32] also propose classifying instances in bags according to their contributions to the classification of related bag.

2.2 Image Retrieval

Image retrieval studies are focused around two main domains, namely content-based image retrieval and text-based image retrieval. Content-based retrieval relies on user provided query images, where visually similar images are searched, given a query image. An extensive survey on content-based image retrieval can be found in [34]. In text-based image retrieval problem, on the other hand, the user query is provided in terms of text, as opposed to query images. The aim is to generate a good ranking of the images based on their relevancy to the queried textual term(s). In this work, we focus on this text-based image retrieval problem and reranking for improving its results.

Image reranking has been a recent topic of interest. Tian and Tao [7] provide a recent and extensive review over the subject. Mainly, the proposed approaches so far differ in the type of features (such as textual, high-level visual and low-level visual features), and the type of learning method(such as clustering, classification, etc.) they utilize. In computer vision, visual reranking has also been used

for automatically collecting datasets that can further be helpful in recognition. Fergus et al. [5] was one of the initial efforts to use image search and reranking to automatically learn category descriptions and they adopt pLSA-based methods for this purpose. Fritz and Leibe [35] applied an LDA-based model for reranking problem. Graph-based models have also been explored. Hsu et al. [36] proposes a random walk based formulation over context graphs for reranking. In their influential work, Ying and Baluja [37] apply the famous PageRank algorithm to the visual content exploration of images. Recently, Liu et al. [38] proposes a reranking mechanism based on spectral filtering and graph based ranking.

Textual features has been explored in quite a number of studies for improving the image reranking [39, 40]. In [41], Shroff et al. used multimodal features such as text, metadata and visual features together to retrieve and rerank images and build an automatic reranking. Geng et al. [42] proposes a content-aware ranking system, in which visual cues are incorporated to the ranking learning process and jointly utilize the textual and visual features. In our approach, we do not make use of any textual cues, just use the initial ranking produced by the text query.

The work of Li et al. [43, 44] is the closest to our work, in the sense that they also apply Multiple instance learning to image reranking. Instead of relying on the initial text-based retrieval order, these works cluster the retrieved images and consider each cluster as a separate bag. In our algorithm, however, no prior clustering is needed, we dynamically construct bags based on the retrieval order and leverage multiple MIL classifiers. In the experimental section, we compare our method to Li et al.'s work.

2.3 Recognizing Actions From Still Images

Human action recognition has been an active research area for computer vision for a while. For an extensive review, the interested reader can refer to one of the recent surveys over the subject [45, 46] and the references therein. Most of the existing work focuses on action recognition in videos, which makes use of motion

cues and temporal information [47]. Action recognition in still images, however, is a more challenging problem, due to the lack of motion information and the difficulty of foreground subject segmentation.

In comparison to the large amount of work available for action recognition in videos, action recognition in still images is a less studied problem and is recently gaining attention. Wang, et al. [48] utilize deformable template matching for computing the distance between human poses and grouping similar poses. Thurau and Hlavac [49] use non-negative matrix factorization on pose primitives, where the pose primitives are learnt from non-cluttered videos and applied to images for finding the closest pose. In [50], the pose models are learnt from action images and those models are applied to classify actions in videos.

In more recent work, Yao and Fei Fei [51] have looked into the relationship between poses and objects and model the interactions using grouplet features. Object-person interactions are explored in other works such as [52, 8, 9, 53]. Delaitre et al. [54] has studied the use of bag-of-features and part-based representations using structural SVMs. Later on, Yao et al. [55] explore the use of random forests with discriminative decision trees. In their most recent work, Yao et al. [3] propose a part and attribute based model, which makes use of explicit object detectors for aiding action recognition in still images.

Prest et al. [10] also propose weakly supervised learning of human-object interactions. In [10], the objects having similar relative location with respect to the person are searched for the most recurring configuration for each action. For each image, their formulation is restricted to select one object window, whereas in our MIL approach, more than one object region can contribute to the recognition of the actions. Moreover, we do not enforce any spatial constraint for the objects and allow contributing object windows to come from any region of the image.

2.4 Interaction Recognition From Videos

There has been little prior research in recognizing interaction between individuals compared to single human action recognition. Datta et al. [56] have studied people on people violence in videos. Park and Aggarwal [57] proposed simultaneously segmentation and tracking multiple body parts of interacting humans in videos. Ryoo and Aggarwal [58] utilized spatio-temporal based methods, which is known have good performance on atomic and periodic actions, in their hierarchical framework proposed for human interaction recognition.

In [56], [57] , [58] interactions are studied in a hierarchical manner and heavily depend on low level processes such as background subtraction, body parts etc. On the other hand recently computer vision community focused on real world video data which obtained from TV shows, Youtube etc. and low level processes most probably may fail for such complex videos. In this context study of Patron-Perez et al. [4] is different. They aim to recognize two-person interactions such as hand-shake, high-five from video sequences which are extracted from TV shows. They address the problem of recognizing interactions between two people in videos and introduce a person-centered descriptor.

Patron-Perez et al. [4] claim that interaction most probably occurs around faces and face orientations contains important cues to understand the type of the action since two-person face to each other when they are in interaction. Faces and orientations are also considered as important cues in several studies. Fethi et al. [59] studied social interactions in egocentric videos considering faces and locations of faces.

Some other studies on interactions have wide scale viewpoints. [60] focus on group interactions instead of focusing on two individuals interactions by employing a structured SVM framework to capture structure of group activities. Gaidon et. al. [61] purpose a weakly supervised method for activity recognition and have good performance on human-human interaction problem.

Another different approach is studied in [62] where authors focus on how

people interact in still images. Their method is closely related to "Visual Phrase" approach [63]. In the same way they say that complex interactions can be modeled as a single representation rather than separate representation. They propose a joint model of body pose estimation by focusing personal space between people when they are interacting.

In our study we basically focus on the work of Patron-Perez el. al. [4] and we use Multiple Instance Learning. In contrast to instance based learning, multiple instance learning operates over bags of instances, where each bag is composed of multiple instances. There are several studies that use multiple instance learning for selection and categorize human actions. [64], [65]. Leung et. al. use [66] multiple instance learning for categorization of videos to handle noisy labels that comes from wrongly tagged videos. In [67] authors use multiple instance learning to simultaneously obtain segmentation labels and categorization in realistic video footage with clutter. Yun et. al. [68] also study on two-person interactions and use multiple instance learning. However their interaction dataset is depth and motion capture data. They extract 3D features and use to construct multiple instance learning bags.

Chapter 3

Ensemble of Multiple Instance Classifiers for Image Re-ranking

Text-based image retrieval may perform poorly due to the irrelevant and/or incomplete text surrounding the images in the web pages. In such situations, visual content of the images can be leveraged to improve the image ranking performance. In this study, we look into this problem of image re-ranking and propose a system that automatically constructs multiple candidate "multi-instance bags (MI-bags)", which are likely to contain relevant images. These automatically constructed bags are then utilized by ensembles of Multiple Instance Learning(MIL) classifiers and the images are re-ranked according to the final classification responses. Our method is unsupervised in the sense that, the only input to the system is the text query itself, without any user feedback or annotation. The experimental results demonstrate that constructing multiple instance bags based on the retrieval order and utilizing ensembles of MIL classifiers greatly enhance the retrieval performance, achieving on par or better results compared to the state-of-the-art.

The rest of the study is organized as follows: Section 3.1 introduces the proposed approach of constructing bags for multiple instance classifiers. Experimental evaluation is provided in Section 4.2.

3.1 Image Re-ranking with Ensemble of MIL Classifiers

We propose a system which automatically learns the queried textual concept by exploring the visual content of the noisy set of retrieved images and produces an improved ranking result. Our formulation is based on multiple instance classifiers, which treat the retrieved images as bags of positive instances. The formation of the *"multi-instance bags (MI-bags)"* is the key aspect of our algorithm. During this formation, we do not use any manual labeling of the retrieved images, but only assume that the retrieved set of images include some relevant images.

In this study, we propose a number of methods for constructing candidate bags, so that multiple-instance classifiers learned upon them form discriminative classifiers. These classifiers can then be used for image re-ranking and consequently improve image retrieval performance.

We first review multiple instance learning (MIL) paradigm and discuss why it is suitable for the problem of image re-ranking and categorization. Then, we present our approach on constructing MI-bags for MIL classification.

3.1.1 Overview of Multiple Instance Learning

In image retrieval, once the text query is input to a text-based image search engine, such as Google or Yahoo Image Search, a set of images is returned. These returned results are not always perfect, and most of the times, irrelevant images occur in higher ranks on the retrieved list. By analyzing the visual content of retrieved images, classifiers for the queried concept can be learned, and using these classifiers the relevant images can be ranked higher in an updated retrieval result.

Working on single image instances and building supervised classifiers using each image would require the availability of user feedback data or large scale annotation effort. When there is no such data available, which is the case with the

traditional text-based query system, the text-based retrieval order can provide an initial cue on the relevancy of the images to the queried concept. Text-based retrieval order is mostly formed using textual information surrounding the images, user click data, etc., and is likely to contain a certain number of in-class images. Based on this observation, we can assume that in-class images are returned throughout the retrieved list, although these in-class images can be ranked lower in the list or scattered throughout the list.

Since the exact labels for the class of the individual images are unknown, working over single images using supervised classification methods is not possible. However, if we assume that the in-class images are present throughout the list, we can form *"bags"* of the images and assume that each bag contains at least one positive example for the query. By this way, we can make utilize Multiple Instance Learning over bags of images.

Multiple instance learning is particularly suitable for our problem. Multiple candidate positive bags can be formed by using the text-based retrieval order of the images and thereon, multiple instance learning classifiers can be used to learn the queried concept.

A problem with the static and non-overlapping construction of the bags (as in [43]) is that the positivity assumption of the bags may not necessarily hold. From the nature of the image retrieval, we can assume that some of the bags contain positive images which are related to the queried concept. However, since we do not use explicit user feedback data, we do not know exactly which bags are indeed positive and which bags are negative in training. In order to deal with this issue, we generate multiple hypotheses for candidate bags from the ordered set of retrieved images and learn multiple MIL classifiers over each hypothesis. Our approach then combines multiple classifiers and re-ranks the images based on their classification scores.

3.1.2 Constructing Candidate Bags

Candidate bag generation is the key aspect of our approach. We evaluate different ways for constructing candidate multiple instance bags (MI-bags) which will be used in learning multiple instance classifiers. These different schemes are namely fixed-size bags, dynamic-size bags and sliding window approach. We now describe each of these approaches in detail.

3.1.2.1 Fixed-size Bag Construction

The simplest way to build candidate bags for employing multiple instance learning is to use fixed-size bags. In this approach, the initial list of images are divided into small subsets, i.e. bags, in which each bag contains k images. Then, these bags are utilized in MIL setting as positive instance bags.

More formally, given ranking R, the set of retrieved images is divided into equal k-sized bags, so that each bag contains k images based on R. In this construction phase, first k images that have ranks r_1 to r_k are assigned to bag B_1, images from r_{k+1} to r_{2k} are assigned to bag B_2 and so on. This procedure is illustrated in Figure 3.1.

In the experiments section, we present results with different k values, and see how the choice of k affects image retrieval performance. Since we do not have an explicit information on the positivity of the retrieved images, the best choice for k can be determined empirically. However this would require the availability of manually labeled set of images. In order to overcome this issue, we generate multiple candidate bags with varying k, and train classifiers using each of the constructed set of bags. Using the ensemble of these classifiers, we utilize the outputs of multiple candidate bags of varying sizes, thus bypass the selection of the optimal k value. This approach is further discussed in Sec. 3.1.3.1.

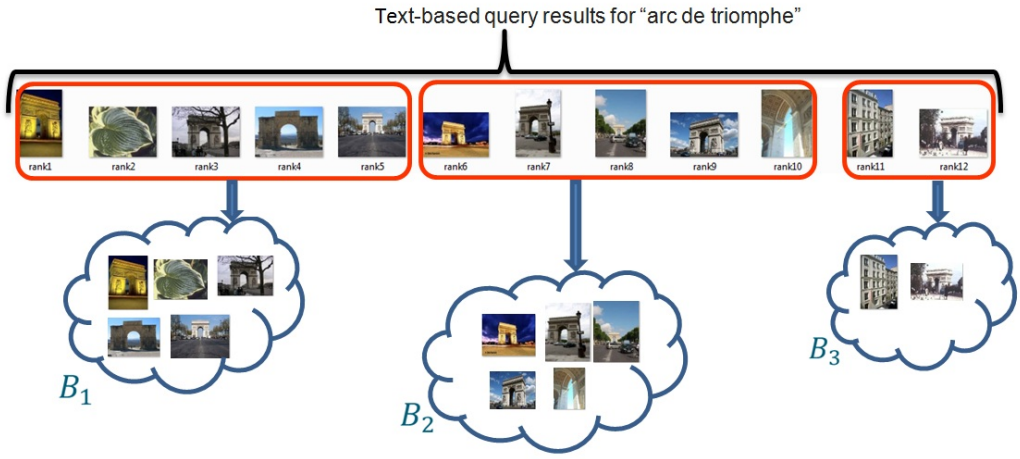


Figure 3.1: Formation of fixed-size bags from the retrieved images. In this example $k = 5$ images form individual instances of a single bag, based on the text-based retrieval order. These bags are then fed into multiple instance classifiers as positive bags.

3.1.2.2 Dynamic-Size Bag Construction

As discussed in the introduction, text-based search engines use surrounding text information accompanying images to retrieve relevant image data. While this text information is mostly noisy and incomplete, it can be seen as a initial point of reference for evaluating the images. In this context, we observe that, while the image search engine performance is far from perfect, the images returned earlier in search ranking, tend to be more relevant to the queried concept. Based on this observation, in order to increase the likelihood of each bag to contain an in-class image, we can form relatively smaller bags for the top ranks of the retrieved list and relatively larger bags from the lower ranks of the list. We call this procedure *“dynamic-size bags”*.

Assuming that the relevancy of the images decreases as the rank of the image increases, we can increase the bag size gradually at each γ interval of received images. More formally, given ranking $R = r_1 \dots r_N$, where N is the size of the image set, the set of retrieved images that have ranks r_1 to r_γ are divided into k-sized bags, images with ranks $r_{\gamma+1}$ to $r_{2\gamma}$ are divided into $(k + \sigma)$-sized bags, where k is the initial bag size, and σ is the amount of size increment. This

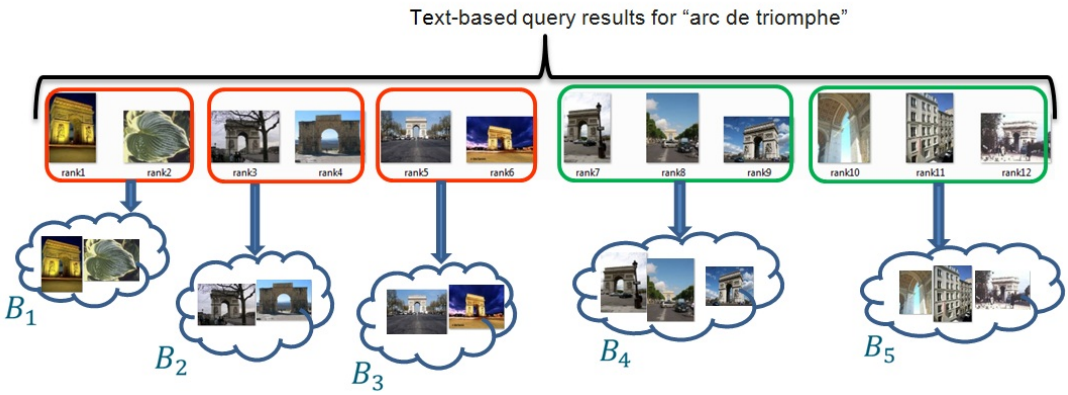


Figure 3.2: Formation of dynamic-size bags from the retrieved images. For the images that returned earlier in the list, smaller bags are formed, and for the images that return later in the list, larger bags are formed. In this example, the initial k is 2 and then, for the lower ranks of the text-based retrieval order k value is incremented by 1 and larger bags are formed.

procedure is illustrated in Fig. 3.2.

By this way, since the images returned later in text-based search ranking tend to be less relevant than the images returned earlier in the search, by increasing the bag size, the probability for each positive bag to include a positive instance is likely to be increased. In the experiments section, we evaluate how varying k,γ and σ affect the retrieval performance.

3.1.2.3 Sliding Window Bag Construction

Since the retrieved images do not have explicit labels, we cannot make sure that the candidate positive bags indeed include a positive instance for the MIL training. In order to deal with this issue, we can generate multiple overlapping bags. By following a sliding window approach, we can generate multiple bags, where at least a portion of these bags are assured to include positive instances. By dense sampling of bags in this way, we make sure that a large portion of the possible bag combinations are evaluated.

The sliding window procedure for building bags is shown in Fig. 3.3. This

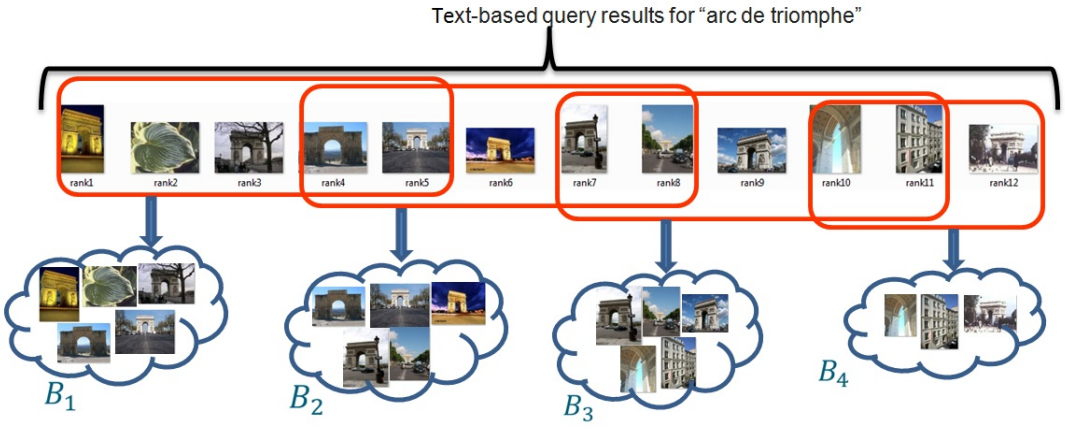


Figure 3.3: Sliding window approach for formation of fixed-size bags from the retrieved images. Here k is fixed ($k = 5$) and step size $M = M = ceil(k/2)$. Sliding window approach generates multiple overlapping bags and provides a dense sampling of the possible bag candidates for MI learning.

approach is analogous to the sliding window approach for object detection, where a window is slided over an image to search for particular occurrences of an object. In our context, by sliding a window over the sets of image instances, we consider each set of instances that falls within the same window as a candidate bag that will be used in MIL procedure.

More formally, given a ranking R of image set $I = \{i_1, \ldots, i_N\}$, starting from image ranked in R_1, we create a k-size bag where images from $R_1 \ldots R_k$ are assigned to B_1. At each sampling step, we increase the index by step size $M = ceil(k/2)$ and create a new bag, so that each new bag is composed of the images within retrieval rank $\{R_{(i-1+M)} \ldots R_{(i-1+M+k)}\}$.

Here, the bag construction can either be based on fixed-size bags, or dynamic-size bags, i.e., the sliding window can either have a fixed size, or increasing size based on the retrieval rank. We evaluate both of these approaches in detail in the experiments section.

3.1.2.4 Constructing Negative Bags

In order to use negative bag constraints of Multiple Instance Learning, it must be made sure that the constructed negative bags do not contain any positive instances. For this reason, while constructing negative bags, we use the images returned for queries other than the search query. We apply a similar scheme that sequentially forms the MI-bags based on the order of the images. However, it is possible that for non-relevant queries, some negative image pattern may emerge amongst the retrieved set for negative queries. In order to refrain from such a pattern, we first cluster the images returned for non-relevant queries by using k-means. Then, the cluster center order is randomized and the images are re-ordered based on the distances to these cluster centers. Then, this new order is used as the negative image set order. By this way, it is made sure that the order of images is randomized and the similar images are not scattered through the list of negative images, to avoid misleading patterns. Once the randomized list of negative images are established, we form fixed-sized bags over this negative image set.

3.1.3 Classification

Once the positive and negative bags are formed via one of the proposed schemes, Multiple Instance Learning algorithms can be applied using the constructed MI-bags. We now present the details of this classification stage.

Our MI-bag formation procedure is independent of the choice of the multiple instance classifier, therefore any multiple instance classifier can be used with our framework. In this study, we utilized Multiple Instance Learning with Instance Selection [32] (MILES) algorithm as the MI-classifier which discussed in Background and Related Work chapter. To measure the distance between a concept instance c_l and a bag instance x_{ij} any standard distance measure that is suitable for the feature space can be used. In our case, since all the features are histogram-based, we can use the χ^2 distance $D(x_{ij}, c_l) = \chi^2(x_{ij}, c_l) = \frac{1}{2}\sum_d \frac{(x_{ij}(d)-c_l(d))^2}{x_{ij}(d)+c_l(d)}$, where d is a feature dimension of the instance feature vector. We evaluate the

effect of choosing different distance functions in the experimental evaluation.

We use an SVM classifier over embedded representation provided by MILES. The original MILES formulation incorporates a L1-regularized linear SVM, which enforces some sparsity on the data. In our case, since the retrieval data can have multiple modes, we experience that using L2-regularized SVM is better suited for this purpose.

3.1.3.1 Ensemble of MIL Classifiers

While forming the positive bags for the MIL framework, the most crucial parameter is the bag size k. The optimal k depends mostly on the order of initial retrieval. Since our algorithm does not make use of any explicit user feedback or labeled data, determining the optimal k value is not possible.

In our empirical experiments, we have observed that the performance is largely dependent on the selection of k value. In order to deal with this issue, instead of learning a single classifier that operates over a single k value, we learn an ensemble of MI classifiers, each of which works on multiple bags formed using different k values. The final classification is decided by averaging the responses of all MI classifiers. By pooling the responses of all classifiers, we bypass the step of choosing the optimal k value and also by combining multiple classifier responses, a more reliable classifier could be achieved. In the experiments section, we evaluate both the effect of choosing different k values and the ensemble classifiers, respectively.

3.2 Experiments

In this section, we evaluate the proposed MI bag construction approach and ensemble classification.

3.2.1 Datasets

In order to evaluate the performance of our method, we use two benchmark datasets. First is the Fergus dataset [5] and the second is the Web Queries [2] dataset.

Fergus Google dataset [5] has been collected via text queries from the Google Image Search. This dataset consists of 7 categories (Airplane, Cars Rear, Face, Guitar, Leopard, Motorbike, and Wrist Watch) and each of these categories includes about 600 images on average. For each category, labeling is done with 0 = "Junk", 1 = "Intermediate" and 2 = "Good" for each image. On average there are 30% "Good" images without major occlusion, but no constraints on viewpoints, scaling and orientations, 20% "Intermediate" images have lower quality when compared to "Good" images, have extensive occlusion and image noise, and 50% "Junk" images that are irrelevant to the category. Following the test setup [5, 41], we consider "Intermediate" and "Good" images as "relevant to the query" but "Junk" as "irrelevant to the query".

Web Queries [2] is a recently compiled dataset, which includes 353 web image search queries. These queries are selected among the frequent terms submitted to image search engines. There are more than 200 images for 80% of the images, and the dataset has 71478 images in total. The images have been scaled to fit within a 150×150 square, keeping the original aspect ratio. Some example topics in this dataset are maps, animals, celebrities from TV, flags, logos, buildings, and so on.

3.2.2 Feature extraction

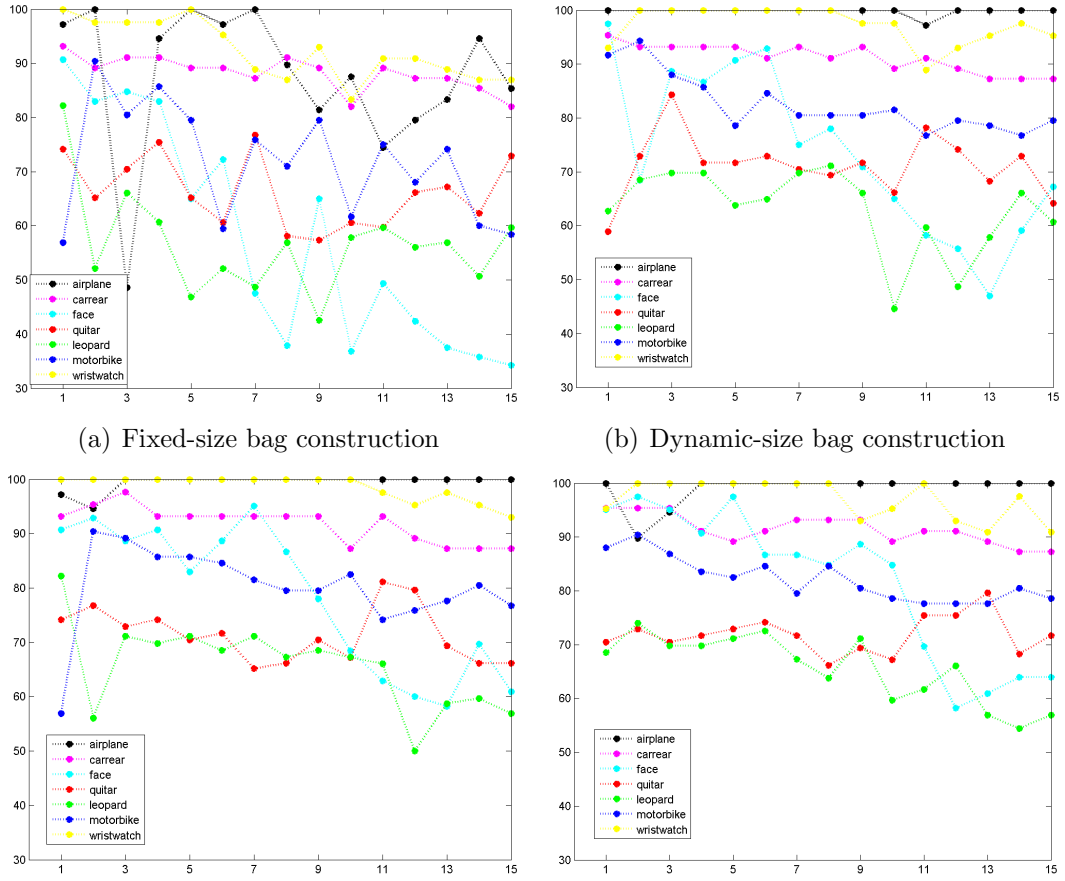
To capture the visual content, each image is represented via its bag-of-words (BoW) histograms. First, dense SIFT descriptors [69] are extracted from each image using VLFeat library [70]. We then cluster these descriptors using k-means (where we set $k = 1000$ in our experiments) and form the visual codebook. Then, each image is represented with its histogram of codewords. While forming the

image representation, $2x2$ spatial tiling is applied to account for coarse spatial information. Each of the local spatial histogram is concatenated with the global BoW histogram of the whole image. The resulting feature vector size is therefore 5000 (1000 for the overall image histogram, 1000 for each spatial quadrant).

3.2.3 Evaluation of the bag-size and bag construction approaches

We first investigate whether there is a fixed bag size k that produces effective results for each dataset. Extensive evaluation of choosing the bag-size k and different MI-bag construction approaches over the Google dataset [5] are given in Fig. 3.4 , Fig. 3.5, Fig. 3.7 and Fig. 3.8. Below, we describe each of the experiments in greater detail.

Fixed-size bag construction: We first evaluate the simplest bag construction method, i.e. using fixed-size bags. For each category, we show the effect of using various bag sizes $k = 1, 2, \ldots, 15$ in terms of precision at 15% recall values in Fig. 3.4(a) and average precision(AP) in Fig. 3.5(a) . The results show that fixed-size bag construction is quite dependent on the choice of k. We observe that the average precision is mostly higher for the lower values of k (such as $k = 1, \ldots, 3$), however, there is no optimal value which performs best for each of the categories. Moreover, the performance fluctuates quite rapidly based on the choice of k. This is not surprising, since for each image query, the relevancy of the initial retrieved ranking list is quite versatile and dependent on many factors of used text-based retrieval scheme. We see that for some choice of k, the re-ranking performance increases, this is due to the generation of more suited MI-bags to the retrieval order. On the contrary, for some choice of k, the performance decreases and this is due to the increased noise content in the MI-bags. Since there is no explicit labels or user feedback, it is not possible to select the optimal k for each query.



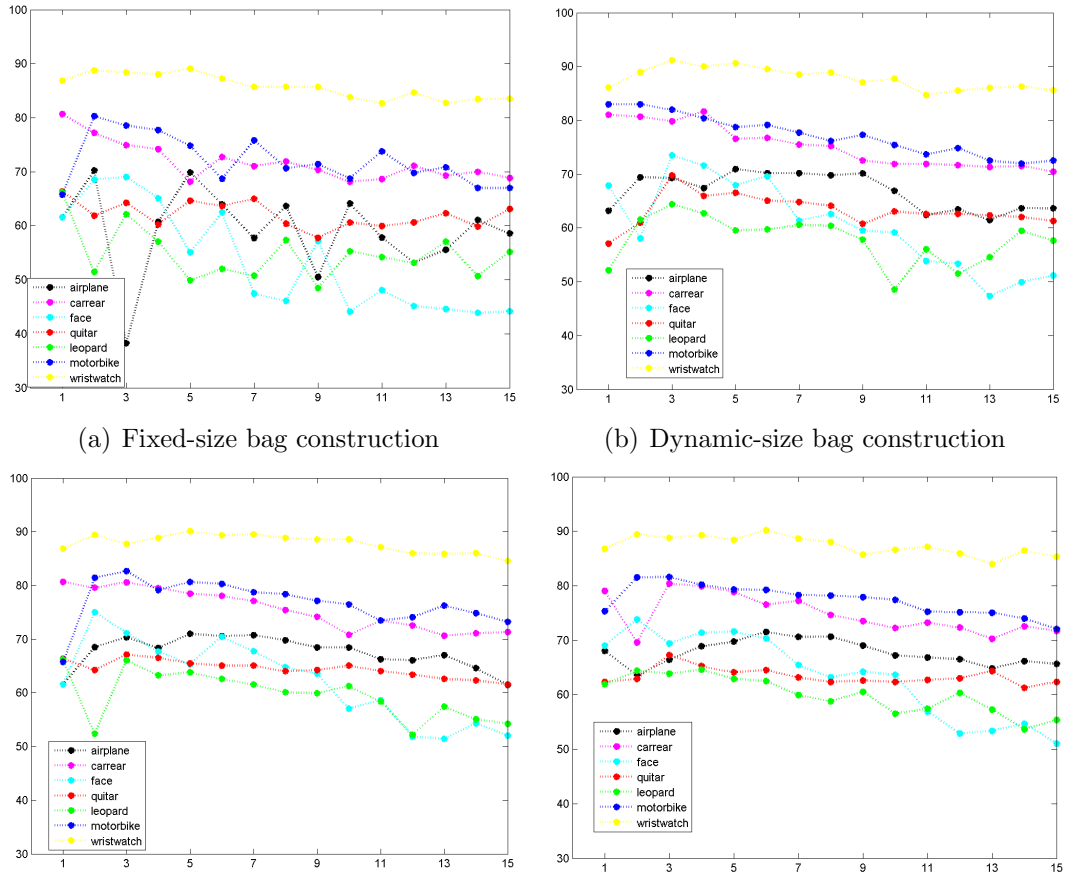
(a) Fixed-size bag construction

(b) Dynamic-size bag construction

(c) Sliding window bag construction

(d) Dynamic-size Sliding window bag construction

Figure 3.4: Effect of choosing different bag sizes k using the four proposed MI-bag construction methods. The results presented here are the precision at 15% recall values achieved on the Fergus dataset [5].



(a) Fixed-size bag construction

(b) Dynamic-size bag construction

(c) Sliding window bag construction

(d) Dynamic-size Sliding window bag construction

Figure 3.5: Effect of choosing different bag sizes k using the four proposed MI-bag construction methods. The results presented here are the average precision(AP) values achieved on the Fergus dataset [5]. We observe that the fixed-size bags are affected very much from the choice of k and produces rather unstable results, whereas the sliding window(SW) and dynamic-size sliding window(DSW) approaches are less affected from the change in k. From this figure, we also observe that there is no global optimal choice of k that produces the best results for all the queries.

Table 3.1: Precision at 15% recall for the dynamic-size bag construction where $\gamma = N/2$ and $k = 2$ for the first interval.

σ	airplane	car_rear	face	guitar	leopard	mbike	wrwatch	mean
1	100	93.18	88.64	72.88	71.14	94.29	100	**88.59**
2	100	93.18	65.00	76.79	69.81	91.67	100	85.21
4	89.64	95.35	97.50	69.35	68.52	91.67	100	87.43
6	100	93.18	95.12	60.56	71.15	85.71	100	86.53
8	100	93.18	88.64	72.88	68.52	74.16	97.56	85.00
10	100	97.62	84.78	66.15	71.15	90.41	97.56	86.81

Dynamic-size bag construction: In dynamic bag construction, we divide the retrieved list of N images to subsets of size $N/2$ and for each subset, the size of the MI-bag is incremented by 1 (i.e. $\gamma = N/2$ and $\sigma = 1$). Figure 3.4(b) and Figure 3.5(b) shows the performance of this method using varying k. In this figure, as with the case of fixed-size bags, the performance is highly sensitive to the choice of k. However, especially for some values of k, the results are better than using fixed-size bags. This result is in accordance with our initial observation that the retrieved list of images tend to contain relevant images ranked higher in the list, whereas the lower portions of the retrieval list contain images that are less relevant. Since the frequency of seeing relevant images decreases as we move down the list, increasing the MI-bag size affects the performance positively.

For dynamic-size bag construction, we evaluate the choice of σ (amount of increase in each subinterval) and γ (the interval size). The results are given in Table 3.1 and in Fig. 3.6, respectively. In Table 3.1, we look into the effect of increasing the bag sizes as we move further down the initial retrieval list. As these results show, in our experiments, we observe no significant trend related to the choice of σ. Overall, increasing the bag size is more effective compared to using fixed-size bags, whereas using gradual increments is likely to be more promising. Based on this observation, we set $\sigma = 1$ for the rest of the experiments.

In Figure 3.6, we show the effect of varying γ intervals, where the retrieval list is divided into $N/2$, $N/3$ and $N/4$ intervals and in each interval the bag size is incremented by 1. We observe that, $\gamma = N/2$ produces slightly better results, thus set $\gamma = N/2$ for the rest of the experiments.

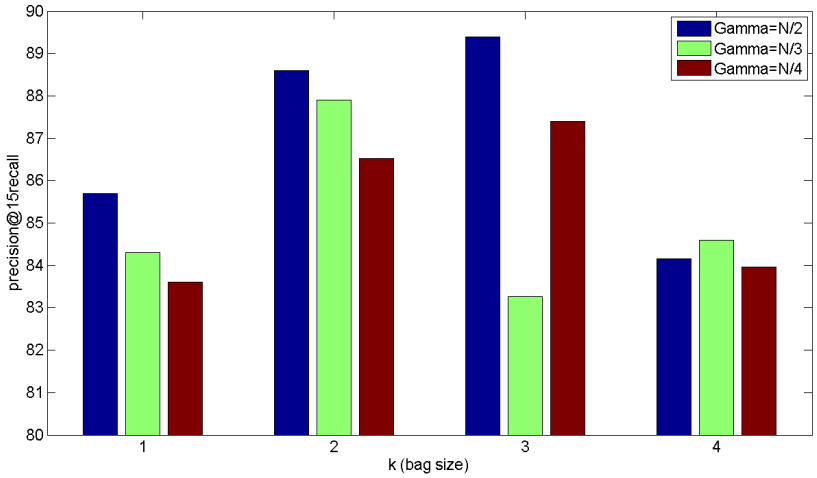
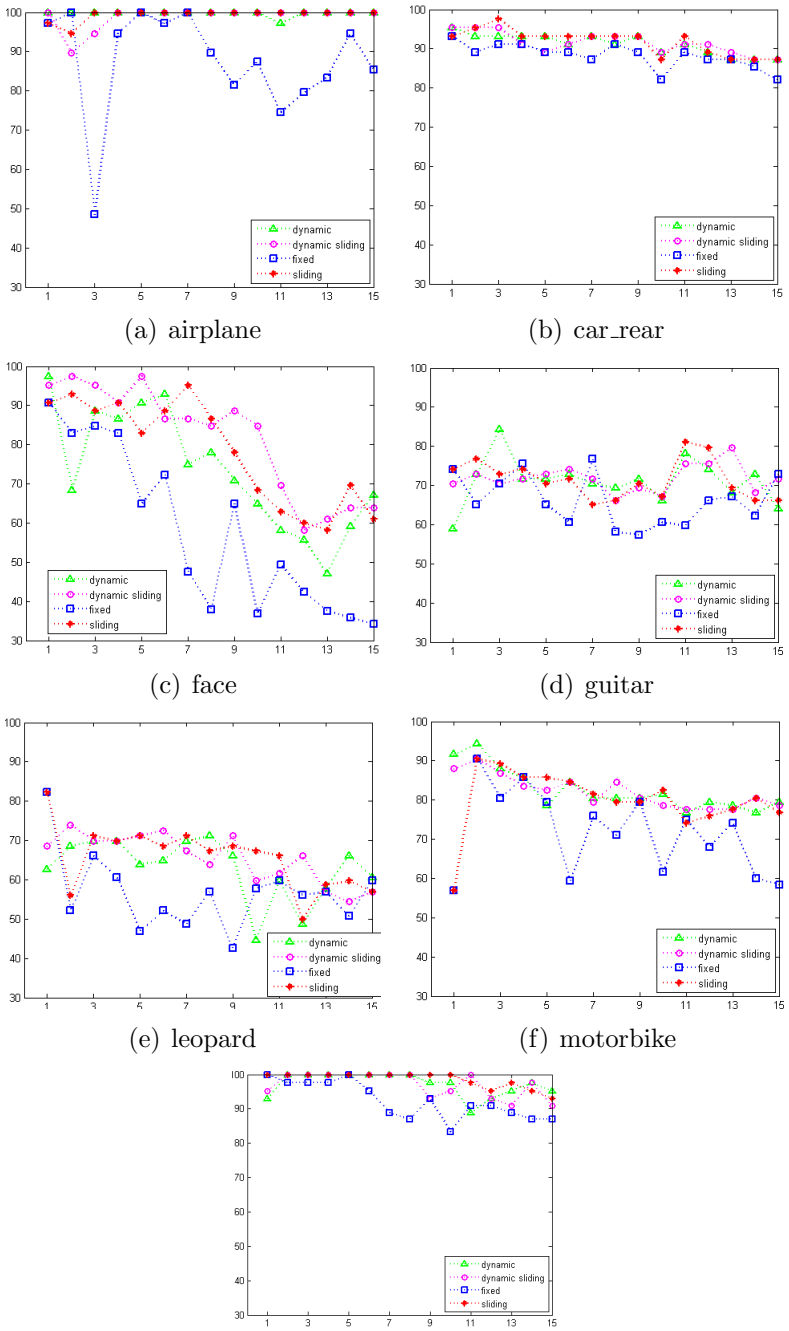


Figure 3.6: Mean Average Precision at 15% recall for the dynamic-size bag construction where $\sigma = 1$ and k changing in Fergus [5] dataset.

Sliding window bag construction Sliding window(SW) approach for constructing MI-bags can be used with both fixed-size bags and dynamic-size bags. For the case with the fixed-size bags, the results are given in Figure 3.4(d) and Figure 3.5(d). From these figures, we observe that SW approach is less affected from the choice of k compared to fixed-size or dynamic-size bag construction methods. On the other hand, still, there is no global k that is optimal for every query. In Figure 3.4(d) and Figure 3.5(d), the results when sliding window approach is used with dynamic-size (dynamic-SW) bags are presented. We observe a similar trend in these results.

Figure 3.7 and Figure 3.8 compares the performance of all the four bag construction methods on different queries in Google [5] dataset. As it can be seen, amongst all four bag construction approaches, the fixed-size bag construction performs the worst. The best performance is achieved by SW approach either with fixed or dynamic-size bags. Figure 3.9 shows the mean performance of those methods with respect to varying k. Again, for different choices of k, either SW or dynamic-SW approach performs the best. We also observe that the performance is relatively higher for lower k values. This implies that, as the bag size increases, the amount of noise present in each bag becomes more dominant and this situation affects classification performance in a negative way.



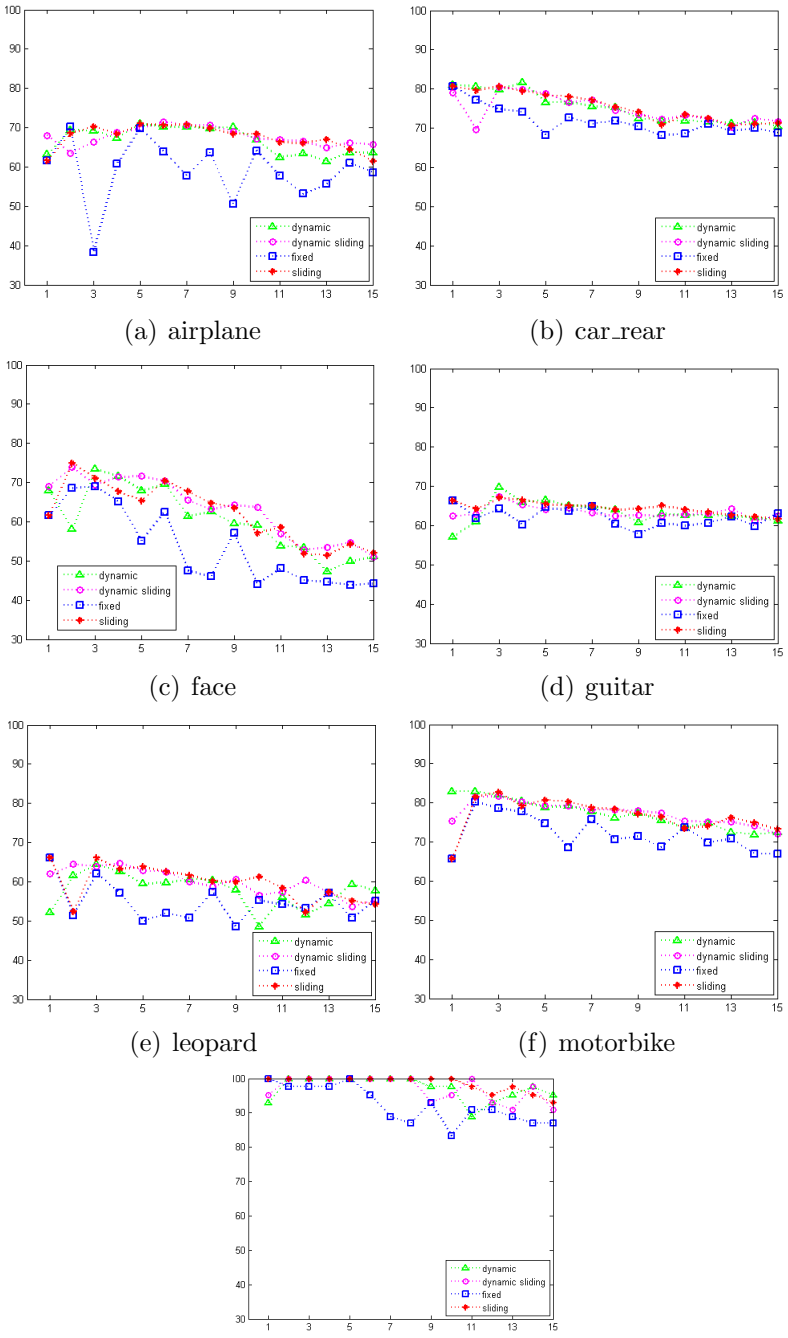
(a) airplane (b) car_rear

(c) face (d) guitar

(e) leopard (f) motorbike

(g) wristwatch

Figure 3.7: The effect of choosing different bag sizes with different bag construction approaches and varying initial bag size k on the Fergus Google dataset [5]. Here, the precisions at 15% recall level are shown.



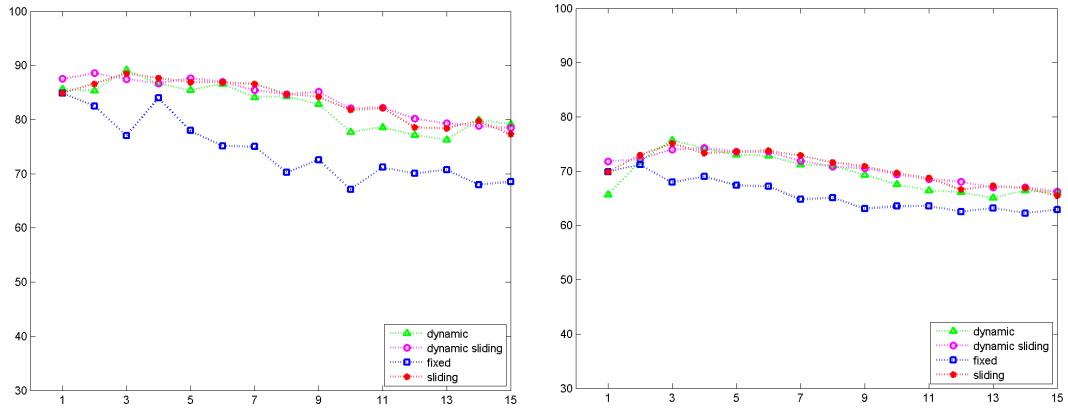
(a) airplane (b) car_rear

(c) face (d) guitar

(e) leopard (f) motorbike

(g) wristwatch

Figure 3.8: The effect of choosing different bag sizes with different bag construction approaches and varying initial bag size k on the Fergus Google dataset [5]. Here, the average precision(AP) level are shown.



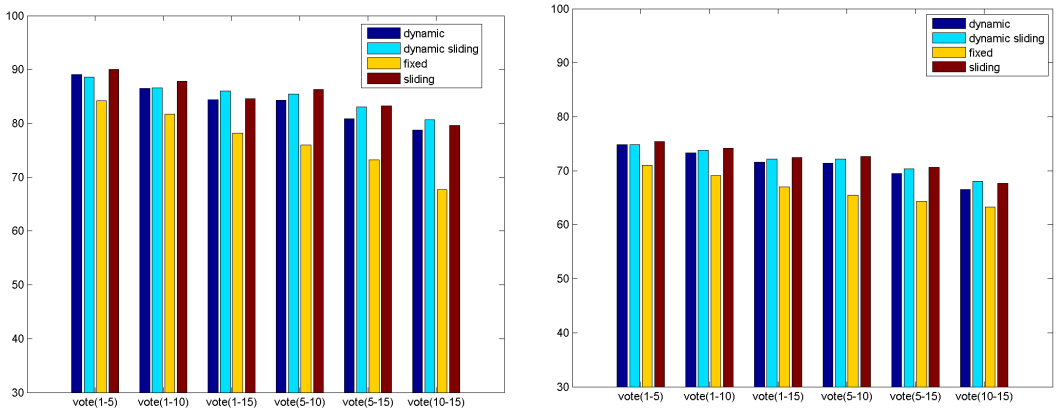
(a) Precision 15% Recall (b) Average Precision

Figure 3.9: Mean performance of the four different MI-bag construction methods on the Fergus Google dataset [5] with respect to changing bag size k. To the left, the precisions at recall 15% are shown, and to the right, the average precision values are given. Sliding window(SW) based MI-bag construction methods are more likely to produce better results.

3.2.3.1 Using Ensembles of MIL Classifiers

The results show that the re-ranking performance is quite affected by the choice of k parameter. Choosing the optimal k parameter is not feasible, since our method do not use any supervision or user feedback. In order to deal with this issue, we propose to train multiple MI classifiers that works on bags of varying sizes. Ultimately, the responses of these classifiers are combined for final decision. In this way, we bypass the need of choosing the bag size and reduce the number of parameters that need to be tuned.

The results of using such ensemble classifiers is shown in Fig 3.10. From these results, we observe that combining multiple classifiers produces more effective reranking results, and on average, 1% to 5% point precision gain is achieved as opposed to using single MI-classifiers with a particular choice of bag size. The best performing method in Google dataset is using sliding window with fixed-size bags, where the bag size is $k \in 1 \dots 5$. Using this range seem to perform the best for all methods in our experiments, therefore, we construct multiple bags of size 1 to 5 in the rest of the experiments. Figure 3.11 compares the performance of



(a) Precision 15% Recall

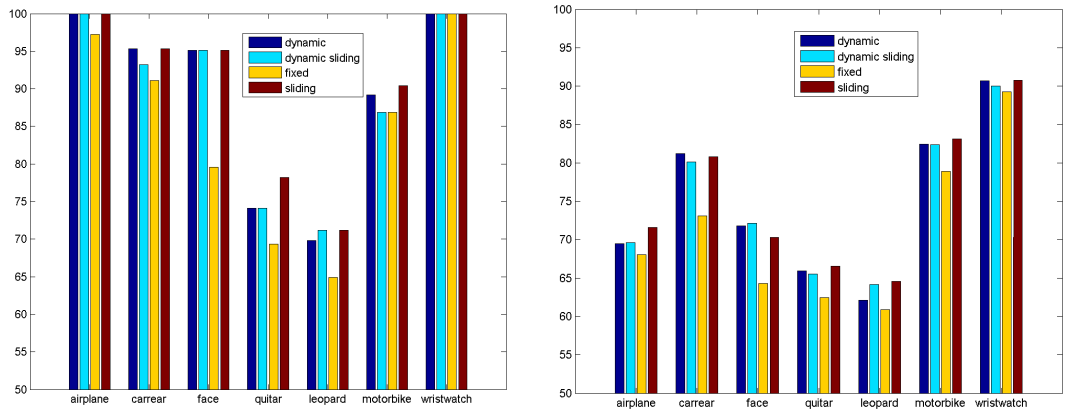
(b) Average Precision

Figure 3.10: Using ensemble of MI classifiers with different bag sizes k and different bag construction schemes over Google dataset. vote(k_1,k_2) shows that $k \in k_1 \dots k_2$. In this dataset, using sliding window(SW) with fixed size bags produces the best result, whereas using SW with dynamic size windows is the second best. According to these results, using classifiers with bags built with $k \in 1 \dots 5$ gives the highest precision.

all the four bag construction methods using ensemble of multiple bags of size 1 to 5 on different queries in Google [5] dataset.

3.2.3.2 Evaluation of Distance function and BoW representation

We further evaluate the effect of the distance function used in instance embedding step of the MILES classifier, i.e. D function in Eq.2.1. The precisions at recall 15% and average precisions are presented in Table 3.2 and Table 3.3, respectively. The experiments show that when Euclidean distance is used, using the square rooted BoW feature vector, which is equivalent to Hellinger kernel over BoW vectors [71], produces better results. Using chi-square distance with standard BoW representation yields the highest precision value at recall 15%. Note that using chi-square distance with square rooted BoW features yields slightly higher average precision.



(a) Precision 15% Recall (b) Average Precision

Figure 3.11: The effect of choosing voting 1-5 for four bag construction methods. To the left shows the precision at recall 15% and the subfigure to the right shows the Average Precision. Using sliding window(SW) with fixed size bags produces the best result among different queries in Google [5] dataset

Table 3.2: Precision at 15% recall level is shown. D corresponds to the distance function for MIL instance embedding step, and BoW representations are either used in standard or in Hellinger-kernelized form.

D	BoW	airplane	car_rear	face	guitar	leopard	mbike	wrwatch	mean
euc	normal	100	95.35	95.12	78.18	71.15	90.41	100	90.0
euc	Hellinger	100	97.62	100	89.58	62.71	95.65	100	92.2
chi	normal	100	100	97.5	82.69	75.51	97.06	100	**93.3**
chi	Hellinger	100	100	92.86	84.31	67.27	95.65	100	91.4

Table 3.3: Average Precision : Parameter optimization and best method. ed : euclidean distance for MILES, ed-sqrt : euclidean distance for MILES with sqrt of BoW histograms, chi : chi-square distance for MILES, chi-sqrt : chi-square distance for MILES with sqrt of BoW histograms,

D	BoW	airplane	car_rear	face	guitar	leopard	mbike	wrwatch	mean
euc	normal	71.56	80.78	70.31	66.56	64.58	83.05	90.75	75.38
euc	Hellinger	68.19	82.22	74.39	71.81	60.62	86.56	92.16	76.56
chi	normal	68.40	83.03	73.49	72.02	64.10	87.00	92.72	77.25
chi	Hellinger	72.11	83.05	73.04	73.04	61.81	85.46	92.95	**77.35**

3.2.3.3 Best performance

We evaluated the best performance of our method. We selected only positive examples returned by search engine to create multiple instance bags, then we constructed our bags via our best method sliding window voting 1–5 . After that we classified with best parameters. Additionally we give k = 1 results. The results are given in Table 3.4 and Table 3.5. *"k=1-5"* : Our best method voting 1-5 sliding window with best parameters. *"k=1-5 pos."* : We used just positive images returned by search engine with best method voting 1-5 sliding window with best parameters and *"k=1"* : only positive instances in query is used for k=1 with best parameters.

Table 3.4: %15 Recall best performance : The best performance of our method sliding windows with $k = 1 \dots 5$ and chi-square distance for MILES, then svm with rbf kernel is used. k=1-5 : all instances in query is used, k=1-5 pos : only positive instances in query is used, k=1: only positive instances in query is used for k=1.

method	airplane	car_rear	face	guitar	leopard	mbike	wrwatch	mean
k=1-5	100	100	97.5	82.69	75.51	97.06	100	93.25
k=1-5 pos.	100	97.62	100	91.49	80.44	97.06	100	95.23
k=1	100	97.62	100	100	82.22	97.06	100	**96.70**

Table 3.5: Average Performance : The best performance of our method sliding windows with $k = 1 \dots 5$ and chi-square distance for MILES, then svm with rbf kernel is used. k=1-5 : all instances in query is used, k=1-5 pos : only positive instances in query is used, k=1: only positive instances in query is used for k=1.

method	airplane	car_rear	face	guitar	leopard	mbike	wrwatch	mean
k=1-5	68.40	83.03	73.49	72.02	64.10	87.00	92.72	77.25
k=1-5 pos.	88.61	90.77	91.18	85.98	75.07	92.85	98.93	89.06
k=1	93.56	95.07	95.67	91.91	82.14	94.89	99.20	**93.21**

3.2.4 Comparison to state-of-the-art

Next, we compare our approach to state-of-the-art approaches both on Fergus and on Web Queries datasets. In Table 3.6, the comparisons for Fergus dataset is given. In this table, *Ours* indicate the results ensembles of MILs with $k =$

$1 \ldots 5$ where the MI-bags are constructed via sliding window (SW) with fixed size bags, since this method performs the best amongst the four alternatives. Chi-square distance is used for MIL instance embedding stage and L2-regularized linear SVMs are used over the embedding space. As shown, our method achieves superior results compared to state-of-the-art for this dataset.

Table 3.6: Comparison to state-of-the art on Google dataset [5]. In this table, precisions (%) at 15% recall are reported.

	airplane	car_rear	face	guitar	leopard	mbike	wrwatch	mean
Google	50	41	19	31	41	46	70	43
[41]	35	-	-	29	50	63	93	54
LogReg [2]	65	55	72	28	44	49	79	56
TSI-pLSA [5]	57	77	82	50	59	72	88	69
WsMIL [72]	100	81	57	52	66	79	95	75.7
SF+MRank [38]	86	100	75	58	63	79	100	80
PMIL [43]	100	75.3	89.9	82.7	86.2	76.6	95.7	86.6
LDA [35]	100	83	100	91	65	97	100	91
Ours	100	100	97.5	82.7	75.5	97.1	100	**93.3**

In Web Queries dataset, we also employ ensembles of MIL classifiers learned over multiple bags, constructed by sliding window approach, where $k = 1 \ldots 5$. Euclidean distance is used for MIL embedding stage. In this dataset, since the modalities within the queries are higher, SVMs with RBF kernel tend to be more effective. Table 3.7 shows the overall results. Our method achieves a MAP of 71.08% on this dataset, which is comparable to state-of-the-art.

Table 3.7: Comparisons to state-of-the art on Web Queries dataset [2] with respect to the Mean average precisions (MAP).

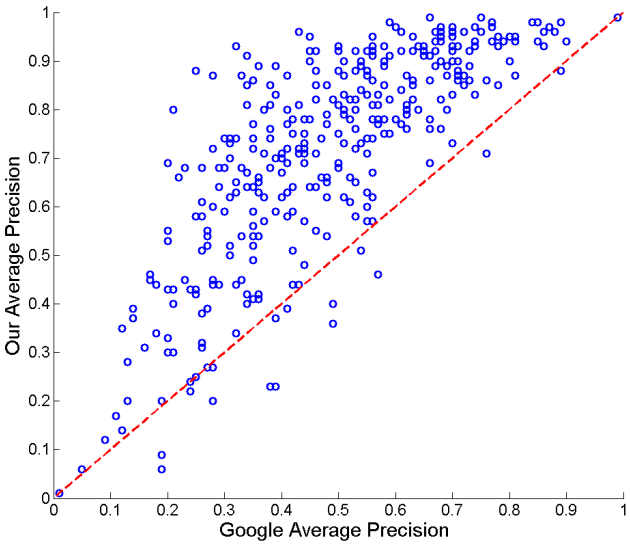
Method	MAP
Search Engine	56.99
[2](visual only)	64.9
[2](visual+textual)	67.3
BLVS [73]	67.0
SpecFilter+MRank [38]	**73.76**
Ours	71.08

We further evaluate our method's performance with respect to the initial

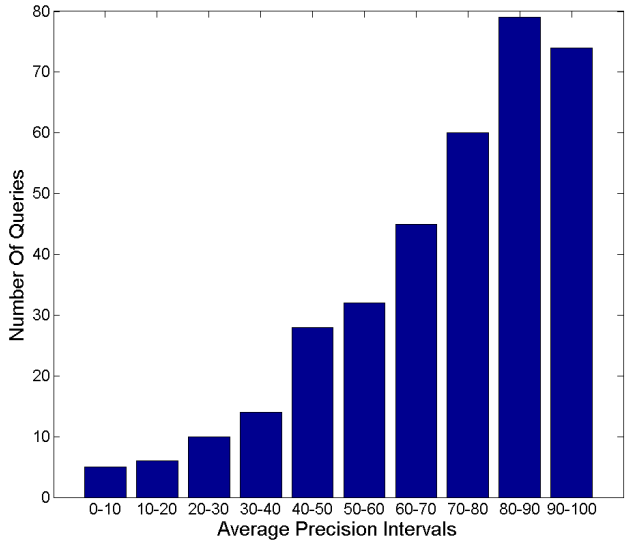
search engine ranking in Fig 3.12. Figure 3.12(a) shows the average precision(AP)s of our reranking method as opposed to their counterpart search engine ranking APs. Out of 353 queries of Web Queries dataset, the AP has degraded in only 14 queries when using our reranking method, and most of the time, our method provides superior ranking compared to the search engine. For some queries that have APs as low as 0.2 or 0.3 in the initial search engine ranking, our method is able to improve the AP to 0.80 and 0.90. Note that, our method does not make use of any auxilary data, textual data or explicit detector/classifier; it relies solely on the visual content and the initial ranking of the images. From Fig 3.12(b), we also observe that most of the queries fall into the high precision range, approximately half of the queries have APs greater than 0.8. In Figure 3.14, some qualitative examples for the re-ranked retrieval lists are given for the Web Queries dataset. Note that our method is able to successfully re-rank various images of queried concept.

Cases of failure: In order to gain further insight about our method's performance, we look at the individual query performance with respect to the positive instance percentage for the queries. Figure 3.13 depicts this evaluation. The linear correlation between the two axes in this graph is rather expected for all methods, since as the percentage of positives increases in the set, the average precision also increases. We observe that our method performs poorly when the ratio of positive instances in the ranking is very small; the AP is especially low when the number of positive instances falls below 3. In this case, the MI classifiers cannot perform well, since there are relatively very few examples to learn from.

We also observe that, for queries that have one or more dominant groups, the performance can be relatively poor. For example, in "Jack Black" query, the dominant set is the black jack table and the multiple instance bags are dominated by such images. Similarly, for "Orsay Museum" query, most of the images show the interior of the museum, whereas only the exterior of the museum is labeled as positive. Our approach tends to rank the interior set of images higher in the retrieval list, and therefore the performance of those queries are inferior. More



(a) Average Precision of Google vs Ours



(b) Average Precision Intervals

Figure 3.12: a) Comparison of our method wrt search engine in terms of individual query APs in Web Queries dataset. We observe that for most of the queries, our method provides higher APs. b) In the result of our method, the distribution of the query APs are shown. Approximately half of the queries have APs ≥ 80

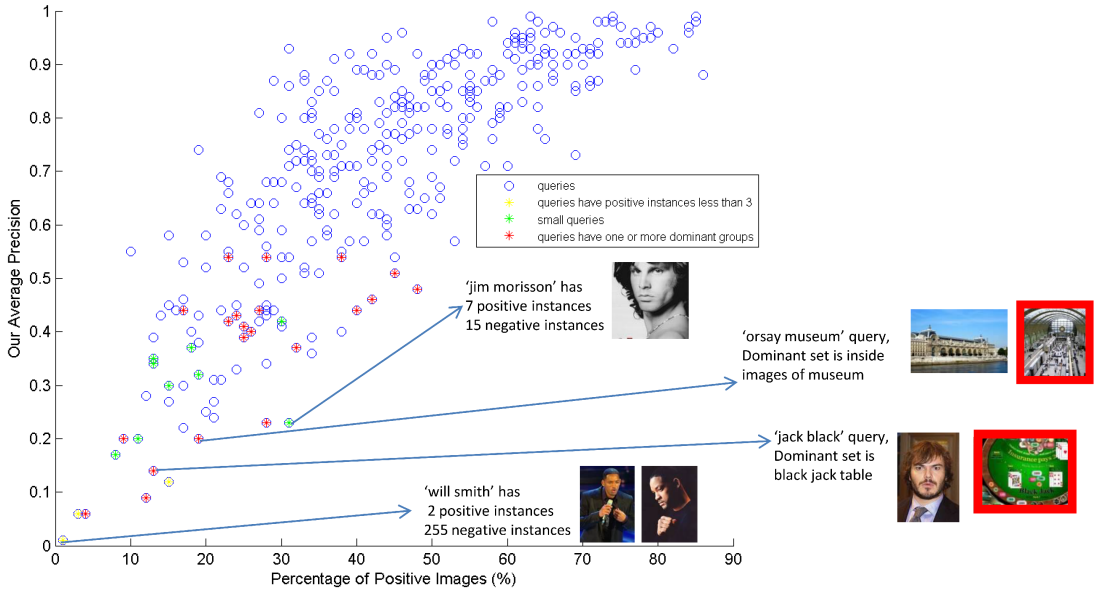


Figure 3.13: Our method's average precision vs the percentage of positive images returned by the search engine. When the number of actual positive instances returned by the initial retrieval are very low for some query(shown in yellow), the classifiers are not able to form reliable models for the queried concepts. Similarly, if the returned image list is relatively sparse, i.e. if it does not include many examples, the AP can also be low (queries shown in green). Another interesting observation is that, when the queries include more than one dominant set (shown in red), the multiple instance learners can focus on the unintended dominant set, and as a result, the re-ranked list can have a lower AP.

examples of such cases, where there are more than one dominant group in the query are shown in Figure 3.15.

Figure 3.14: Examples of the retrieval order obtained by our method. Top 10 images for each query are shown. The queries are (from top to bottom): 4x4 (1st row), Mickey (2nd row), Times Square(3rd row), Italy map (4th row), arc de triomphe (5th row , tomato (6th row), piano (7th row), cat (7th row), dollar (8th row), Dome Florence (9th row), Leonardo di Caprio (10th row), crocodile (11th row), shark(12th row), Guernica(13th row), firefighter truck(14th row). The irrelevant images for each query are marked with red.

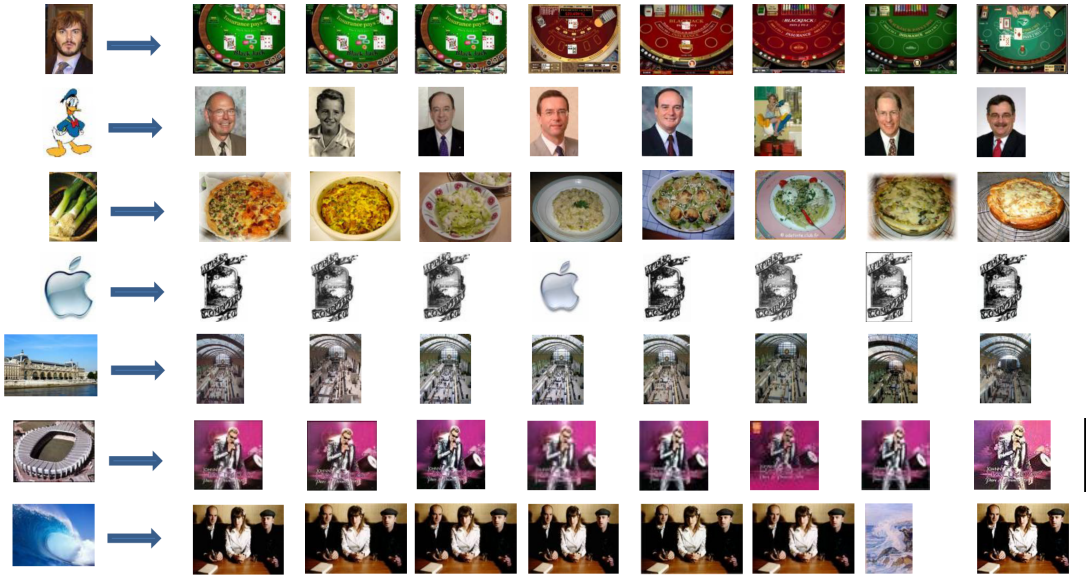


Figure 3.15: Examples for the cases in which our method performs relatively poor. For each query, the positive example is given to the left of the list, and to the right is the re-ranked order obtained by our algorithm. The queries are (from top to bottom): Jack Black (1^{st} row), Donald Duck (2^{rd} row) , leeks (3^{rd} row), logo apple (4^{th} row), Orsay museum (5^{th} row), Parc des Princes (6^{th} row), wave (7^{th} row). As it can be seen, in this queries, there are more than one dominant visual case in the retrieval list, and our method focuses on the more frequent one. For example, for Orsay museum query, the images returned are mostly from the inside of the museum, which are labelled as negative for that query. Simiarly, for the "leek" query, the returned images mostly consist of dishes made with leek, which is also another dominant visual occurence.

Chapter 4

Recognizing Actions in Still Images using Multiple Instance Learning

We propose a multi-cue based approach for recognizing human actions in still images, where relevant object regions are discovered and utilized in a weakly supervised manner. Our approach does not require any explicitly trained object detector or part/attribute annotation. Instead, a multiple instance learning approach is used over sets of object hypotheses in order to represent objects relevant to the actions. We test our method on the extensive Stanford 40 Actions dataset [3] and achieve significant performance gain compared to the state-of-the-art. Our results show that using multiple object hypotheses within multiple instance learning is effective for human action recognition in still images and such an object representation is suitable for using in conjunction with other visual features.

The remaining of the this chapter is organized as follows: We first present the various features utilized for recognizing actions in still images, especially the MIL approach for objects in 5.1. In Section 4.2, we present the extensive evaluation of the features in the Stanford 40 actions [3] dataset.

4.1 Multiple Features for Actions in Still Images

4.1.1 Multiple Instance Learning for Candidate Object Regions

In order to recognize actions in still images, the related objects can be particularly important. In this paper, instead of using explicit object detectors, we investigate whether we can automatically learn potential object regions that can boost action recognition performance. For this reason, we extract several candidate object regions and use these object regions in a Multiple Instance Learning(MIL) framework.

We assume that the objects that the people are interacting with are visually salient objects. We use objectness measure [6] for finding visually salient regions within the image. Objectness measure uses several cues (such as multi-scale saliency, color contrast, edge density, etc.) in an image to identify regions for generic objects. We use this measure to identify candidate object hypotheses. Figure 4.1 shows example images. As it can be seen, in some images, objectness measure is able to locate objects of interest such as rowing boat. However, this measure also generates some noisy regions that do not include any related object.

In our implementation, we sample 100 windows from each image based on their objectness measure, i.e, the probability of containing an object. The authors of [6] recommend sampling 1000 image windows to cover all possible objects, but it would be very costly for the scalability of the approach. Therefore, we limit the sampling to 100 windows. We then extract dense SIFT feature vectors from each of these windows, and describe each via its bag-of-words representation using $2 \times 2 + 1 \times 1$ spatial tiling. The used codebook size is 1000 and the final feature vector dimensionality is 5000.

After sampling 100 windows from each image, we use k-means over the appearance feature vectors and group these 100 windows into 10 clusters. We use

Figure 4.1: Candidate object regions found by objectness measure [6]. The person bounding box is shown in blue and object regions are in red. Candidate object regions form the instances of the corresponding MIL bags.

the cluster centers as our representation of candidate object regions. This step reduces the number of candidate object regions and also focuses on more condensed regions of potential objects. It is also likely that this clustering step smooths out the effect of the noise within candidate object regions. Figure 4.2 shows our work step by step.

As a result, we obtain multiple candidate regions from each image, some of which are likely to contain relevant objects for particular actions. However, we do not know which of these regions are related to the action. This case is particularly suitable for Multiple Instance Learning (MIL), since there are several candidate regions where some of them are noisy and some of them could potentially include related contextual object for the action. In the traditional supervised learning, the learning procedure works over instances x_i and their corresponding labels y_i. In contrast, multiple instance learning operates over bags of instances, where each bag B_i is composed of multiple instances x_{ij}. In our formulation, each image can be considered as a "*bag*" of possible object regions and each extracted candidate

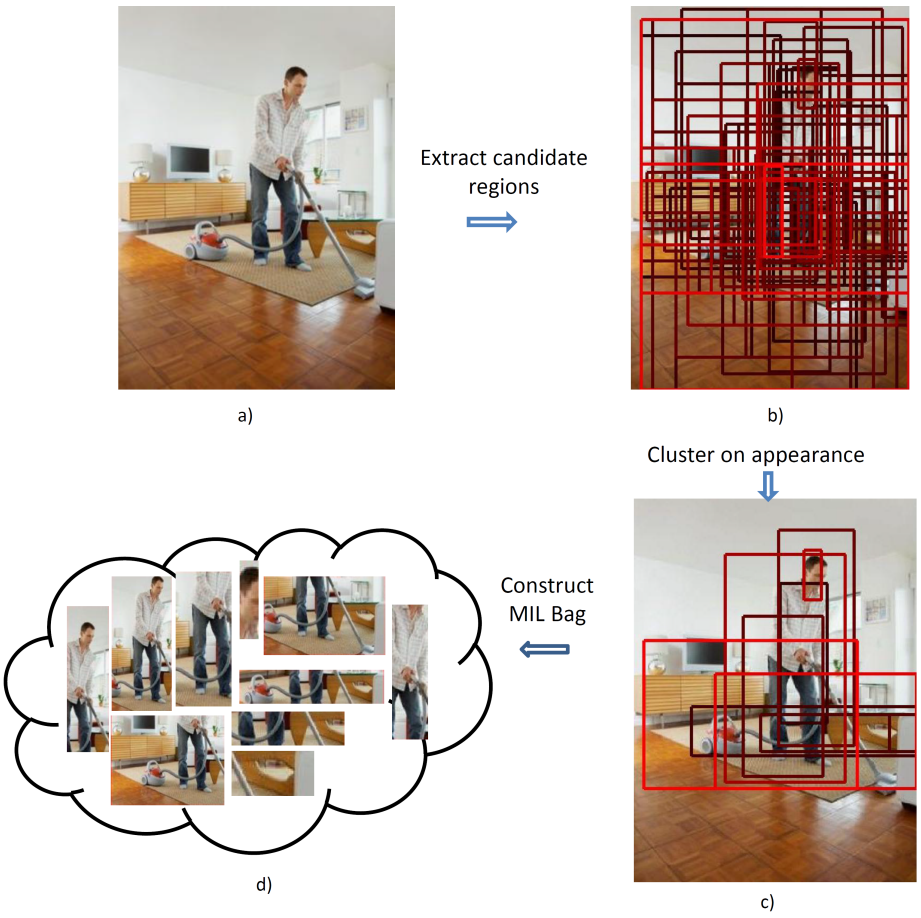


Figure 4.2: Formation of bags from the still images. We first sample 100 windows from image a) based on their objectness measure [6] which is image b). Then we use k-means over the appearance feature vectors and group these 100 windows into 10 clusters. We show the closest candidate windows to 10 clusters in c). Then we form our bags d).

object region is a corresponding "*instance*" inside the bag. A bag B_i is labeled as positive, if at least one of the instances x_{ij} within the bag is known to be positive, whereas it is labeled as negative, if all the instances are known to be negative. This form of learning is referred as "semi-supervised" (or "weakly supervised"), since the labels for the individual instances (in our case, individual object regions) are not available, and only labels of the bags are given.

Given the extracted candidate bounding boxes, we adopt Multiple Instance Learning with Instance Selection (MILES) [32] algorithm for learning the related object regions. MILES algorithm works by embedding the original feature space

x, to the instance domain $\mathbf{m}(B)$. Each bag corresponds to an image and therefore has an associated label $Y_i \in A$, where $A = \{a_1, ..., a_M\}$ is the possible set of M actions. Each bag is represented by its similarity to each of the instances in the dataset. Using the embedded representation, we then train an L2-regularized SVM with RBF kernel for each action class in a one-vs-all manner.

In our formulation, since the number of images and number of windows extracted from each image is high, we can cluster the instances and find the "concept instances" for a more scalable representation. We use the Euclidean distance for the distance between a concept instance c_l and a bag instance x_{ij} and for the concept instances c_l, we either use all the object regions or cluster the instances via k-means and use the cluster centers as c_l for each action. We evaluate the effect of this clustering in the experiments section.

4.1.2 Facial Features for Action Recognition

For quite a number of actions, facial features can be an indicator of the ongoing action. For example, for catching action, the person can be looking into some direction focusing on the thrown object. Similarly, the objects around the face can be a cue for the actions such as talking on the phone, brushing teeth, and so on. Based on this observation, we investigate the effect of facial features for generic action recognition in still images. In [74], it has been shown that facial features can be useful in interaction recognition, and here we investigate their effect to generic actions.

With this intuition, we run a face detector [75] and for images in which the faces are detected, we extract an extended bounding box around the face area as shown in Fig. 4.3. For the images in which no face is detected, we use the top region of the person bounding box as the face area. From these regions, we extract dense SIFT [76] features and employ bag-of-words. We cluster the face images and form a codebook using k-means ($k = 1000$). Then using 2×2 spatial tiling, we extract the codeword histograms from each of the spatial bins. We also concatenate the bag-of-words histogram of the overall face region, hence the final

Figure 4.3: The first three images show the person bounding boxes and the face detector outputs, and the latter ones shows face regions determined wrt person bounding boxes.

feature vector size becomes 5000.

4.1.3 Additional Features

We also include additional features which are frequently used for action recognition to our evaluation framework. For this purpose, we extract the Histogram of Oriented Gradient(HOG) features from the person regions in the image. Furthermore, bag-of-words(BoW) representations extracted from person bounding boxes have also been evaluated. For this purpose, similar to BoW extracted around the faces, the SIFT features are densely extracted from the person regions and k-means clustering (with $k = 1000$) is applied to form the corresponding codebook. Then, 3×3 spatial binning is applied and all the codebook histograms from each spatial bin are concatenated with the global histogram extracted from the whole person region. In the end, the final feature vector for person BoW representation is 10000 dimensional.

In addition to the features extracted from the person region, we also consider

the features from the original image and form the BoW representation from the whole image. This is also extracted in a similar manner to person BoW, where $3 \times 3 + 1 \times 1$ spatial tiling is used and the resulting feature vectors from each spatial bin are concatenated altogether to form a 10000-dimensional vector.

4.2 Experiments

4.2.1 Datasets and Experimental Setup

In the experiments, we use the Stanford 40 Actions dataset [3], which contains 40 actions and 180-300 images for each action. We use the same train/test split provided, which includes 4000 train images and 5532 test images. The bounding boxes for the people doing the action are provided with the dataset. In our experiments, we use these bounding boxes in extracting person/face HoG and BoW features, both in the train and test phases, simulating the case with a perfect person detector, as in [54].

We train a one-vs-all SVM classifier for each of the feature representations separately. The final classification scores are obtained by linearly combining individual classifier confidences giving an equal weight for each feature representation.

4.2.2 Performance of the individual features

Example object/image regions that are discovered by the MIL training stage are shown in Fig. 4.4. For the visualization purposes, number of candidate object regions in this example run is limited to 10 and the top regions mapped to the most contributing concept instances are displayed. As it can be seen, the algorithm is quite successful in discovering the related object regions. In the "cooking" image, the dish region is discovered, whereas in "walking the dog" example, the dog is successfully located. The MIL method also finds the person region as a

top contributing region in most of the cases.

In Table 4.1, we evaluate the effect of the clustering individual instances versus using all instances in the objectness-based MIL formulation. While the clustering provides a scalable representation that requires much less time (clustering with $k = 300$ runs ~ 14 times faster than no clustering case), using all the candidate object regions for instance embedding produces far more effective results in terms of the classification performance.

We then evaluate the performance of the individual features. Accuracy and mean Average Precision(mAP) values achieved by using individual features are shown in Table 4.2. As it can be seen, the best performance is obtained using our MIL framework over the candidate object regions. This demonstrates that without explicit object detectors, we can extract useful information from the candidate object regions generated, in a weakly supervised manner by means of the multiple instance learning formulation.

Person-based features are also informative. Interestingly, performance of the BoW extracted from the whole image is higher than BoW extracted from the person bounding boxes only. This indicates that, the overall image contains more information than the person bounding box itself and the context information accompanying the person is useful for action recognition.

Figure 4.5 shows the performance of the individual features with respect to each action. Overall, the combination of all the features works the best for most of the actions. Interestingly, for some actions such as "climbing, rowing a boat, smoking and using computer" the performance of the proposed MIL framework performs better than using all features. BoW features over the facial region

Table 4.1: Accuracy and mean average precision(mAP) achieved by our MIL approach.

	accuracy	mAP
objectMIL ($k = 300$)	37.08	34.03
objectMIL ($k = 1000$)	46.78	46.01
objectMIL (no clustering)	**51.34**	**51.80**

Table 4.2: Accuracy and mean average precision(mAP) of individual features and the combinations.

	accuracy	mAP
personHOG	24.75	19.35
personBoW	28.56	21.53
faceHOG	14.01	10.37
faceBoW	17.93	13.83
imgBoW	33.51	26.32
objectMIL	51.34	51.80
imgBoW+objectMIL	52.30	52.23
All(w/o objectMIL)	41.47	36.63
All	**55.93**	**55.55**
Yao [3]	NA	45.7

works best for the actions like "climbing, rowing a boat, playing violin, jumping, watching TV, shooting an arrow, brushing teeth". This is not surprising, since in these actions either the facial expression is representative of the action or the related object is closer to the face area. For "climbing, riding a horse, rowing a boat, playing guitar, riding a bike, playing violin, jumping, throwing frisby, running, applauding, holding an umbrella" kind of actions, HoG features around the face area are even more informative than the BoW counterpart. This may be due to the importance of orientation of faces in these type of actions.

4.2.3 Comparison to state-of-the-art

We compare our method to the state-of-the-art method of Yao et al [3] in Table 4.2 and Figure 4.6. Yao et al.'s method is based on part and attribute representation, where each image is represented via a sparse set of "*action bases*". These action bases are defined as the high level interactions between individual action attributes and action parts. In this respect, the attributes that describe an action are annotated and a discriminative binary classifier is trained for each action attribute. Moreover, each part is modeled by the output of an object detector (pre-trained on ImageNet data) or a pre-trained poselet detector [77].

In Table 4.2, `imgBoW+objectMIL` result shows the performance of our method

without using any person bounding box information and `All` shows the performance of the proposed method using all features described in Section 5.1. Compared to the state-of-the-art result of Yao et. al [3], our method achieves significantly better results, while using much less supervision. Even without assuming the availability of a person detector, the objectness-based MIL method combined with image BoW features provide $\sim 6.5\%$ performance improvement in this extensive dataset.

Combining image BoW features with the object MIL representation provides a slight increase of 0.4% in mAP compared to object MIL alone. On the other hand, considering remaining person-based and face-based features provide an additional 3.3% increase in mAP.

Looking at Fig. 4.6, we observe that our method outperforms the parts and attributes method of [3] for most of the actions, especially for "climbing, playing guitar, playing violin, fixing a car, cooking, smoking, cooking, applauding, phoning, taking photos, texting message" actions. This indicates that without using any explicit object/part detector, our method is able to discover the recurring objects or image regions that contribute to the recognition. On the contrary, [3] outperforms our method especially in "riding a horse, rowing a boat, riding a bike, walking the dog, shooting an arrow, fishing, holding an umbrella, running" actions. This may be due to the success of the explicit detectors in locating certain objects and also due to the shared nature of the attribute classifiers.

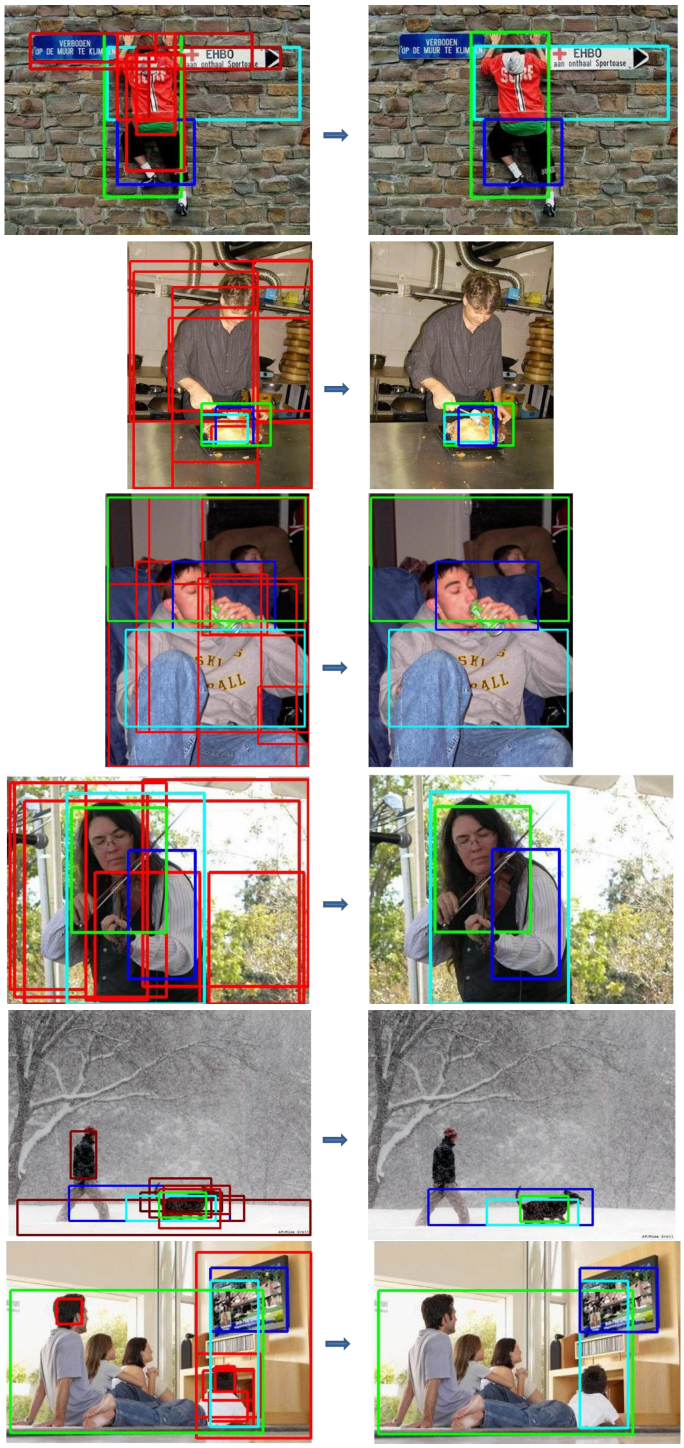


Figure 4.4: An example execution of the MIL framework (best viewed in color). Amongst the 10 example object regions extracted by [6] from the training set, the top 3 regions that contribute to the classification are shown in green, cyan and blue respectively.

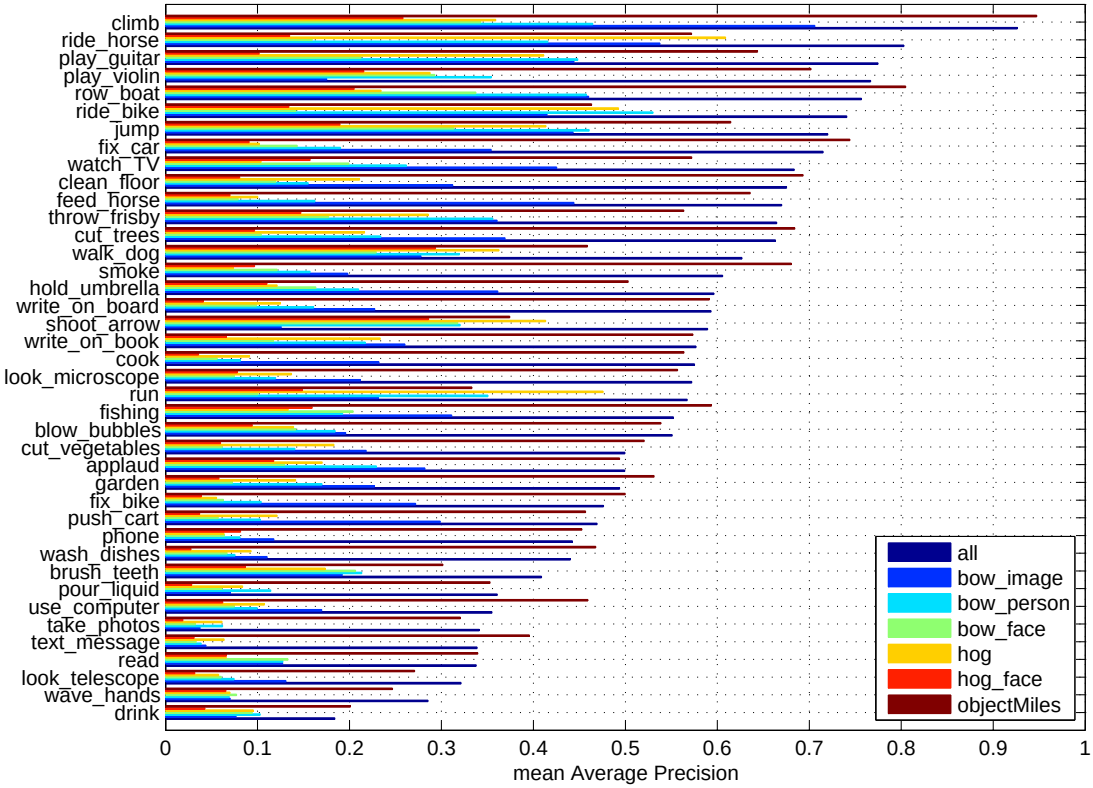


Figure 4.5: Per action mAPs for each of the features (best viewed in color and magnified). Overall, combining all the features' responses works the best. For some actions, the performance of object MIL approach is even better than the combination.

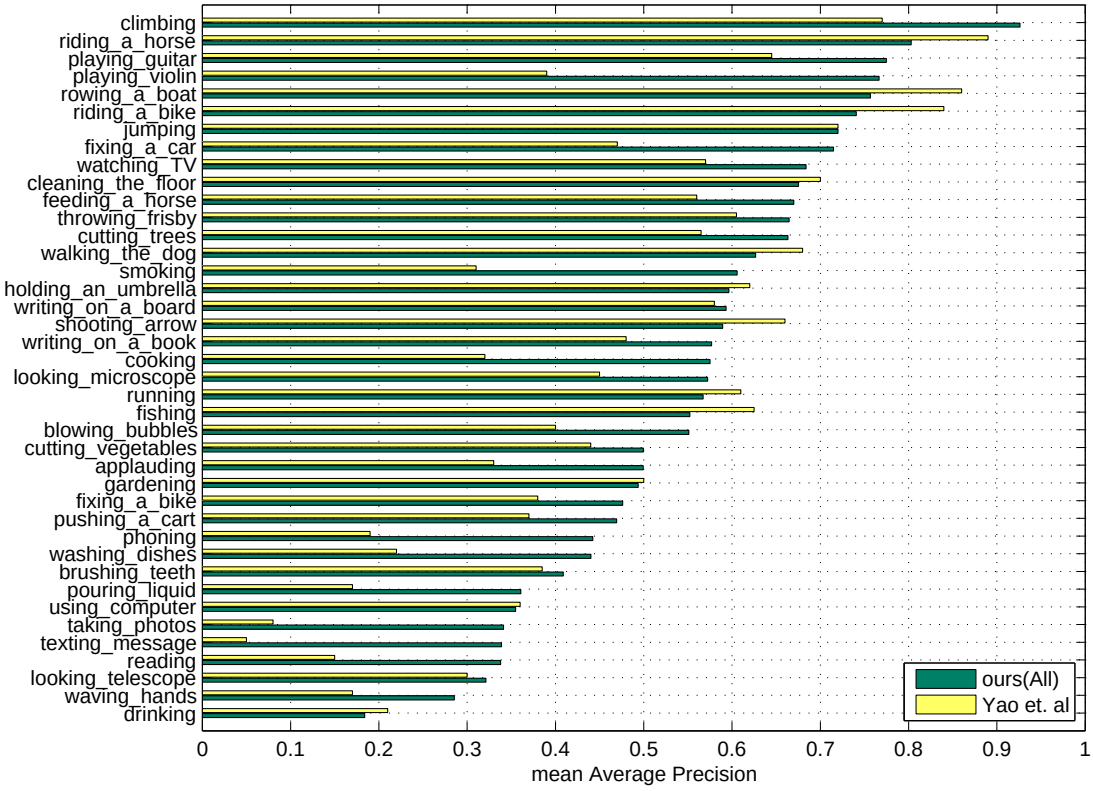


Figure 4.6: Comparison of the proposed approach with that of Yao et al. [3] in terms of classification performance of the individual action classes.

Chapter 5

Recognizing human interactions using Multiple Instance Learning

In this work we look into the problem of recognizing human interactions from videos and propose an approach for two-person interaction recognition which integrates multiple features over different regions types. Real-world videos are weakly annotated; a video has a class label however we do not know in which frame in the video sequence the interaction occurs. We formulate this problem in a multiple instance learning (MIL) framework and form each frame as a bag instance and each video as a bag. We extract several features which are shape, motion and relative distance between two-person from face and body regions over frames. Additionally since the spatial information can be used to complement other features we reformulate our MIL framework by embedding relative distance of two-person to shape and motion information. We test our method on the realistic "TV Interactions" dataset [4] and achieve significant performance gain compared to the state-of-the-art. Our results show that using multiple instance learning with different visual features from different body parts is effective to understand the type of the interaction. Using spatial information together with motion and shape information, we show that better performance is possible.

The remaining of the this chapter is organized as follows: We first present

various features utilized for recognizing two-person interaction from videos in 5.1. In Section 5.2, we present our bag construction way and multiple instance learning formulation of our method. Finally in Section 5.3 we present extensive evaluation of the features in the "TV Interactions" dataset [4]

5.1 Multiple Features for Two-Person Interactions

5.1.1 Modeling Person-Person Relationships

In order to recognize two-person interaction pose of the individuals, orientation of their faces, their motion, distance between them are particularly important. However extracting such type of meaningful and informative features from whole frame is quite difficult. In order to avoid learning background noise we form person-centered descriptors and extract motion, shape and relative distance between individuals features from person regions only.

Since we work with two-person interaction videos, for each interaction we form our descriptor as a combination of two-person region features. For frames include two people, we name the leftmost person as the first person and the next as the second person. Our final descriptor for a frame is the concatenation of the two person region features. In two-person interaction videos it is expected that there should be two people when interaction occurs. So defining such a concatenated descriptor is completely suitable.

We consider two region types for people in each frame. Facial features may be important cues to understand the type of the interaction because while two-person interacting they face each other and orientation of faces changes among interactions. So we use upper body region of a person which perfectly cover faces as one of our region type. Other region type is whole body of person since interaction occurs between individuals changes their pose, some of body parts involve the interactions in different positions, such as hands. In Figure 5.1 we

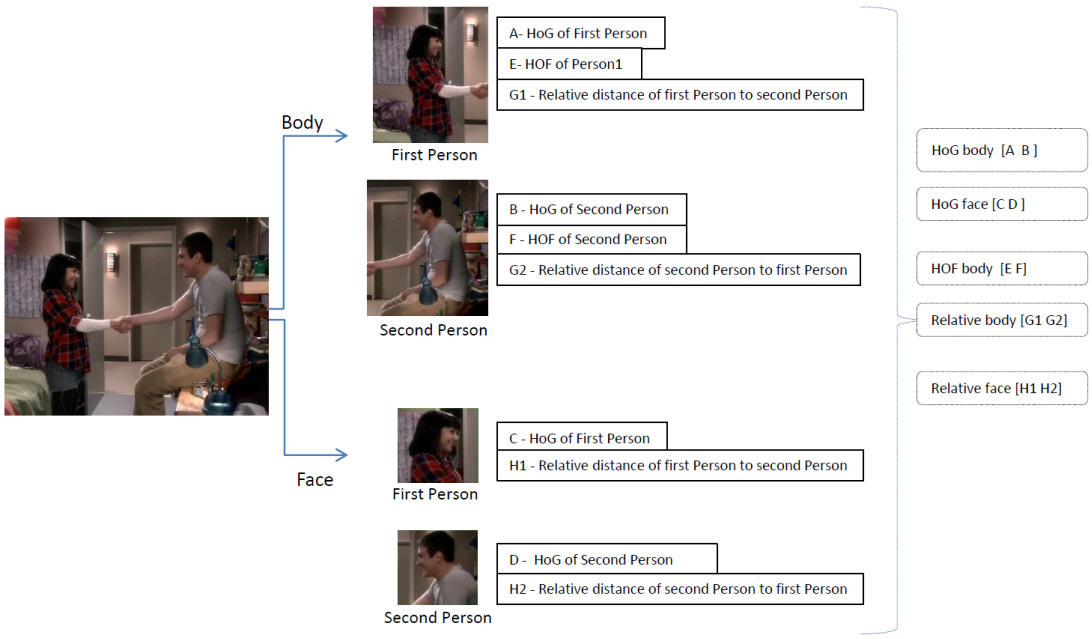


Figure 5.1: Formation of descriptors for different types of features over face and body regions for two-person in a frame of an interaction video. As it can be seen face regions cover faces perfectly and body regions cover all body part of two-person. We name the leftmost person as the first person and the next as second person. We extract multiple features from face and body regions for two-person. Features belong to the first and the second person are then concatenated to create our final descriptors for each feature type.

show our different region types with related features in detail for a frame of an interaction video.

In our implementation we form our descriptors over face and body regions as exampled in Figure 5.1. Body region bounding boxes are extended versions of face bounding boxes. Likewise label for each frame in videos, in unconstrained real world videos any of face or body region bounding boxes are not provided. However there are person detectors have good performance to locate these regions in an unsupervised way. Eichner et al. [78] propose a method to extract upper body locations of people in images based on Felzenszwalb et. al. [79] person detector and additionally they use Viola-Jones face detector [75] to obtain more accurate and less noisy detections. This method may be used over frames of videos to locate regions.

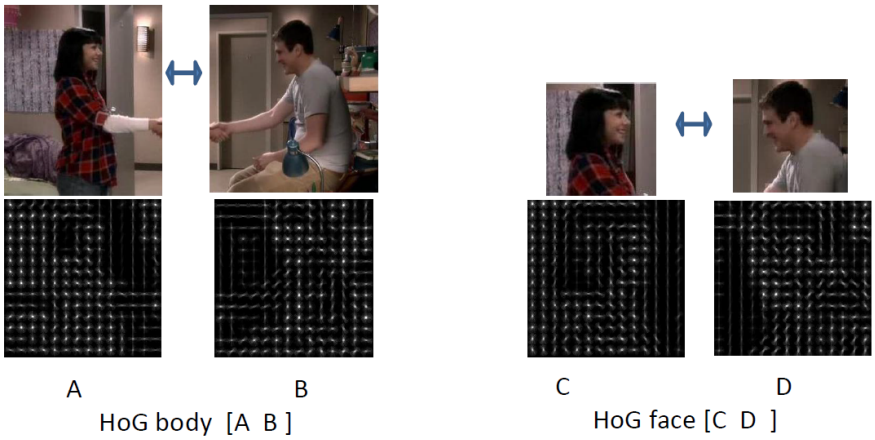


Figure 5.2: We extract shape information from both face and body regions. After first and second person HoG features are extracted, they are concatenated to get final descriptors. `hog_body` and `hog_face` are two of our five descriptors

5.1.2 Image Representation

Our aim is to extract meaningful and informative features from face and body regions of two people. We use features which are frequently used for action recognition and then combine them.

Histogram of Oriented Gradient (HoG) : For two-person interaction case we expect that the pose of the person will be informative. In order to account this pose information, we extract Histogram of Oriented Gradients (HoG) [80] from both body and face regions in each frame. Than we concatenate the first person and the second person HoG features. This step is illustrated in Figure 5.2.

Histogram of Optical Flow (HOF) : We expect that motion features will be complimentary to human shape features. In order to account motion information, we extract the Histogram of Optical Flow (HOF) features from only person regions in each frame. We follow the algorithm proposed in [81], first we extract optical flow of each frame by using previous frame. Then to get spatial information we divide feature region to 3x3 grids and finally form optical flow histograms for four orientations from each grid. Our final future vector has [36x1]

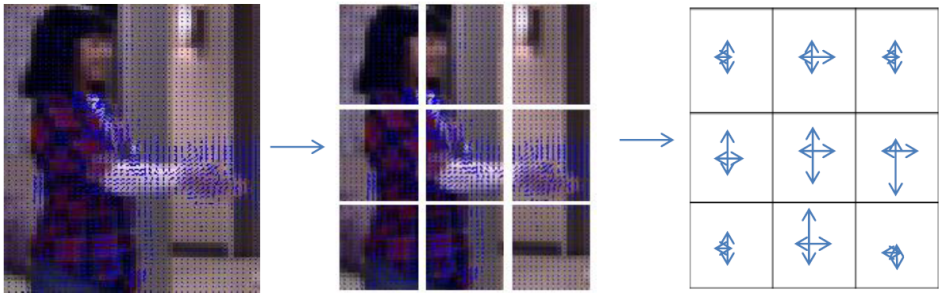


Figure 5.3: We extract motion information from only body regions. After optical flow extracted using previous frame we divide body region to 9 blocks to get spatial information and using four orientations we form our final future vector with dimensions [36x1]. This calculation is done for second person also and then two feature vectors are concatenated to get final descriptor. `hof_body` is one of our five descriptors.

dimensions. This step is illustrated in Figure 5.3.

Relative distance features People interact in many ways and these interactions show a large amount of variability. While interacting, people keep a certain distance to each other based on the interaction type. In order to capture this information and include it in our framework, we encode spatial relations of two-person for body and face regions in each frame. First, we calculate the distance between individuals based on the x and y coordinates of the face and body regions. In order to achieve invariance to differences in scale we normalize the distances of two-person with respect to the height of the first or second person's bounding box. At the end, we concatenate these relative distances and get our final descriptor with 1+1 = 2 dimensions. We exampled this feature in Figure 5.4.

In our study we use the relative distance features between two-person as complementary features to motion and shape features. In first case, we use these features separately as shown in Figure 5.4. In second case we look at exploiting spatial information by embedding it to motion and shape information. We achieve this by incorporating a spatial kernel for multiple instance learning approach, which we describe below.

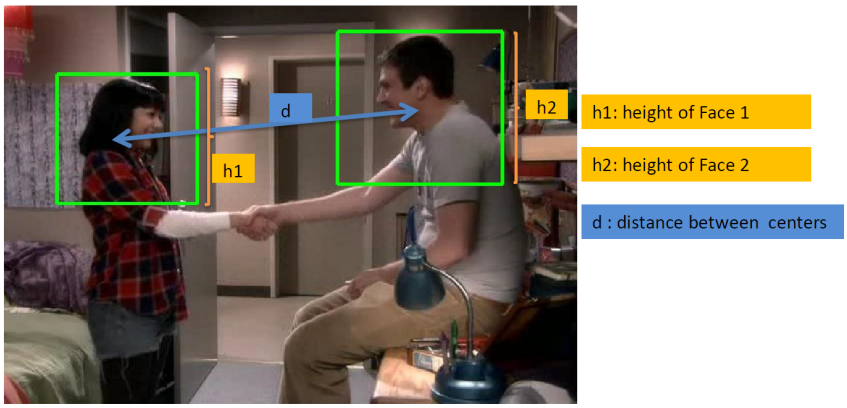


Figure 5.4: We extract relative distance information from both face and body regions and we example here for face regions. Relative distance of the first person to the second person is $d/h1$ and relative distance of the second person to the first person is $d/h2$. Where d is the distance between individuals, $h1$ and $h2$ are the height of first and second person's face region bounding box. After first and second person relative distances are extracted, they are concatenated to get final descriptors. `relative_body` and `relative_face` are two of our five descriptors

5.2 Multiple Instance Learning Approach

In this section, we describe the Multiple Instance Learning framework used and evaluate constructing suitable candidate bags (MI-bags) for interaction videos.

5.2.1 Bag Construction

We are particularly interested in the problem of recognizing interactions in unconstrained real world videos. In this problem we do not have the information that in which frames interaction occurs. The only label is interaction class label which provided for whole video sequence. Interactions occur in somewhere in sequence; however, beginning and ending frames are not clearly defined and there may be irrelevant actions in some frames. This case is particularly suitable for multi-instance representation. In multiple instance learning, a bag is required to contain at least one positive frame for the particular interaction. Since in interaction videos there are some frames (relevant frames) where target interaction occurs and also some frames (irrelevant frames) where absolutely no interesting

Figure 5.5: Example bag creation way for frames include two person with body bounding boxes, blue for no-interaction, and green for interaction. There are 4 sample frames from an interaction video selected to construct a bag. As it can be seen features extracted from first and second person regions are concatenated and added to the bag as an instance. At the end bag has two instances where interaction occurs which shown with red squares.

interactions performed by individuals. For this purpose, we form each video as a bag and each frame as an instance and then formulate multiple instance learning. In each positive bag, target interactions happens somewhere in bag however in negative bags any interaction doesn't happen. Figure 5.5 shows our multi-instance bag creating way. A number of frames from a video are selected and then from two body regions extracted features are combined and added to bag.

Videos obtained from real world sources are not so perfect. For example videos may contain multiple people and this may be a problem for our descriptor creating way because our formulation require two people in each frame. In this case an algorithm studying interaction recognition from frames may need to develop a complex solution however our algorithm successfully handle this situation with the benefits of multiple instance learning. According to this we order person region bounding boxes in x dimension and name them as first person, second person, third person and so on. We start with the leftmost person; accept this person as reference. Then we match this reference person with remaining ones stay in right-side respectively to create our two-person based descriptor. Then the second leftmost person becomes new reference person and so on. Figure 5.6 shows an example bag construction in case of frames include multiple people. Although

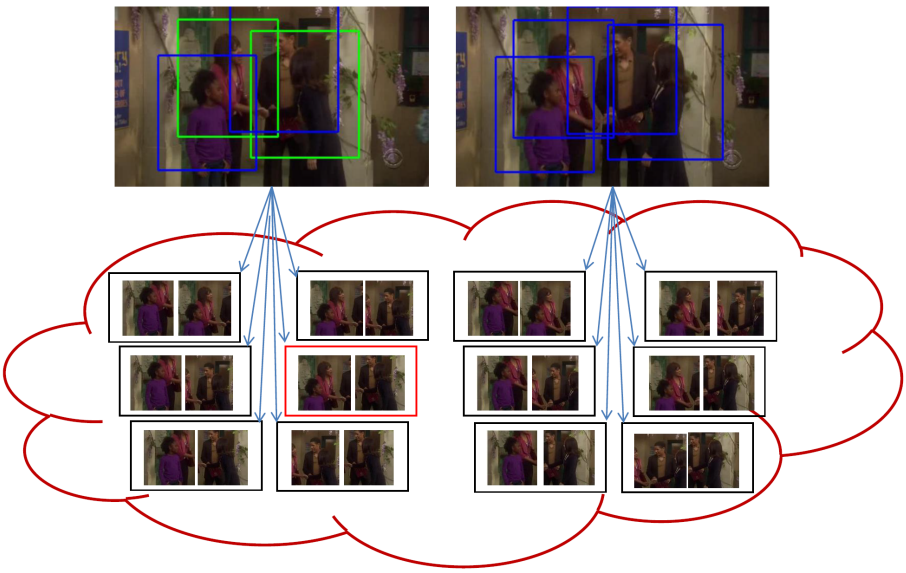


Figure 5.6: Example bag creation way for frames include multiple people with body bounding boxes, blue for no-interaction, and green for interaction. There are 2 sample frames from an interaction video selected to construct a bag. As it can be seen starting from leftmost person a match is done over other person regions stay in right-side. Then features extracted from each match regions are concatenated and added bag as an instance. At the end bag has one instance where interaction occurs which shown with red square. Multiple people in an interaction video cause many negative instances in interaction bags.

our solution is simple, there are disadvantages since number of irrelevant instances in some bags increases and recognition of these bags may perform poorly. The increase in instance number in bags increases computation time also.

Besides, typical object detectors may perform poorly and do not have accurate responses for locating certain people for some cases. In some frames there may be only one detected person. Since our descriptor needs two people regions we tackle this problem by taking average over related features of the corresponding interaction. We complete the missing person's features with these averaged values.

Additionally we study formation of test bags frame by frame and use this representation for in interaction recognition which we show in Figure 5.7. In training stage bag creation way doesn't change where a bag corresponds to a video however in testing each frame is considered as a bag. Scores are assigned

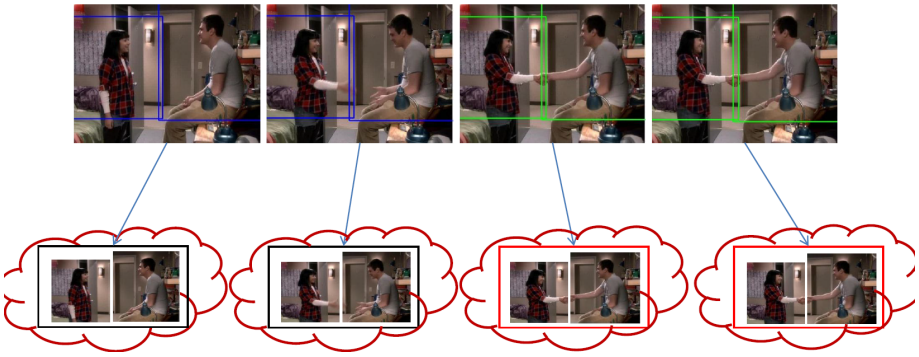


Figure 5.7: Features extracted from first and second person region are concatenated. For every match a bag is created. In last two bags include red squares interaction occurs.

to the frames for each interaction. Because of irrelevant frames and ambiguous poses, misclassications may be observed. To smooth these out, we average scores over frames of each video for each interaction and assign these values to videos. Finally it turns out an evaluation of interactions over frames we call it frame based evaluation.

5.2.2 MIL Classification and Spatial Embedding

In this study, we utilize Multiple Instance Learning with Instance Selection [32] (MILES) algorithm. Particularly, we propose a variant of the spMILES method proposed by Ikizler-Cinbis and Sclaroff [82] for the use with our human interaction framework.

First, we evaluate our features separately using standard definition discussed in Background and Related Work chapter. For five types of features we utilize MILES algorithm. Using the embedded representation provided by MILES, we train an L2-regularized SVM with RBF kernel for each feature type. In the end the scores from all classifiers are combined linearly using equal weights for the features and using weights determined via linear SVMs from training data.

Second as we discussed in 5.1 spatial relations of person regions can provide additional information to learn a good model. We add two multiplicative spatial

kernels to the feature-based similarity. We formulate it as follows:

$$s(c_l, \mathbf{B}_i) = \max_j \left(\phi_{feat}(x_{ij}, c_l) \phi_{spatial_x}(x_{ij}, c_l) \phi_{spatial_y}(x_{ij}, c_l) \right), \tag{5.1}$$

$\phi_{feat}(x_{ij}, c_l)$ is the similarity between feature vectors. $\phi_{spatial_x}(x_{ij}, c_l)$ is the spatial closeness between a concept instance c_l and a bag instance x_{ij} in x dimension and $\phi_{spatial_y}(x_{ij}, c_l)$ is the spatial closeness between a concept instance c_l and a bag instance x_{ij} in y dimension. Which are defined as follows;

$$\phi_{feat}(x_{ij}, c_l) = \exp\left(-\frac{D(x_{ij}, c_l)}{\sigma} \right), \tag{5.2}$$

$$\phi_{spatial_x}(x_{ij}, c_l) = \exp\left(-\frac{|d_x(p_1, O) - d_x(p_1\,', O')|\,|d_x(p_2, O) - d_x(p_2\,', O')|}{\sigma_x} \right), \tag{5.3}$$

$$\phi_{spatial_y}(x_{ij}, c_l) = \exp\left(-\frac{|d_y(p_1, O) - d_y(p_1\,', O')|\,|d_y(p_2, O) - d_y(p_2\,', O')|}{\sigma_y} \right), \tag{5.4}$$

where p_1 is first and p_2 is second person in bag instance x_{ij} and O corresponds the center of first person p_1 and second person p_2. $p_1\,'$ is first and $p_2\,'$ is second person in concept instance c_l and $O\,'$ corresponds the center of first person $p_1\,'$ and second person $p_2\,'$. $D(.)$ measures the similarity between a concept instance c_l and a bag instance x_{ij}. $d_x(.)$ measures the distance in x coordinate $d_y(.)$ in y coordinate. σ_x and σ_y are the bandwidth parameters to adjust the sensitivity of the measure to the spatial differences. We select σ , σ_x and σ_y parameters using cross-validation over the training set.

Eq. 5.1 allow us to consider similarity between feature vectors of two-person regions and relative distance of two-person in both x and y dimensions together. For $D(.)$, we used Euclidean distance. Both $d_x(.)$ and $d_y(.)$ are normalized with respect to the related person bounding box size.

Finally for each three types of features which include spatial information inside we use the embedded representation provided by MILES and train an L2-regularized SVM with RBF kernel. Combination of three features done linearly

using equal weights or using weights determined via linear SVMs training.

5.3 Experiments

5.3.1 Datasets and Experimental Setup

In order to evaluate the performance of our method, we use realistic"TV Interactions" dataset collected by Patron-Perez el.al. [4] . This dataset consist of 300 videos in total extracted from different TV show. The dataset contains four interactions: "hand shake", "high five", "hug" and "kiss" (each appearing in 50 videos) and negative examples (100 videos) which don't contain any of four interactions. It is a quite challenging dataset with lots of camera viewpoint, scale of interaction, different viewing directions. The length of the video clips ranges changes between 30 and 600 frames. For every frame the upper body bounding boxes, discrete head orientation and interaction label for each person are provided. For each category of interaction, dataset is divided into 2 related subsets, we apply leave-one-out cross validation over these subsets, following the same evaluation methodology of [4].

In our experiments, we use bounding boxes provided by dataset in extracting our features, both in the train and test phases. We select a subset of frames for each video, $n = 30$ by uniform selection we assume that at least one frame include target interaction is selected for each video.

5.3.2 Evaluation

In this section we first provide a detailed evaluation of individual features then look into the effect of spatial embedding. Finally we evaluate negative videos.

5.3.2.1 Performance of Individual Features

We first evaluate the performance of the individual features. Average Precision (mAP) values achieved by using individual features are shown in Table 5.1 and Table 5.2. For this evaluation we don't use negative videos. We consider only videos contain an interaction.

The first five rows of Table 5.1 include the individual recognition performance of video based evaluation for each of the feature type. Where each video corresponds to a bag therefore a bag contains multiple frames. Note that in these experiments we do not use any embedded spatial information. In contrast, we use spatial relations between two-person as additional separate features. The results show that using HoG features over face regions `hog_face` has the best performance among others and followed by HoG features over body regions `hog_body`. This observation shows that shape features are very informative to understand the type of the interactions. Although for "hand shake" interaction `hog_body` has the best performance among others. For other three interaction `hog_face` provides the best performances. This is not surprising, since these interactions occur closer to the face area. On the other hand performance of `hog_body` is not as good as performance of `hog_face`, because of its extended region bounding box it is affected by background noise more. Only "hand shake" interaction shows the best representation with HoG features over body region. Motion-based features are also informative for some interactions. "high five" interaction is a relatively quicker one than the others and `hof_body` gives the best performance for this interaction. As it can be seen, relative distance features have also good performance on interactions. This demonstrates that spatial locations of two-person can provide useful information and encourage us to embed this information to our multiple instance learning framework.

The last two rows of Table 5.1 represent the performance of linear combination of individual features . We manage this combination in two ways first with equal weights and second with weights provided by SVMs over training scores of each feature. Two of these combinations type do not perform big differences. However surprisingly with both combination ways "high five" interaction shows a great

Table 5.1: Average Precision values of video based evaluation method for 5 types of features individually. Negative video data is not included. The first five rows shows performance for our five feature type. The last two rows represent result of linear combination of these features.

method	hs	hf	h	k	mean
hog_body	63.08	52.06	62.24	68.64	64.32
hog_face	58.11	63.60	74.13	75.97	**66.15**
hof_body	59.90	63.99	49.64	49.25	59.47
relative_body	55.73	53.54	66.53	56.92	61.39
relative_face	54.40	54.01	67.73	61.37	62.50
equal weights	62.52	74.92	81.30	73.06	**72.95**
svm weights	65.46	73.48	76.28	75.61	72.71

increase.

The first five rows of Table 5.2 include the individual recognition performance of frame based evaluation for each of the feature type . Where in testing stage we treat each frame as a bag then a score is assigned to each frame. Score of an interaction video is determined by assigning average score obtained over frames for each interaction. We observe similar performance patterns among individual features for frame based evaluation and video based evaluation. However there is a general decrease in AP values for many feature types. Especially `hof_body` perform relatively poor for "high five" interaction. Combination of all features also shows a big decrease for this interaction type. The general decrease may be caused by scores of irrelevant frames. If so the decrease in "high five" interaction is not surprising since during frame selection we observed that subset of frames for "high five" videos include less positive frames than other interactions because this interaction is faster than the others and total number of frames include target interaction is already low. These results indicate that frame based evaluation shows worse performance than video based evaluation because negative effect of irrelevant frames cannot be boosted as successful as video based evaluation.

In Figure 5.8 some qualitative examples for highest ranked true and false positives are given for features over body regions `hog_body`, `hof_body` and `relative_body` with video based evaluation . It can be observed that among true positives our features assign highest top 3 ranks to true interactions. We

Table 5.2: Average Precision values of frame based evaluation method for 5 types of features individually. Negative video data is not included. The first five rows shows performance for our five feature type. The last two rows represent result of linear combination of these features.

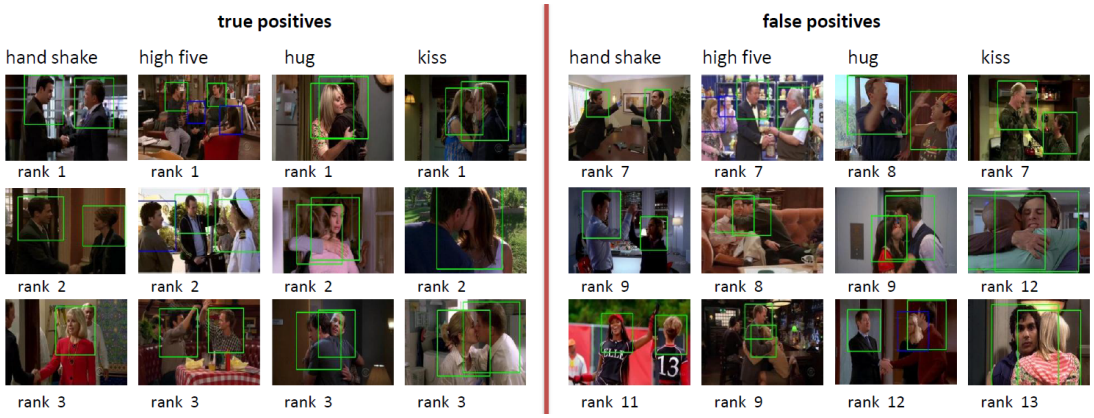
method	hs	hf	h	k	mean
hog_body	63.40	56.15	57.77	64.86	59.81
hog_face	57.40	64.82	72.78	73.52	**61.97**
hof_body	51.26	53.64	47.77	40.12	53.62
relative_body	55.45	53.51	63.33	55.70	57.52
relative_face	55.75	53.62	60.27	59.35	58.43
equal weights	62.11	58.64	75.96	70.21	66.73
svm weights	64.87	63.95	71.49	68.87	**67.29**

observe that "hand shake videos are confused with "high five" videos for all three feature types. In the same way "hug" and "kiss" interactions tend to confused to each other. This could be because this interaction pairs more close to each other than others in terms of shape and relative distance between two-person. Surprisingly highest top 3 ranks are also assigned to videos include multiple people.

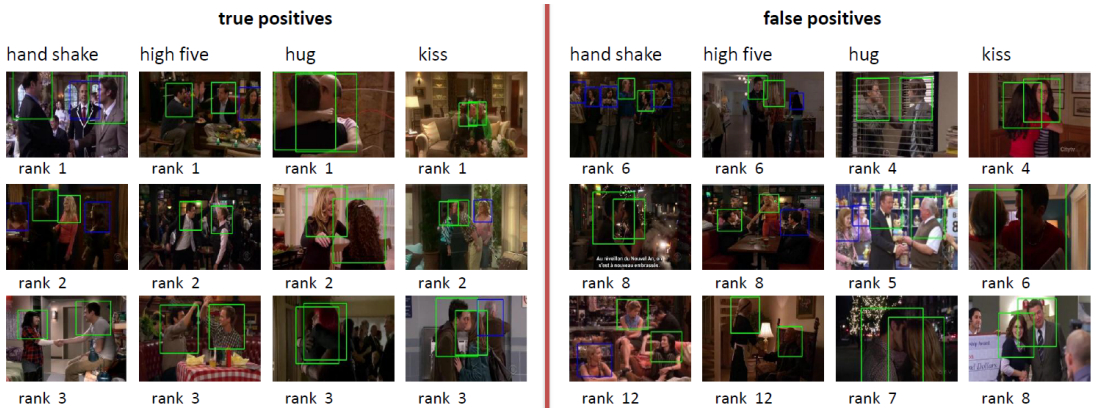
5.3.2.2 Performance of Spatial Embedding

We modified multiple instance learning approach MILES by using spatial kernels in addition to similarity between feature vectors. As expected combining spatial relations of two-person to feature vectors provide additional information. However with combination of spatial features surprisingly we get better results than combination of individual features.

The first three rows of Table 5.3 include the individual recognition performance for each of three feature type with spatial relations of two-person. These result obtained via video based evaluation. Shape features provide better performance than motion features, however adding spatial information increase the performance of all three feature type. HoG features over face regions `hog_face_spatial` has the best performance among others. Especially for "hug" and "kiss" interactions spatial information increase the performance noticeable.



(a) `hog_body`



(b) `hof_body`

(c) `relative_body`

Figure 5.8: Highest ranked true and false positives for `hog_body`, `hof_body` and `relative_body` features. Ordering is done based one Average Precision values obtained from video based evaluation. Negative video data is not included.

Table 5.3: Average Precision values of video based evaluation method for three feature types with embedded spatial relations. Negative video data is not included. The first three rows show the performance of feature types with spatial relations. The last two rows represent result of linear combination of these features.

method	hs	hf	h	k	mean
hog_body_spatial	66.90	69.99	77.50	74.48	72.22
hog_face_spatial	63.57	66.47	87.06	84.18	**75.32**
hof_body_spatial	67.72	69.61	69.62	55.60	65.64
equal weights	70.25	70.29	84.18	80.25	**76.24**
svm weights	68.57	70.03	83.68	80.13	75.60

Table 5.4: Average Precision values of frame based evaluation method for three feature types with embedded spatial relations. Negative video data is not included. The first three rows show the performance of feature types with spatial relations. The last two rows represent result of linear combination of these features

method	hs	hf	h	k	mean
hog_body_spatial	69.13	65.09	79.22	74.96	72.10
hog_face_spatial	69.98	64.04	83.75	77.46	**73.81**
hof_body_spatial	60.79	63.10	65.02	52.55	60.36
equal weights	74.83	68.24	82.40	73.38	**74.71**
svm weights	68.93	67.53	81.69	74.40	73.14

Then we look at the overall combination results of spatial features. The combination of all the features gives the best performance among all our evaluations.

The first three rows of Table 5.3 include the individual recognition performance for each of three feature type with spatial relations of two-person. These result obtained via frame based evaluation. Adding spatial information increases performance of the features. Frame based evaluations still show worse performance than video based evaluation in case we use spatial information. However we can say that by adding spatial information frame based recognition becomes more accurate.

Table 5.5: Average Precision values of video based evaluation method for 5 feature types individually. Negative video data is also included to this evalution. The first five rows shows performance for our five feature types. The last two rows represent result of linear combination of these features.

method	hs	hf	h	k	mean
hog_body	47.42	42.24	61.95	60.91	52.03
hog_face	38.68	56.50	71.91	70.74	**54.49**
hof_body	42.33	43.25	47.65	44.01	47.80
relative_body	34.72	33.65	65.27	54.55	50.44
relative_face	30.97	38.82	65.72	57.61	51.20
equal weights	46.87	60.21	83.03	74.08	66.05
svm weights	49.65	57.63	82.39	77.35	**66.76**

5.3.2.3 Performance with Negative Video Data

We now evaluate the performance of the individual features for evaluations when negative videos used also. Here we show video based results. The first five rows of Table 5.5 include the individual classification performance for our five feature types. With the increase of irrelevant videos Average Precision decreases. HoG features over face regions `hog_face` has the best performance among others and combining features increases overall performance. Especially for "hug" and "kiss" interactions interestingly overall performance is good in case we use negative videos. For "hand shake" interaction performance is very bad. This could be because this interaction has similar features with negative videos and confused to each other. In Table 5.6 we present the result of spatial embedding of relative distances to motion and shape features. Here we present the video based evaluation performance. Spatial embedding increases the performance for both individuals and combination of features.

5.3.3 Comparison to state-of-the-art

We compare our best performance with the state of the art method in Table 5.7. We used manual annotations provided with dataset and we compare our methods performance with values of Patron-Perez el.al. [4] which are the evaluation values of their method for manual annotations. In this study Patron-Perez el.al. [4]

Table 5.6: Average Precision values of video based evaluation method for 3 types of features individually with spatial information. Negative video data is also included for this evalution. The first three rows shows the performance of our three feature type. The last two rows represent result of linear combination of these features.

method	hs	hf	h	k	mean
hog_body	49.81	52.29	83.51	66.10	62.93
hog_face	41.87	53.33	86.92	81.03	**65.79**
hof_body	53.61	48.99	63.75	42.86	52.30
equal weights	53.58	57.60	86.64	70.66	**67.12**
svm weights	50.83	56.96	85.77	73.69	66.81

Table 5.7: Comparison to the state-of-the art on "TV Interactions" dataset. In this table, Average Precision values are reported. We present our method's video based performance with spatial relations for both negative video data included and not included case.

method	hs	hf	h	k	mean
Patron-Perez et.al. [4]	57.83	51.08	71.16	76.54	64.15
ours = video based + spatial	70.25	70.29	84.18	80.25	**76.24**
Patron-Perez et.al. [4] + Neg.	45.30	45.07	62.00	70.58	55.74
ours = video based + spatial + Neg.	53.58	57.60	86.64	70.66	**67.12**

introduce a person-centered descriptor. Their work relies on detection of upper-body and estimation of head orientation. They also use structured learning to capture spatial relations between person regions.

Looking at Table 5.7 our method provides an increase 18% in mean average precision compared to [4] for the case no negative videos are used. And for all types of interactions provide better performance. For the negative video added case our method provides an increase 20% in mean average precision compared to [4] and again provide better performance for all interactions than [4] .

Chapter 6

Conclusion

In this thesis we introduce novel solutions for three computer vision problems; image retrieval, recognizing actions from still images, interaction recognition from videos by successfully applying Multiple Instance Learning

Image retrieval: In Image Retrieval, we proposed a simple yet effective approach based on multiple instance learning for the problem of image re-ranking. Our approach relies on the construction of multiple candidate MI-bags based on the retrieval order of the images. Assuming that the initial retrieval list contains images of interest, our approach constructs multiple bags and learns multiple MI-classifiers over these bags. Then, the images are reranked based on the decision scores of the resulting ensemble of MI-classifiers. Our approach is shown to perform quite successfully compared to the state-of-the-art and significantly outperforms the initial ranking list of produced by the search engines.

Our approach do not make use of any explicit feedback, or auxiliary data such as surrounding text or additional training data. The presented method only relies on the visual content of the retrieved images. Given the simplicity of the approach, it can easily be incorporated to more sophisticated schemes, where more complex learning algorithms or more complex visual features are utilized. Considering additional modalities of data can also be explored as a

future direction.

Recognizing actions from still images: In this study, we have proposed a method that leverages the candidate object regions in a weakly unsupervised manner via Multiple Instance Learning and evaluated the performance of this method in combination with other visual features for human action recognition in still images. Our experimental results show that the proposed MIL framework is suitable for extracting the relevant object information, without the need for explicit object detectors. We have achieved better classification performance compared to the state-of-the-art on the extensive Stanford 40 actions still image dataset.

Our findings indicate possible future directions, particularly, using richer representations over salient object regions and improving weakly supervised learning of relevant objects.

Interaction recognition from videos: In this study we presented a multiple instance learning (MIL) based approach for two-person interactions recognition for unconstraint daily videos. We propose a method with multiple features extracted from person regions and we form our descriptor as a concatenation of person regions in frames. Then we form each frame as a bag instance and each video as a bag.

We extended our approach using spatial relations of person regions by adding spatial kernels to MIL framework. Using spatial information embedded to other features, we show that better performance is possible.

The results demonstrate that proposed approach offers considerable improvement over two-person interaction recognition performance. We have achieved better recognition performance compared to the state-of-the-art on the "TV Interactions" dataset. Our results are promising on automatic annotations provided with dataset and our feature directions include to study with automatic annotations.

In standard supervised learning, learning procedure works over instances however in multiple instance learning inputs of learning procedure are bags of instances. Multiple instance learning problems involve ambiguous training examples. Label information available for the bags but not necessarily for the instances. In many computer vision problems obtaining training instance labels is a serious problem due to nature of images and use of weakly-labeled data is an attractive solution for these problems. So multiple instance learning may be useful for some computer vision problems. Although it seems to be useful, designing of bags and instances should be done in a reasonable way. We have presented three novel approaches for computer vision problems using multiple-instance learning. From performance of our solutions, we can say that we have successfully formed multiple instance paradigm to our problems.

Bibliography

[1] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artif. Intell.*, vol. 89, pp. 31–71, Jan. 1997.

[2] J. Krapac, M. Allan, J. Verbeek, and F. Jurie, "Improving web-image search results using query-relative classifiers," in *IEEE Conference on Computer Vision & Pattern Recognition (CVPR '10)*, (San Francisco, United States), pp. 1094–1101, IEEE Computer Society, 2010.

[3] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. J. Guibas, and L. Fei-Fei, "Human action recognition by learning bases of action attributes and parts," in *International Conference on Computer Vision (ICCV)*, (Barcelona, Spain), November 2011.

[4] A. Patron-Perez, M. Marszalek, A. Zisserman, and I. Reid, "High Five: Recognising human interactions in TV shows," in *British Machine Vision Conference*, 2010.

[5] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman, "Learning object categories from google's image search.," in *Proceedings of the 10th International Conference on Computer Vision, Beijing, China*, vol. 2, pp. 1816–1823, Oct. 2005.

[6] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?," in *IEEE Conf. on Computer Vision and Pattern Recognition*, (San Francisco, USA), 2010.

[7] X. Tian and D. Tao, "Visual reranking:from objectives to strategies," *IEEE Multimedia*, vol. 18, pp. 12–21, March 2011.

[8] A. Gupta, A. Kembhavi, and L. S. Davis, "Observing human-object interactions: Using spatial and functional compatibility for recognition," *TPAMI*, vol. 31, pp. 1775–1789, 2009.

[9] B. Yao and L. Fei-Fei, "Modeling mutual context of object and human pose in human-object interaction activities," in *CVPR*, (San Francisco, CA), June 2010.

[10] A. Prest, C. Schmid, and V. Ferrari, "Weakly supervised learning of interactions between humans and objects," *IEEE TPAMI*, vol. 34, pp. 601–614, 2012.

[11] P. Viola, J. C. Platt, and C. Zhang, "Multiple instance boosting for object detection," in *In NIPS 18*, pp. 1419–1426, MIT Press, 2006.

[12] P. Dollár, B. Babenko, S. Belongie, P. Perona, and Z. Tu, "Multiple component learning for object detection," in *ECCV*, 2008.

[13] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *CVPR*, 2009.

[14] B. Zeisl, C. Leistner, A. Saffari, and H. Bischof, "On-line semi-supervised multiple-instance boosting," in *Proceedings of the British Machine Vision Conference*, June 2010.

[15] Z.-J. Zha, X.-S. Hua, T. Mei, J. Wang, G.-J. Qi, and Z. Wang, "Joint multi-label multi-instance learning for image classification," in *CVPR*, pp. 1–8, June 2008.

[16] O. Y. an Vasant Honavar, "Multi-instance multi-label learning for image classification with large vocabularies," in *Proceedings of the British Machine Vision Conference*, pp. 59.1–59.12, 2011.

[17] O. Maron and A. L. Ratan, "Multiple-instance learning for natural scene classification," in *ICML*, pp. 341–349, 1998.

[18] O. Maron and T. Lozano-Prez, "A framework for multiple-instance learning," in *Advances In Neural Information Processing Systems*, pp. 570–576, 1998.

[19] Q. Zhang and S. A. Goldman, "Em-dd: An improved multiple-instance learning technique," in *In Advances in Neural Information Processing Systems*, pp. 1073–1080, 2001.

[20] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *NIPS*, pp. 561–568, MIT Press, 2003.

[21] J. Wang and J.-D. Zucker, "Solving the multiple-instance problem: A lazy learning approach," in *In Proc. 17th International Conf. on Machine Learning*, pp. 1119–1125, 2000.

[22] S. Andrews and T. Hofmann, "Multiple instance learning via disjunctive programming boosting," in *In Advances in Neural Information Processing Systems (NIPS*16)*, pp. 65–72, 2004.

[23] J. Ramon and L. De Raedt, "Multi instance neural networks," in *Proceedings of the ICML-2000 workshop on attribute-value and relational learning*, pp. 53–60, 2000.

[24] Z. H. Zhou and M. L. Zhang, "Neural networks for multi-instance learning," tech. rep., In: Proceedings of the International Conference on Intelligent Information Technology, 2002.

[25] Y. Chevaleyre and J.-D. Zucker, "Solving multiple-instance and multiple-part learning problems with decision trees and rule sets. application to the mutagenesis problem," in *Lecture Notes in Artificial Intelligence*, vol. 2056, pp. 204–214, 2001.

[26] H. Blockeel, D. Page, and A. Srinivasan, "Multi-instance tree learning," in *Proceedings of the 22nd international conference on Machine learning*, ICML '05, (New York, NY, USA), pp. 57–64, ACM, 2005.

[27] T. Deselaers and V. Ferrari, "A conditional random field for multiple-instance learning," in *ICML*, June 2010.

[28] C. Leistner, A. Saffari, and H. Bischof, "Miforests: Multiple instance learning with randomized trees," in *11th European Conference on Computer Vision*, 2010.

[29] Z. hua Zhou, Y. yin Sun, and Y. feng Li, "Multi-instance learning by treating instances as noni.i.d. samples," in *In Proceedings of the 26th International Conference on Machine Learning*, 2009.

[30] J. R. Foulds and E. Frank, "A review of multi-instance learning assumptions," *Knowledge Eng. Review*, vol. 25, no. 1, pp. 1–25, 2010.

[31] Y. Chen and J. Z. Wang, "Image categorization by learning and reasoning with regions," *J. Mach. Learn. Res.*, vol. 5, pp. 913–939, Dec. 2004.

[32] Y. Chen, J. Bi, and J. Z. Wang, "Miles: Multiple-instance learning via embedded instance selection," *IEEE TPAMI*, vol. 28, pp. 1931–1947, 2006.

[33] Z. Fu, A. Robles-Kelly, and J. Zhou, "Milis: Multiple instance learning with instance selection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 5, pp. 958–977, 2011.

[34] J. L. Ritendra Datta, Dhiraj Joshi and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Computing Surveys*, vol. 40, no. 2, pp. 1–60, 2008.

[35] M. Fritz and B. Schiele, "Decomposition, discovery and detection of visual categories using topic models," in *IEEE Conference on Computer Vision & Pattern Recognition (CVPR)*, (Anchorage, United States), IEEE Computer Society, 2008.

[36] W. Hsu, L. Kennedy, and S.-F. Chang, "Reranking methods for visual search," *IEEE MultiMedia*, vol. 14, no. 3, pp. 14–22, 2007.

[37] Y. Jing and S. Baluja, "Visualrank: Applying pagerank to large-scale image search," *IEEE Transactions on Pattern Recognition and Machine Intelligence(TPAMI)*, vol. 30, no. 11, pp. 1877–1890, 2008.

[38] W. Liu, Y.-G. Jiang, J. Luo, and S.-F. Chang, "Noise resistant graph ranking for improved web image search," in *IEEE Conference on Computer Vision & Pattern Recognition (CVPR)*, (Colorado Springs, United States), IEEE Computer Society, 2011.

[39] T. Berg and D. A. Forsyth, "Animals on the web," in *CVPR*, 2006.

[40] D. Grangier and S. Bengio, "A discriminative kernel-based model to rank images from text queries," *IEEE Transactions on Pattern Recognition and Machine Intelligence(TPAMI)*, vol. 30, no. 8, pp. 1371 –1384, 2008.

[41] F. Schroff, A. Criminisi, and A. Zisserman, "Harvesting image databases from the web," in *ICCV*, pp. 1–8, oct. 2007.

[42] B. Geng, L. Yang, C. Xu, and X.-S. Hua, "Content-aware ranking for visual search," in *CVPR*, 2010.

[43] W. Li, L. Duan, D. Xu, and I. W. Tsang, "Text-based image retrieval using progressive multi-instance learning," in *ICCV*, 2011.

[44] L. Duan, W. Li, I. W. Tsang, and D. Xu, "Improving web image search by bag-based re-ranking," *IEEE Trans. on Image Processing (T-IP)*, pp. 3280–3290, 2011.

[45] R. Poppe, "A survey on vision-based human action recognition," *Image Vision Computing*, vol. 28, pp. 976–990, June 2010.

[46] D. Weinland, R. Ronfard, and E. Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," *CVIU*, vol. 115, pp. 224–241, February 2011.

[47] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *CVPR*, 2008.

[48] Y. Wang, H. Jiang, M. S. Drew, Z.-N. Li, and G. Mori, "Unsupervised discovery of action classes," in *CVPR*, 2006.

[49] C. Thurau and V. Hlavac, "Pose primitive based human action recognition in videos or still images," in *CVPR*, 2008.

[50] N. Ikizler-Cinbis, R. G. Cinbis, and S. Sclaroff, "Learning actions from the web," in *Int. Conf. on Computer Vision*, 2009.

[51] B. Yao and L. Fei-Fei, "Grouplet: a structured image representation for recognizing human and object interactions," in *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition*, (San Francisco, CA), June 2010.

[52] C. Desai, D. Ramanan, and C. Fowlkes, "Discriminative models for static human-object interactions," in *Workshop on Structured Models in Computer Vision*, 2010.

[53] V. Delaitre, J. Sivic, and I. Laptev, "Learning person-object interactions for action recognition in still images," in *NIPS*, 2011.

[54] V. Delaitre, I. Laptev, and J. Sivic, "Recognizing human actions in still images: a study of bag-of-features and part-based representations," in *BMVC*, 2010.

[55] B. Yao, A. Khosla, and L. Fei-Fei, "Combining randomization and discrimination for fine-grained image categorization," in *CVPR*, (Springs, USA), June 2011.

[56] A. Datta, M. Shah, N. Da, and N. D. V. Lobo, "Person-on-person violence detection in video data," in *In Proc. Intl Conference on Pattern Recognition*, pp. 433–438, 2002.

[57] S. Park and J. K. Aggarwal, "Simultaneous tracking of multiple body parts of interacting persons," *Comput. Vis. Image Underst.*, vol. 102, pp. 1–21, Apr. 2006.

[58] M. S. Ryoo and J. Aggarwal, "Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities," in *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 1593–1600, 2009.

[59] A. Fathi, J. K. Hodgins, and J. M. Rehg, "Social interactions: A first-person perspective," in *CVPR*, pp. 1226–1233, 2012.

[60] T. Lan, L. Sigal, and G. Mori, "Social roles in hierarchical models for human activity recognition," in *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[61] A. Gaidon, Z. Harchaoui, and C. Schmid, "Recognizing activities with cluster-trees of tracklets," in *BMVC*, (Guildford, Royaume-Uni), Sept. 2012.

[62] Y. Yang, S. Baker, A. Kannan, and D. Ramanan, "Recognizing proxemics in personal photos," in *CVPR*, pp. 3522–3529, 2012.

[63] M. A. Sadeghi and A. Farhadi, "Recognition using visual phrases," 2011.

[64] S. Ali and M. Shah, "Human action recognition in videos using kinematic features and multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 288–303, 2010.

[65] N. Ikizler-Cinbis and S. Sclaroff, "Object, scene and actions: combining multiple features for human action recognition," in *Proceedings of the 11th European conference on Computer vision: Part I*, ECCV'10, (Berlin, Heidelberg), pp. 494–507, Springer-Verlag, 2010.

[66] T. Leung, Y. Song, and J. Zhang, "Handling label noise in video classification via multiple instance learning," in *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, (Washington, DC, USA), pp. 2056–2063, IEEE Computer Society, 2011.

[67] K. Prabhakar and J. M. Rehg, "Categorizing turn-taking interactions," in *ECCV (5)*, pp. 383–396, 2012.

[68] K. Yun, J. Honorio, D. Chattopadhyay, T. Berg, and D. Samaras, "Two-person interaction detection using body-pose features and multiple instance learning," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pp. 28–35, 2012.

[69] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91–110, Nov. 2004.

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

[71] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[72] S. Vijayanarasimhan and K. Grauman, "Keywords to visual categories: Multiple-instance learning for weakly supervised object categorization," in *CVPR*, 2008.

[73] N. Morioka and J. Wang, "Robust visual reranking via sparsity and ranking constraints," in *Proceedings of the 19th ACM international conference on Multimedia*, MM '11, pp. 533–542, 2011.

[74] A. Patron-Perez, M. Marszalek, I. Reid, and A. Zisserman, "High five: Recognising human interactions in tv shows," in *British Machine Vision Conference*, 2010.

[75] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *CVPR*, 2001.

[76] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91–110, 2004.

[77] L. Bourdev and J. Malik, "Poselets: Body part detectors trained using 3d human pose annotations," in *ICCV*, 2009.

[78] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari, "2d articulated human pose estimation and retrieval in (almost) unconstrained still images," *International Journal of Computer Vision*, vol. 99, pp. 190–214, 2012.

[79] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models.," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, 2010.

[80] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886–893 vol. 1, 2005.

[81] N. Ikizler, R. G. Cinbis, and P. Duygulu, "Human action recognition with line and flow histograms," in *In Proc. ICPR*, 2008.

[82] N. Ikizler-Cinbis and S. Sclaroff, "Object recognition and localization via spatial instance embedding," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, pp. 452–455, 2010.