University of Genova
Istituto Italiano di Tecnologia

Data Driven Approaches for Image & Video Understanding: from Traditional to Zero-shot Supervised Learning

Abhinaba Roy

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Computing of the University of Genova, November 2018

Abstract

In the present age of advanced computer vision, the necessity of (user-annotated) data is a key factor in image & video understanding. Recent success of deep learning on large scale data has only acted as a catalyst. There are certain problems that exist in this regard: 1) scarcity of (annotated) data, 2) need of expensive manual annotation, 3) problem of change in domain, 4) knowledge base not exhaustive. To make efficient learning systems, one has to be prepared to deal with such diverse set of problems. In terms of data availability, extensive manual annotation can be beneficial in obtaining category specific knowledge. Even then, learning efficient representation for the related task is challenging and requires special attention. On the other hand, when labelled data is scarce, learning category specific representation itself becomes challenging. In this work, I investigate data driven approaches that cater to traditional supervised learning setup as well as an extreme case of data scarcity where no data from test classes are available during training, known as zero-shot learning. First, I look into supervised learning setup with ample annotations and propose efficient dictionary learning technique for better learning of data representation for the task of action classification in images & videos. Then I propose robust mid-level feature representations for action videos that are equally effective in traditional supervised learning as well as zero-shot learning. Finally, I come up with novel approach that cater to zero-shot learning specifically. Thorough discussions followed by experimental validations establish the worth of these novel techniques in solving computer vision related tasks under varying data-dependent scenarios.

Acknowledgements

I would like to express gratitude to Professor Dr. Vittorio Murino, Dr. Biplab Banerjee, Dr. Jacopo Cavazza, Dr. Cigdem Beyan, my friends at PAVIS and lastly my parents for their constant support in this journey.

‘Imagination is more important than knowledge. Knowledge is limited. Imagination encircles the world.’

Albert Einstein

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivations and Objectives

Data driven approach [LCHL07, RPE$^+$05, BMB$^+$13] is defined as the practise of inspecting upon data first and then choose the best strategies that suits the requirement(s). By analyzing the current and past data from different (re)sources, researchers as well as private companies can get a sense of the magnitude and size of the problem in hand. This, in turn, help them understand the problem in hand in a clearer way and consequently make accurate decisions. Many research approaches [FFFP06, FF$^+$03] as well as applications still are bound by the constraints of scarcity of data build their models taking such disadvantages into account. Even in cases with abundantly available data, data scenarios can change quickly and render a model was built upon can quickly turn unreliable in making a decision. A data driven approach essentially takes into account these changing scenarios, provides options to consider ever-changing nature of data and thus providing more accurate analysis. In an industrial setup, data also impacts marketing immensely and is fundamental to business strategy, advertising, and sales. It helps with management and operation on data, allowing researchers to operate faster and more efficiently, ultimately positively impacting final outcome.

In the current age of deep learning based approaches, supervised training with large data has proven to be promising in problems such as classification, regression, prediction etc. Notable scenarios that can arise based on the availability of data are as below:

1. The best possible situation in supervised learning setup is when data during the training phase spans equally over all the possible classes and all data labels are verified by humans. On the other hand, situations where data is scarce, training becomes difficult.

2. In extreme case scenarios only a few data is available for training, known as few-shot learning. There are situations when only a single image is available for training. This is known in research community as one-shot learning. Possibly, the most difficult case is when there no data from test classes are available during training phase, a problem known as zero-shot learning (ZSL).

These starkly different situations demand closer inspection of the data and intelligently designed approaches that cater to such conditions and constraints. In this thesis, I particularly inspect two opposite scenarios in data dependent approaches 1) When ample labeled data is available for training: traditional supervised learning. 2) No training data is available for the test classes: zero-shot learning. THe novelty of this work lies in the fact that I discuss thoroughly the necessity of investigating into data resources first and then make decision on how to train a system. At the same time, I propose novel approaches in the two situations that fit the requirements based on data availability.

In a traditional computer vision problem such as recognition of object/action categories, the training procedure focuses on extraction of relevant features from available data of each categories. Primary goal is to obtain compact and relevant ”label information”. One of the most prominent ways to achieve this is codebook generation. A codebook (dictionary) is a set of vectors that span a new/sub-space of the original data-space. This eventually provides a ”compact” as well as informative version of the original data. A standard supervised learning framework that incorporates codebook generation, follows the three standard stages: 1) Extraction of local descriptors, 2) Codebook (dictionary) generation and feature encoding, and 3) Classification based on the encoded features. Although success of such a framework depends upon a number of factors, and effective codebook generation is undoubtedly the most noteworthy. Especially if labeled data is available in abundance, a codebook is detrimental to the quality of category specific representation and thus the consecutive computer vision related task such as classification.

A standard codebook generation process is based on vector quantization of local descriptors extracted from the available training data in which the cluster centroids define the codewords; the basic building blocks that are ultimately used to encode the underlying visual entities. Needless to mention, the quality of the extracted local descriptors affect representation power of the codewords which, in turn, has direct impact on the recognition performance. For instance, the descriptors extracted from background regions or the ones shared by many visual categories add little to the discriminative capability of the codebook in comparison to the ones specifically extracted from the objects of interest. However, it is impossible to ensure the selection of potentially useful local descriptors in advance since such feature extraction techniques are typically engineered and ad hoc. In other words, there are certain immediate advantages if the most discriminative local descriptors are used for the purpose of a cogent codebook construction, though the process is intrinsically complex in general. Selection of the discriminative local descriptors for effective codebook generation is dealt in the first part of this thesis work. We take up action recognition from images and video as the field of application. We come up with a simple algorithm which gradually filters out unrepresentative descriptors before constructing a compact global codebook. The proposed method is generic in the sense that it can work with different types of local features irrespective of the underlying visual entities they refer to. For example, we represent each still image by a large pool of category independent region proposals [ADF12]. Each region proposal is represented by convolutional neural network (CNN) features (4096 dimensions) obtained from a pre-trained network. More specifically, we propose a sequential method for codebook construction which first clusters the local descriptors of each entity using the non-parametric mean-shift (MS) technique [CM02]. The cluster centroids thus obtained represent the reduced set of non-repetitive local features. Another round of MS clustering on the new set of local descriptors calculated from all the entities of a given category is followed and the centroids thus obtained are employed to build a temporary codebook specific to each category. Further, we propose an adaptive ranking criteria to highlight potentially discriminative codewords from each category specific codebook and the global dictionary is built by accumulating these reduced set of codewords from all the categories. We argue that our codebook construction technique explicitly incorporate the class support and a novel notion of distinctiveness based on conditional entropy is introduced.

Intelligently designed codebooks may be well suited for supervised multi-class classification problems but they do not necessarily solve all the problems present in a large scale dataset.

Even with availability of labeled data, action recognition is a difficult problem at its core when dealing with realistic videos (UCF-101, Youtube, Olympics, etc.) [SZS12]. This is primarily due to the large intra-class variations within an action category and the fine-grained nature of several action classes, besides other issues derived by background clutter, occlusions, illumination variations, and changes of camera viewpoint. This leads to the need for robust video representations capable of addressing the aforementioned problems adequately. Currently, available video-level features are broadly assorted into two groups: ad-hoc hand-crafted and deep features [HHP16]. A large set of hand-crafted features have been proposed in literature to date, which abstracts local spatio-temporal variations in the video content. These features include space-time interest points (STIP) [Lap05], cuboids [WXDL11], dense and improved trajectories [WKSL13]. Methods making use of such features usually consist of two stages: feature detection in order to highlight local interest points followed by the generation of descriptors characterizing the regions surrounding them. Although such features have been successfully applied for action recognition [WTVG08] [WKSL13], they are not fundamentally optimized as visual discriminants and may lack distinctive capability. On the contrary, deep learning based models (convolutional neural networks [KSH12a], recurrent networks [DWW15]), which are discriminatively trained from a large volume of labelled data, learn from a very basic to high level feature hierarchies automatically. Deep models proposed in this context include 3D-CNN [KTS$^+$14] and two-stream networks [SZ14a], which have proved to analyse video data successfully. However, they require a large amount of labelled videos for training, and most of the available datasets are relatively small. In addition, most of the current CNN-based models for video analysis largely overlook the intrinsic differences between the static and dynamic aspects of the videos. There are some endeavours to effectively combine both the facets in an unified framework [WQT15], but still they require a huge training time for end to end learning, which further affects the scalability of such features for real-time video analysis.

In this context, it is evident that: 1) It is indeed important to encode local scene characteristics (static and dynamic) from videos given the peculiar fine-grained nature of several action classes; 2) unlike the hand-crafted features, deep features are expected to embed high level discriminative interpretations associated with such local regions; and 3) the evolution of such abstract characteristics of the local regions over time, if extracted automatically, bears far reaching significance in designing a robust action recognition system.

Inspired from these considerations, the second part of the thesis proposes a novel mid-level feature representation for action videos. In this regard, we introduce the notion of *concept*, a mid level representation capable of capturing the evolution of motion salient local regions over consecutive video frames. The concepts are thus represented as chain graphs of such coherent regions at the temporal scale, and we propose an iterative maximum bipartite graph matching strategy for obtaining them automatically. The extracted concepts are less in number, rich in semantic information, can capture long-range video dynamics, and are able to highlight the subtle differences between action categories efficiently. In addition, the constraint regarding the explicit detection of objects being interacted can completely be bypassed given the automatic unsupervised discovery of concepts.

In order to prove the generality and robustness of this descriptior, it is necessary to employ them on supervised classification scenarios with labled data as well as situations with scarce data (zero shot learning, in the present case). While we rely on the metric learning based multi-class Support Vector Machines (SVM) for fully supervised recognition, we propose a novel semi-supervised zero-shot learning (ZSL) strategy which can benefit from the semantic descriptive power of our visual embedding effectively. In contrast to the standard ZSL approaches where an attribute [LNH14a] or word-embedded space [XHG15] is learnt from existing feature space for semantic interpretation, we completely rely on the visual embedding and directly learn correspondence between the label and the proposed concept spaces. Similarly, for zero-shot recognition, our descriptor exhibits enhanced performance but without the need of costly human annotations of attributes or semantic word embedding.

Exclusive scenarios where no data of test classes are present in training time requires transferring of a classification model trained on a set of *seen* classes and deploying it on a set of completely different classes - the *unseen* ones [LNH09, FEHF09, XSA17]. In order to achieve that, zero-shot learning (ZSL) approaches take advantage of *semantic embeddings* which act as a sort of “bridge” between seen and unseen classes. Depending on the nature of semantic embedding, ZSL approaches can be classified in two categories, one based on attributes and the other based on distributed word embeddings.

Attribute-based methods leverage human defined attributes to describe the classes to be discriminated.

Specifically, attributes are binary vectors in which each entry denotes the presence/absence of a particular feature characterizing the ”object” or class. For instance, in the case of animal classification [LNH09], a model trained on *zebras* is able to recognize *horses* since informed that both have four legs/hoofs, are mammals, have a mane and they both eat grass. Crucially, since attribute annotation is an expensive process, *distributed word embeddings* (DWEs) - such as word2vec [MSC$^+$13] or GloVE [PSM14] - are used as surrogates. They are learnt from a deep neural network that creates continuous embeddings from a text corpus by imposing that two words have nearby embeddings if they often occur close to each other in a sentence.

Each one of these two types of semantic embeddings has drawbacks. Finding an exhaustive list of attributes by manual annotations is usually expensive and difficult; on the other hand, DWE-based approaches are not easily interpretable. More importantly, the semantic information provided by attributes/word-embeddings is typically unable to encode semantic patterns in visual data. For instance, still in the example of the *zebra*-to-*horse* transfer above quoted, a great boost to ZSL would be provided by noting that, in addition to sharing attributes, zebras look extremely similar to horses - apart from the stripes.

Recently, a few works [MGS14, KXFG15a, JWS$^+$17a, QLSH17] have tried to incorporate visual information in the semantic embeddings by aligning the geometry of *semantic space* (made of either attributes or DWEs) onto the *visual space* produced by the visual feature representation, usually provided by fully connected layers of a convolutional neural network (CNN). These works leverage the implicit assumption that, among the visual and the semantic spaces, the former is preferable and the latter should be modified accordingly. Differently, we posit that semantic and visual spaces are equally important since providing complementary sources of information. Therefore, as opposed to modifying the semantic embedding on the basis of visual patterns, we propose to **augment** it by using visual semantic information extracted from the data itself. This *augmentation* is semantic in nature since it exploits the class similarity information obtained from a deep neural network in the form of *soft labels* [TSS07], which are finally jointly considered with the semantic attributes/DWEs. This is performed by devising an optimization process in which the latent attributes are inferred in the resulting *visually-driven* augmented space, which globally includes the semantic embedding, the visual features and the soft labels.

Differently from the hard labels (e.g., one-hot encoding), which only describe the correct class, soft labels [TSS07] estimate the likelihood probability distribution for an arbitrary instance to belong to every class. Therefore, if two classes are visually similar to each other, we expect this similarity to be captured by soft labels, and we claim that this fact can boost performance in ZSL methods.

In summary, this thesis discusses thoroughly the different data dependent scenarios and presents novel approaches applicable to different data driven scenarios. More specifically, the main contributions of this thesis are the summarized in the following.

1.2 Contributions

1. A novel supervised discriminative dictionary learning strategy is introduced for the purpose of action recognition from still images as well as videos. This takes advantage of available labeled training samples to adaptively rank local features which are both robust and discriminative. Further clustering of local features is done at the entity and category levels to eliminate the effects of features corresponding to non-recurrent or background locations. The adaptive ranking paradigm proposed holds wider applications in areas including feature selection, ranked set generation for retrieval etc. This serves as an ideal framework for fully supervised learning.

2. A framework for extracting discriminative mid-level representations of action videos is introduced. Novel algorithms for both the feature detection and description stages are also proposed. An extension of this framework can cope with the paradigm of zero-shot action recognition. In that regard, a semi-supervised clustering formulation is introduced which effectively embeds a semantic interpretation of the proposed visual embedding space. Essentially, this framework can handle traditional supervised learning as well as zero-shot learning scenarios.

3. A visually-driven semantic augmentation (VdSA) method for zero-shot learning (ZSL) approach is proposed that augments the semantic information coming from attributes/Distributed Word Embeddings (DWE) with that of the visual patterns embedded in a deep network's soft labels. This is well suited for training with scarse data.

4. To the best of my knowledge, this work is the first to highlight two starkly different scenarios An unified discussion of such opposite requisites show the need for data-driven approaches. At the same time, this work introduces novel approaches to improve in both traditional and zero-shot learning.

1.3 Publications

1.3.1 Journal

1. Abhinaba Roy, Biplab Banerjee, Vittorio Murino. Discriminative Body Part Interaction Mining for Mid-Level Action Representation and Classification. **Journal of Visual Communication and Image Representation, Elsevier**. [RBM18]

1.3.2 Conferences

1. Abhinaba Roy, Biplab Banerjee, Vittorio Murino. Discriminative Latent Visual Space For Zero-Shot Object Classification. 24^{th} **International Conference on Pattern Recognition (ICPR)**, 2018. [RBM]

2. Abhinaba Roy, Jacopo Cavazza, Vittorio Murino. Visually-driven Semantic Augmentation for Zero Shot Learning. **British Machine Vision Conference (BMVC)**, 2018. [RCM]

3. Abhinaba Roy, Biplab Banerjee, Vittoro Murino. A Novel Dictionary Learning based Multiple Instance Learning Approach to Action Recognition from Videos. **6th International Conference on Pattern Recognition Applications and Methods (ICPRAM)**, 2017. [RBM17b]

4. Abhinaba Roy, Biplab Banerjee, Vittorio Murino. Discriminative Dictionary Design for Action Classification in Still Images. 19^{th} **International Conference on Image Analysis and Processing (ICIAP)**, 2017. [RBM17a]

Chapter 2

Related Work

The first part of this thesis deals with codebook construction (dictionary learning). Subsequent chapter deals with mid-level feature representation for videos. Since these two chapters deal with action recognition in images and videos, we discuss the related works in action recognition in brief here, stressing upon works with mid-level features. Then we provide a comprehensive discussion of related dictionary learning techniques. Finally, we discuss about zero-shot learning.

2.1 Action recognition

2.1.1 Images

Recognition of human actions [CWS$^+$15] has been tried using images of an action [YWM10]. Following the standard dictionary learning based image classification scenario, a typical framework extracts dense SIFT [Low04] from the training images and codebook is constructed by clustering the SIFT descriptors by k-means clustering. Further, efficient encoding techniques including bag of words (BoW), LLC, Fisher vector are used to represent the images before the classification stage is carried out in such a feature space [CLVZ11].

Since the inherent idea of the BoW based frameworks is to learn recurring local patches, a different set of approaches directly models such object parts in images. Such techniques either initially define a template and try to fit it to object parts or iteratively learn distinctive parts for a given category.

Discriminative part based models (DPM) [FGMR10] are used extensively for this purpose and they served as the state of the art for a period. The hierarchical DPM model is used to parse human pose for action recognition in [WTLF12]. An efficient action and attributed representation based on sparse bases of local features is introduced in [YJK$^+$11]. An expanded part model for human attribute and action recognition is proposed in [SJS13]. The effects of empty cavity, ambiguity and pooling strategies are explored in order to design the optimal feature encoding for the purpose of human action recognition in still images in [ZLP$^+$16].

Very recently, the part learning paradigm has gained much attention because of its ability to represent mid-level visual features. Given a large pool of region proposals extracted from the images, such techniques iteratively learn part classifiers with high discriminative capabilities. Methods based on partness analysis [JVJZ13], deterministic annealing for part learning [SJ15] etc. are some of the representative in this respect.

2.1.2 Videos

Video level features for the sake of human action recognition are broadly classified into: low, mid and high level deep features. Amongst low level features, a large number of endeavours are based on encoding of local spatio-temporal variations in the videos in terms of the STIP features [LMSR08]. Apart from STIP; tracklets [RS10], cuboid, dense trajectories [WKSL11] are used successfully for human action recognition exhibiting improved recognition performance measures. Arguably, dense interest points based descriptors outperform their sparse counterparts for most of the datasets [WUK$^+$09]. Ideally, the descriptors corresponding to the local interest points need to be encoded into fixed length vectors for the sake of classification and several encoding strategies are used for this purpose [DTS06] [PSM10]. Following the overwhelming success of deep CNN based models in image classification, a number of deep architectures have been proposed recently in relation to video based action recognition [JXYY13]. In this connection, [KTS$^+$14] proposes to train a 3D CNN model which considers each video as a stack of consecutive frames. On the other hand, the popular two stream networks proposed in [SZ14a] consider two separate CNN models for encoding the appearance and motion components of a video and further fuse the response of both the networks at a later stage. In addition, some of the recent frameworks consider the videos as time-series data [FGO+17] and introduce recurrent networks based models for action recognition in videos [TBF+15] [SMS15]. Low level features are pixel based, spurious and mostly susceptible to physical changes (illumination, shadows, clutter etc.) in videos, whereas deep features lack transferability of abstract action concepts.Alternatively, Mid level features, which can take care of most of these shortcomings, have gained much attention in vision community in recent times. The majority of existing mid-level features for videos are based on grouping spatially consistent low-level features into clusters. Such clusters are the prototypes of the characteristics observed in video sub-volumes. [RKS12] proposes to group locally dense trajectories to form motion consistent action parts. Similarly, motion atoms and phrases are used to temporally segment the videos in [WQT13]. A binary partition tree based hierarchical video representation is introduced in [PS13] which is based on grouping the dense trajectories at different levels. A set of classifiers over the spatio-temporal volume is considered to learn exemplar-based video features in [TT16]. Each classifier is an exemplar SVM defined discriminatively on the low-level features of each video volume. Action proposals are used in [YY15]. Very recently, two-stream CNN network is coupled with the improved dense trajectories to define a context-sensitive mid-level feature [WQT15] for videos. In addition, unsupervised discovery of video tubes (tubelets [JVGJ+14], APT [vGJGS15]) are closely related to mid-level feature disclosure.

2.2 Dictionary Learning

Dictionary learning strategies can be supervised or unsupervised in nature. A class of unsupervised dictionary learning strategies compute over-complete sparse bases considering the idea of alternate optimization [OF97]. Such techniques iteratively update the dictionary components and sparse coefficients for the input samples using k-SVD and matching pursuit based methods. [WYY+10] proposes the LLC technique where a locality constraint is added to the loss function of sparse coding. [LBRN06] introduces an l_1-norm based sparse coding algorithm where feature-sign search is applied for encoding and Lagrange dual method for dictionary learning. Effective sampling strategies for the BoW model is the focus of discussion in [NJT06] where several aspects including the codebook size, clustering techniques adopted etc. are exhaustively studied.

On the contrary, the supervised approaches include the class support in building the dictionary. Label consistent SVD [JLD11], logistic regression based sparse coding [MPS$^+$09] explicitly consider the class discrimination in designing the sparse bases for dictionary learning. Two different clustering based approaches for keypoints selection are introduced in [LTCK16] for the purpose of dictionary learning based generic scene recognition. Distance measures among the keypoints are modeled in an online fashion to filter out keypoints with low generalization capability.

2.3 Zero-shot Learning

The majority of the initial endeavors on zero-shot visual classification are based notion of human annotated semantic attributes which are used to model the category prototypes. [LNH14a, LNH09] are some of the foremost works in this line where the authors propose direct (DAP) and indirect attribute prediction (IAP) paradigms requiring explicit mappings between the attributes and the visual categories. On the other hand, [AHS15] proposes a hierarchical approach for attribute label propagation for visual classes. From a different point of view, [RSS11] introduces a hierarchical ZSL framework using the notion of class taxonomy. [RPT15b] develops a linear network to handle relations between classes, attributes and features. Nonetheless, the major shortcoming of such approaches is primarily due to the fact that explicit human annotations are costly, more often domain specific, and subjective.

Alternatively, semantic representations barring the need for defining explicit problem specific attributes have recently been brought into attention to the ZSL community [LBSF$^+$15]. Auxiliary information obtained from linguistic sources are extensively studied for this purpose [APHS13, FCS$^+$13, CCGS16]. Especially, vector embeddings based on the idea of distributed word representations have gained popularity. In particular, a vector-space is extracted from linguistic knowledge bases, e.g., Wikipedia, news database etc. in terms of sophisticated mapping modules like Word2Vec[GL14] and the class names are projected onto such a learned space preserving the semantic properties of the classes at large. An embedding between visual features and these signatures are learned and the notion of similarity between signatures in the vector-space is further used to carry out the inference stage for ZSL. Inspired by this analogy, [APHS13] uses auxiliary information to learn an additional embedding between attributes and class labels.

Besides the extensive research on defining a robust semantic embedding space, many of the recent ZSL techniques aim at learning visually meaningful feature descriptors for ZSL. [FCS$^+$13] proposes a deep learning model which aims at learning the relation between images and the corresponding class name embeddings. [NMB$^+$13] conveys that a convex combination of the class-name embeddings, weighted by class posterior probabilities given by a pre-trained CNN model, can be used to map images to the respective class-name embedding space. The zero-shot paradigm considers non-overlapping training and test classes and aims at classifying previously unseen test instances. That is realized using attributes [LNH14a] which reduce the semantic gap between low-level visual features and class descriptors. Visual to attribute classifiers are typically modelled on an auxiliary dataset, and novel categories are subsequently specified by a human agent in terms of the attributes which accredit recognition in absence of training data. There are only a few methods in the literature which adopt this paradigm for action recognition [LKS11] since it is costly and ambiguous to define attributes for actions. On the other hand, the semantic embedding space based ZSL techniques are based on a distributed representation of the class names in terms of text words [XHG15]. Regressors are then used to map low-level visual features into this word-vector space. Zero-shot recognition is subsequently performed by mapping novel category instances and class names to this common space and performing nearest neighbour matching [XHG15]. However, the mapping from the visual features to semantic space is a complex process and such methods are not expected to generalize well if the unseen action categories (eg. running vs walking) are largely different from the seen ones (hand-waiving, hand-clapping). In addition, sparse coding based methods are also used for ZSL based action recognition [KXFG15b], where the feature spaces of both the seen and unseen classes are assumed to be related by substantial domain-shift.

Chapter 3

Dictionary Learning

In a supervised learning setup, the best possible scenario is when data during training phase spans equally over all the possible classes and all data labels are human verified. In this case, one does not need to worry about imbalance in data creating unnatural bias in the learned model. The problem of training a system focuses on better representation of data and/or features for the relevant task. In this section, we concentrate on the problem of classification, especially action classification in images and videos. Since ample data is available, the system concentrates on better (and discriminative) representation of each action categories for better classification performance. In that regard, dictionary learning becomes important. In this chapter we come up with a novel dictionary learning based approach that is well suited for situations with ample labeled data, in this particular case, action recognition in images and videos. We follow a standard action classification framework consists of four major stages: 1) Extraction of local features, 2) Discriminative dictionary construction, 3) Feature encoding, 4) Action classification.

For notational convenience, let us consider that $TR = \{X_i, Y_i\}_{i=1}^{N}$ constitutes N training examples belonging to L action categories where each X_i represents an image or a video and Y_i is the corresponding class label. Entities in TR are represented by a set of local descriptors $F_i = \{F_i^1, F_i^2, \ldots, F_i^{\alpha_i}\}$ where $F_i^k \in \mathbb{R}^d$ and $d = 4096$ or $d = 162$, respectively, depending on whether the underlying X_i is an image or a video. In addition, α_i represents the number of local descriptors extracted from X_i. Further, $\{C_1, C_2, \ldots, C_L\}$ represents the set of category specific codebooks learned by the proposed algorithm by exploiting the local features extracted from TR, whereas $C = [C_1C_2 \dots C_L]$ is the global codebook obtained by the concatenation of the local ones.

3.1 Extraction of Local Features

We consider category independent region proposals to highlight local regions in still images whereas the popular STIP features are used for video streams.

Region proposal generation techniques highlight region segments in the image where the likelihood of the presence of an object part is high. This provides a structured way to identify interesting locations in the image and thus reduces the search space for efficient codeword generation. We specifically work with the objectness paradigm for region proposals generation from still images which is based on modeling several aspects regarding the characteristics of the objects in a Bayesian framework. Each region proposal is further represented by the CNN features. We prefer the ImageNet pre-trained VGG-F [CSVZ14] model which has an architecture similar to AlexNet [KSH12a], and comprises of 5 convolutional layers and 3 fully-connected layers. The main difference of VGG-F and AlexNet is that VGG-F contains less convolutional layers and uses a stride of 4 pixels leading to better evaluation speed than the AlexNet architecture. In case of videos, the representation of local variations depends on local STIP keypoints. STIP features are the extension of the Harris corner detectors for images to the spatio-temporal domain. They are detected at locations where the video frame level intensities have significant local variations in both space and time. Histogram of oriented gradients (HOG) and histogram of optical flow (HOF) features are extracted around each STIP point.

3.2 Discriminative Dictionary Learning

We first build category specific codebooks and then concatenate all the local codebooks to generate a global codebook.

3.2.1 Separate dictionary learning for each category

For a given $l \in \{1, 2, \ldots, L\}$, the dictionary learning process is summarized as follows:

1. For each training instance with the category label l, we first group the local descriptors using MS clustering and consider the cluster centroids as constituting the reduced set of local descriptors. MS is an iterative, non-parametric clustering method which does not require an estimation of the number of clusters as input. Instead, it relies on the kernel density estimate in the feature space to group samples which form dense clusters. Given $F_i = \{F_i^1, F_i^2, \ldots, F_i^{\alpha_i}\}$, the kernel density estimate at a point F_i^k is expressed as

   $$f(F_i^k) = \frac{1}{\alpha_i h^d} \sum_{m=1}^{\alpha_i} K(\frac{F_i^k - F_i^m}{h}) \tag{3.1}$$

   where K is a radially symmetric kernel function and h defines the width of the Parzen window to highlight the neighbourhood around F_i^k. A cluster is identified as the region where the data density is locally maximum. This can alternatively be interpreted as the local regions where $\nabla f \approx 0$. ∇f can efficiently be calculated by iteratively shifting the centroids of the Parzen windows until the locally dense regions are reached [CM02].

   Since all the descriptors in a dense region in the feature space highlight near similar local features, the mean-shift clustering is able to select one unique representative for all of them. Further, since mean-shift implicitly estimates the number of clusters present in the dataset, hence, the problem of over-merging is greatly reduced. On the other hand, spherical clustering techniques like k-means and fuzzy c-means create suboptimal codebooks as most of the cluster centroids fall near high density regions, thus under-representing equally discriminant low-to-medium density regions. MS resolves such problem by focusing on locally dense regions in the feature space. Let $\widehat{F_i} = \{\widehat{F_i^1}, \widehat{F_i^2}, \ldots, \widehat{F_i^{\alpha_i}}\}$ represents the new set of local descriptors for the i^{th} training instance where each $\widehat{F_i^k}$ represents a cluster centroid.

2. Once $\widehat{F_i}$s are constructed for all the training instances with category label l, we vector quantize all such $\widehat{F_i}$s using MS clustering to build a temporary codebook $C_l = \{C_l^1, C_l^2, \ldots, C_l^{\beta_l}\}$ for

the category with each C_l^k representing a codeword (cluster centroid). Similar to the previous stage, it is guaranteed that C_l is ensured to capture all the potential local features for the l^{th} category.

$\{C_1, C_2, \ldots, C_L\}$ are constructed in the similar fashion for $l \in \{1, 2, \ldots, L\}$. It is to be noted that the labels of the codewords depend upon the action categories they refer to. Further, the sizes of the C_ls may differ from each other. The C_ls thus obtained are not optimal in the sense that they contain many codewords with low discriminative property. Such codewords need to be eliminated in order to build robust category specific codebooks. However, we need a measure to rank the descriptors based on their discriminative ability. In this respect, the following observations can be made:

- A potentially discriminative codeword is not frequent over many of the categories constituting the dataset.
- Most of its nearest neighbors in $\{C_1, C_2, \ldots, C_L\}$ share the same class label with the codeword under consideration.

We model the first observation in terms of the idea of conditional entropy whereas the second observation is replicated by the tf-idf score.

For a given codeword C_l^k, we find out the labels of its T nearest neighbours over the entire set of codewords in $\{C_1, C_2, \ldots, C_L\}$ and subsequently define the conditional entropy measure as:

$$H(Y|C_l^k) = -\sum_{l'=1}^{L} p(l'|C_l^k) \log_2 p(l'|C_l^k) \tag{3.2}$$

where Y is the class lebel and $p(l'|C_l^k)$ represents the fraction of the retrieved codewords with label l'. For discriminative codewords, i.e. the ones which do not span many categories, H is small whereas the value of H grows with the selection of codewords shared by many categories.

In addition to the H score, we also expect the nearest neighbours to be populated from the same category as of C_l^k. In order to impose this constraint, we define the tf-idf score for C_l^k as follows:

$$TI(C_l^k) = \frac{|C_{l'}^{k'}|C_{l'}^{k'} \in knn(C_l^k)\ AND\ l' = l|}{|C_{l'}^{k'}|C_{l'}^{k'} \in knn(C_l^k)|} \tag{3.3}$$

Both the measures are further combined in a convex fashion to define the ranking measure as follows:

$$Rank(C_l^k) = w_1 \, \frac{1}{H(Y|C_l^k)} + (1 - w_1)\, TI(C_l^k) \tag{3.4}$$

We repeat this stage for all the codewords in $\{C_1, C_2, \ldots, C_L\}$. As already mentioned, the $Rank(C_l^k)$ has high values for potentially discriminative and category specific codewords.

3.2.2 Number of codeword selection

We rank the codewords on the basis of the $Rank$ scores. In order to select the number of optimal codeworks to select, we use an adaptive algorithm. This is in stark contrast to the related work done in [RBM17a], where top B codewords were chosen in a greedy fashion in order to define the final codebook $\widehat{C_l}$ for category l. For this adaptive algorithm, we make a spanning tree of the code words C_l. We first create a spanning tree G with C_l as nodes and the edge weights as the difference between the features. Note that the nodes are connected sequencially based on the ranked list. On this spanning tree, we carry out a dominant set clustering [PP07]. More specifically, we carry out a binary clustering on the graph to get two subgraphs. We select the subgraph (set of codewords) with higher rank. This results in variable numbers of codewords in each of the classes. Algorithm 1 describes the adaptive number of codeword selection process.

3.2.3 Global dictionary construction

The local codebooks obtained in the previous stage are concatenated in order to obtain a global codebook $\widehat{C} = [\widehat{C_1}\widehat{C_2} \ldots \widehat{C_L}]$.

Algorithm 1: Adaptive number of codewords selection

```
Input: $C_l$
Output: $\widehat{C_l}$
1. Construct a linear chain graph $G = \{C_l, E\}$ with the extracted parts from a given class as
   the nodes and the edge weights are defined as Euclidean distance between the corresponding
   features ($E$).
3. Perform dominant set clustering on $G$ to to obtain two sub-graphs;
4. Choose the subset ($\widehat{C_l}$) with the parts having higher ranks according to the proposed cost
   function;
```

3.3 Feature encoding using $\widehat{C}$

We represent each visual entity with respect to $\widehat{C}$ for still images and videos separately. We find that LLC based encoding works best while dealing with the CNN features in case of action recognition in still images, whereas Fisher vector outperforms other BoW based encoding methods for video based features. For each entity, we consider all the initially extracted local features for encoding.

3.4 Action classification

The final classification is performed using random forest ensemble classifier [Bis06a]. The decision tree learning algorithm used is information gain and bootstrap aggregation is employed to learn the ensemble model. Thus the forest reduces classifier variance without increasing bias. Random subspace splitting is used for each tree split and we consider $\sqrt{d}$ features for each split given d original feature dimensions. The generalization is performed by applying majority voting on the outcomes of the learned trees.

Label propagation

In order to refine and further strengthen the classification results from random forests, we apply an additional round of label propagation. In it's original form, label propagation is a semi-supervised classification way to propagate labels from labeled sample to unlabeled samples [ZG02]. Based on the idea that samples should have same labels if they are neighbors to each other, label propagation

”propagates” labels of labeled samples to unlabeled samples according to the proximity. The more similarly samples are, the more easily propagate labels between them. The similarity of samples is calculated as and exponential of negative distance between them (Equation 3.5).

$$w_{ij} = exp(-\frac{d_{ij}^2}{\sigma}) \tag{3.5}$$

where d_{ij} is the distance between i^{th} and j^{th} sample. The degree of difficulty for label propagation is described by probabilistic transition matrix T_{ij}, which is defined as

$$T_{ij} = P(x_i \rightarrow x_j) = \frac{w_{ij}}{\sum_{k=1}^{l+u} w_{kj}} \tag{3.6}$$

where T_{ij} is the probability of switching label (inference) from instance x_i to x_j. Larger w_{ij} leads to larger T_{ij} that allows labels to propagate more easily. The process of label propagation continues until labels of all samples tend to be stable. Label propagation defines a label matrix Y with all the label probabilities of data points x_is. In this work, we attempt to propagate the labels of most confident labels onto the other inferences with less confidences. First we detect all the output labels from the random forest classifier which most likely provide true inference for a certain test instance. In the cases where there is ambiguity amongst the tree classifier outcomes, we expect propagation of confident labels will solve this issue and provide better classification performance. In order to measure the ambiguity between individual tree classifiers in the random forest, we device a measurement to determine the *confidence* of inference for each test instances. We define the confidence of the inferred labels in terms of agreement of trees in majority voting of random forest. Hence confidence of inferred y_i from a test instance x_i is denoted by

$$conf_i = \frac{N_{y_i}}{N} \tag{3.7}$$

where N_{y_i} denotes the number of trees which inferred y_i as the label of x_i and N is the total number of trees. We rank the test samples in order of confidence of inferences from them. Consecutively, we use top 20% confident inferences and apply label propagation to improve the outcome of the rest of the inferences. The algorithm related to label propagation is shown below.

```
Algorithm 2: Label propagation for improved classification
  Input: Test instances with confident inferences $(x_1, y_1), ..., (x_l, y_l)$. Other test instances with
         not confident inferences $(x_{l+1}, y_{l+1}), ..., (x_{l+u}, y_{l+u})$. $y_i$ is the inference from random
         forest for instance $x_i$. $Y_L = y_1, ..., y_l$; $Y_U = y_{l+1}, ..., y_{l+u}$; $Y = [Y_U; Y_L]$
  Output: Labels of all instances
  1. Calculate $w_{ij}$ between samples;
  2. Calculate probabilistic transition matrix $T$;
  3. Structure matrix Y;
  4. $TY \to Y$;
  5. Normalize $Y$;
  6. Reset $Y_L$;
  7. Repeat step 4 until $Y$ does not change/maximum number of iteration is reached;
```

3.5 Experimental details

3.5.1 Dataset

We consider the Stanford-40 [YJK$^+$11] still image action recognition database and UCF-50 [RS13] video based action recognition dataset to evaluate the effectiveness of the proposed framework. Stanford-40 actions is a database of human actions with 40 diverse action types, e.g. brushing teeth, reading books, blowing bubbles, etc. The number of images per category ranges between 180 to 300 with a total of 9352 images. We use the suggested [YJK$^+$11] train-test split with 100 images per category as training and remaining for testing. On the other hand, the UCF-50 dataset contains videos representing 50 actions in a unconstrained environments. The dataset contains a total of 6700 videos with about $100 - 150$ videos per category. This dataset is a superset of the popular UCF-11 dataset. We randomly select 60% of the videos per category to represent the training set and the remaining 40% is used to evaluate the classification performance of the proposed framework.

3.5.2 Experimental setup

The following experimental setup is considered in order to evaluate the performance of the proposed framework for both the datasets.

- MS clustering is used in conjunction with the Gaussian kernel. The adaptive bandwidth parameter (h) is fixed empirically as $\frac{D}{m}$ $(1 \leq m \leq 10)$, where D is the average pairwise distance of all the local descriptors extracted from all the visual entities of each category. The same setup is repeated for MS clustering in the entity and the category levels(section §3.2.1).

- We extract 500 region proposals per image for the Stanford-40 dataset. Figure 1 depicts the extracted region proposals for a pair of images from the dataset. We further discard proposals which are largely overlapping to each other (overlap of $\geq 50\%$) in order to highlight potentially discriminative local patches in the images. The STIP keypoints are extracted from the videos using the publicly available implementation of [Lap05].

- The number of final distinctive codewords selected for each class are set adaptively as discussed before. As for feature encoding, for LLC, $100 - 200$ nearest neighbors per local descriptor are considered to encode the images. We select the optimal hyper-parameters by cross-validation. Each image in the Stanford-40 dataset is optimally represented by a sparse vector of length 8000×1 (100 neighbors in LLC).Whereas each video has a feature length of 32400×1 $(2 \times \mathrm{feature\ dimension}(162) \times \mathrm{No.\ of\ Gaussian\ components}(100))$.

- Each component tree in the random forest model is essentially a classification and regression tree (CART) [Bis06a]. We conduct experiments with random forest of different sizes ($500 - 2000$) and find that a random forest with 1000 CART trees exhibits superior performance.

- For post processing based on label propagation, we fix the threshold for ”confident” inferences to 20%. This value is determined by 5 fold cross validation for both image and video data.

- We compare the overall classification performance of the proposed technique with the representative techniques from the literature. All the experiments are repeated multiple times and the average performance measures are reported.

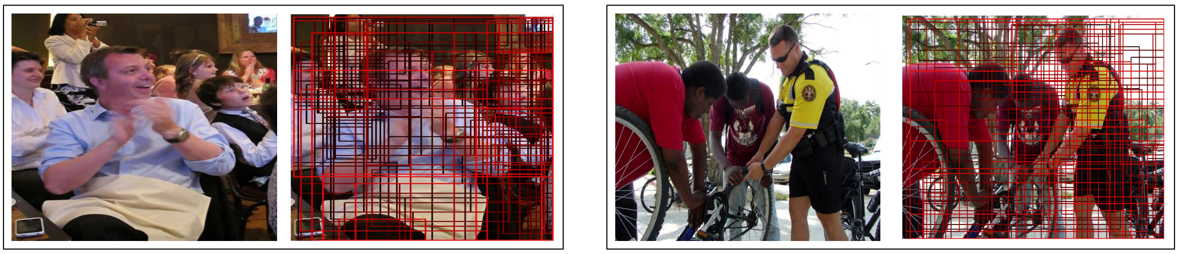


Figure 3.1: Extraction region proposals from images of Stanford-40 using objectness

3.5.3 Performance evaluation

In this section we evaluate the performance of our framework on action recognition over benchmark datasets with still images (Stanford-40) and videos (UCF-50).

Evaluation on Stanford-40

We evaluate the performance of our approach in two folds. First we provide an ablation study to evaluate performance of our adaptive number of codeword selection.The results are shown in Table 3.1. In the previous work of [RBM17a], number of codewords are chosen empirically after cross validation, still adaptive selection improves that performance by 0.4%. This goes on to show that adaptive selection not only eliminate the need to execute cross-validation to select the number of code words, but also improves previous performance by a margin.

Next, we demonstrate the effect of label propagation (§3.4) in our pipeline. We compare the effect of addition of label propagation in Table 3.2. We achieve a performance improvement of 1.8%. A simple assumption that neighbouring data points tend to have similar labels clearly improves the performance of random forest. This also shows that label propagation is an effective tool to reduce the tree classifier confusions that arise in a random forest. It also adds helps majority voting, that is applied in random forest alleviate the problem of tree classifier confusions and avoid misclassification.

Finally, we evaluate the performance of our approach against other related action recognition pipelines in literature. Table 3.3 mentions the accuracy assessment of different techniques for the Stanford-40 dataset. The performances of the methods based on hand-crafted SIFT-like features are comparatively less ($\approx 35.2\%$) [WYY$^+$10]. This can be attributed to the fact that differences in human attributes for many of the action classes are subtle. Label consistent K-SVD provides 32.7% accuracy. Part

Table 3.1: Effect of adaptive number of code word selection for Stanford-40

Method	**Classification accuracy**
Top B (B=200 chosen empirically) codewords and LLC encoding [RBM17a]	49%
Proposed framework with adaptive number of codewords and LLC encoding	49.4%

Table 3.2: Effect of label propagation for Stanford-40

Method	**Classification accuracy**
Proposed framework with adaptive number of codewords and LLC encoding	49.4%
Proposed framework with adaptive number of codewords and LLC encoding +label propagation	**51.2%**

learning based strategies obtain better recognition performance in this respect by explicitly modelling category specific parts. Classification accuracy of 40.7% is obtained with the generic expanded part models (EPM) of [SJS13] which is further enhanced to 42.2% while the contextual information is incorporated in EPM. The best performance with shallow features obtained for this dataset is 45.7% by [YJK$^+$11] which performs action recognition by combining bases of attributes, objects and poses. Further they derive their bases by using large amount of external information. It is worth noting that, the ImageNet pre-trained AlexNet reports a classification accuracy of 46% [KSH12a]. With our framework, we observe an improvement of 5.2% over them. It can be argued that our method encapsulates the advantages of deep and shallow models effectively in a single framework. The CNN based region proposals are capable of encoding high level abstractions from the local regions. Since the images are captured in unconstrained environments, the backgrounds are uncorrelated in different images of a given category. The per category dictionary learning strategy reduces the effects of such background patches and the proposed ranking measure further boosts the proposals corresponding to the shared human attributes, human-objects interaction etc. for a given action category. In contrast to other techniques which are based on SVM classifier, our framework relies on the random forest model which does not explicitly require any cross-validation. We observe that performance of the random forest model gradually improves with growing number of CART trees within the range $500 - 1000$ and a random forest model with 1000 trees outputs the best performance. Further, the addition of label propagation overcomes the problem of misclassification due to confusion created because of ensemble nature of random forest.

Table 3.3: A summary of the performance of our classification framework for the Stanford-40 data in comparison to the literature

Method	**Classification accuracy**
ObjectBank [LSFFX10]	32.5 %
label consistent K-SVD [JLD11]	32.7 %
LLC with SIFT features [WYY$^+$10]	35.2 %
Spatial pyramid matching kernel [LSP06]	34.9 %
Expanded parts model [SJS13]	40.7 %
CNN AlexNet [KSH12a]	46 %
Proposed framework (with adaptive number of codeword selection and LLC encoding & label propagation)	**51.2 %**

Table 3.4: Effect of adaptive number of code word selection for UCF-50

Method	**Classification accuracy**
Top B (B=200 chosen empirically) codewords [RBM17a] and fisher vector encoding	64%
Proposed framework with adaptive number of codewords and fisher vector encoding	**64.5%**

UCF-50 dataset

For UCF-50, we divide our experiments in similar way we do for Stanford-40. Table 3.4 shows the effect of adaptive selection of number of codewords. Similar to that of Standford-40, we observe an increment of 0.5% in result. Table 3.5 shows the effect of label propagation on random forest. Label propagation improves the performance of random forest by 2.2%. Encoding videos properly is inherently more complex than images due to added difficulty of encapsulating changes along progression of time. This in turn creates confusion in an ensemble setting such as random forest. Label propagation works well in this scenarios which is evident from such a high increment of result.

We compare the performance of our framework with that of three different shallow representations from the literatures with similar train-test split (Table 3.6). The standard STIP (HOG + HOF) with the BoW encoding and the frame based GIST descriptors [OT06] exhibit classification performances of 47.9% and 38.8% respectively. Since the differences between many of the action classes in UCF-50 are fine-grained and the videos contain substantial camera motion and cluttered backgrounds, models based on global descriptors fails drastically in discriminating the action classes. The Action-Bank [SC12] model based on learned action templates provides improved recognition performance (57.9%), although it requires numerous supervised information to learn the templates. Improved dense trajectory [WS13]; dense trajectory with additional RootSIFT normalization provided 65.2%.

Table 3.5: Effect of label propagation for UCF-50

Method	**Classification accuracy**
Proposed framework with adaptive number of codewords and fisher vector encoding	64.5%
Proposed framework with adaptive number of codewords and fisher vector encoding + label propagation	**66.7%**

Table 3.6: A summary of the performance of our classification framework for the UCF-50 data in comparison to the literature

Method	**Classification accuracy**
GIST [OT06]	38.8 %
STIP (HOG+HOF) + bag of words	47.9 %
ActionBank [SC12]	57.9 %
Improved dense trajectory [WS13]	65.3%
Proposed framework (with adaptive number of codewords and fisher vector encoding + label propagation)	**66.7%**

All the aforementioned setups are based on the SVM classifiers.

In contrast, the current framework exhibits the best average recognition accuracy of 66.7% (we use GMM with 100 components). The enhancement of the performance of the proposed framework is attributed to the robust ranking measure which selects recurrent and discriminative local features and reduces the effects of background patches by assigning low distinctiveness scores. This is established since the recognition performance of the system sharply decreases (recognition accuracy of $\approx 58\%$ when all the codewords are considered) as more codewords per category are considered to build the dictionary.

Essentially, We introduce a novel supervised discriminative dictionary learning strategy for the purpose of action recognition from still images as well as videos. We take advantage of the available training samples to adaptively rank local features which are both robust and discriminative. Further, we cluster the local features at the entity and category levels to eliminate the effects of features corresponding to non-recurrent or background locations. The adaptive ranking paradigm proposed in this work holds wider applications other areas including feature selection, ranked set generation for retrieval etc.

Chapter 4

Mid-Level Feature Representations

In a standard supervised classification setup, it is sufficient if representations of each category are discriminative enough. With enough labeled data, that can be achieved through proper dictionary learning techniques as discussed in the previous chapter. But for situations where labeled data is not available, there is a need to learn generalized representation from a small number of data. In this chapter, we look into feature representations that are equally effective in dealing with both data dependent scenarios. We take into account the extreme situation of scarcity of data where no example of tst categories are available during training. This is known as the problem of zero shot learning (ZSL). In this chapter, we investigate into a mid-level feature extraction paradigm that equally efficient in standard classification as well as ZSL classification (Fig. 4.1). As a case study, we focus on the problem of action classification from videos.

Firstly, to establish notations used in the first section of this chapter, let there be N labelled training videos $\{(X_i, Y_i)\}_{i=1}^{N}$ where $X_i = \{x_i^1, x_i^2, \ldots, x_i^{n_i}\}$ represents the frames of videos X_i and $Y_i \in \{1, 2, \ldots, C\}$ is the corresponding class label. For a given frame x_i^j, $\{r_{ij}^k\}_{k=1}^{m_i^j}$ represents the m_i^j motion salient region proposals (or the corresponding CNN features, $r_{ij}^k \in \mathbb{R}^{4096}$) extracted from x_i^j. Further, the chain graphs extracted from the videos (or the corresponding pooled features) are denotes as $\{\widehat{c_i^l}\}_{l=1}^{\nu_i}$ for a given X_i. In addition, each X_i is represented by a Q dimensional vector $(\phi_i^1, \phi_i^2, \ldots, \phi_i^Q)$ after the proposed dictionary learning process. All the notations and corresponding methods are described as follows. The proposed mid-level feature extraction framework initially

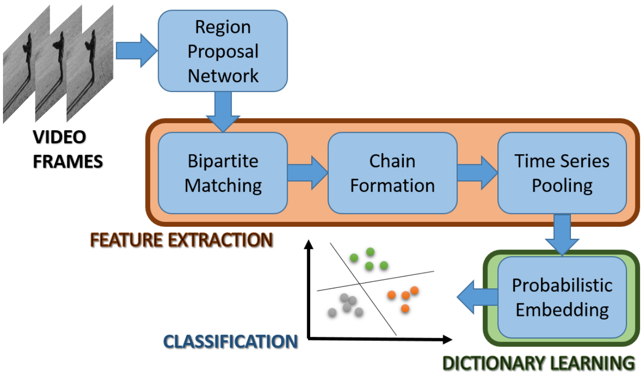


Figure 4.1: The proposed action recognition pipeline: Mid-level Descriptor Extraction, Dictionary Learning, Classification

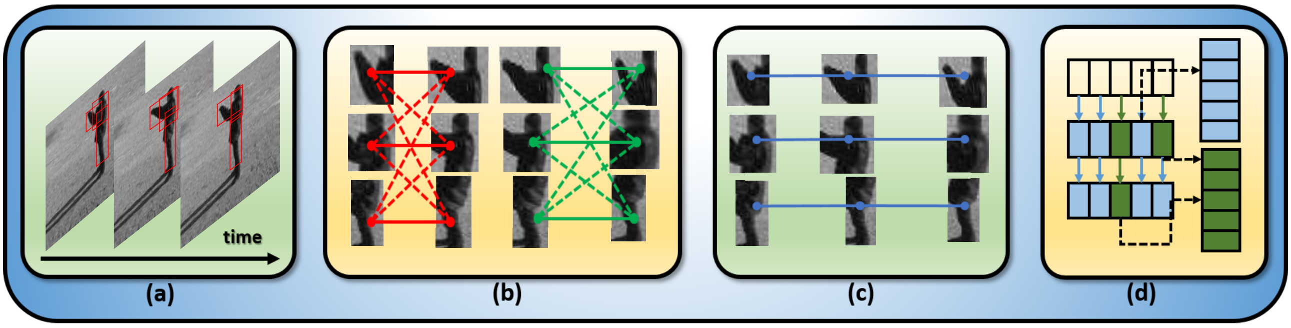


Figure 4.2: Mid level feature extraction: (a) Motion salient region proposals; (b) Pairwise bipartite graph matching (thicken lines denote outcome of the matching algorithm); (c) Chain formation (each chain is representative of a given *concept*); (d) time series pooling (blue denoted positive change and green denotes negative change)

constructs distinct temporal chain graphs (concept detection) of the category independent and motion salient region proposals extracted from the video frames. Further, a time-series pooling based feature extraction from the chains is carried out to encode the evolution of the regions inherently tracked by them (concept description). The stages (see Fig. 4.2) are described in details in the following.

4.1 Motion salient region proposal extraction

Given a video, we are interested in further analysing spatial regions which substantially capture the scene dynamics since they are expected to contribute to the recognition of the underlying activity. For the extraction of such motion salient regions, we rely on the region proposal network (RPN) of [RHGS15] which can generate accurate proposals in a near free-cost manner. In order to further account for the motion content of such region proposals, the graph based motion saliency detector of [HKP06] is used to assign saliency scores ($[0-1]$) to the pixels of the video frames and proposals with high average motion saliency (≥ 0.6) are retained. In addition, we ensure to discard sufficiently large ($\text{area} \geq 2000$ pixels), skewed (aspect ratio more than $6:1$ or $1:6$), and highly overlapping (*intersection over union* ($\text{IoU} > 0.5$)) proposals per frame. Further, high level CNN features (4096×1) are extracted from the selected proposals by using the final fully-connected layer of the ImageNet pre-trained VGG-16 model [SZ14c]. As an outcome, we obtain $\{r_{ij}^k\}_{k=1}^{m_i^j}$ for all the frames ($1 \leq j \leq n_i$) of X_i.

Remark 1. Motion salient region proposals ideally indicate different body-parts of the person pertaining to an action in the corresponding video. Subsequently, such proposals are tracked over consecutive frames in terms of a weighted bipartite graph matching strategy.

4.2 Concept detection

Region proposal selection. In order to track the movement of the selected proposals over the video frames, we propose an efficient iterative bipartite maximum weighted graph matching strategy for each pair of consecutive frames. The region correspondence obtained in this way are further linked in order to form the chain graphs.

Given $\{r_{ij}^k\}_{k=1}^{m_i^j}$ and $\{r_{ij+1}^k\}_{k=1}^{m_i^{j+1}}$ of X_i, an undirected bipartite graph $G(V, E)$ with $V = \{r_{ij}^k\}_{k=1}^{m_i^j} \cup \{r_{ij+1}^k\}_{k=1}^{m_i^{j+1}}$ and the bi-adjacency edge matrix $E_{m_i^j \times m_i^{j+1}}$ is constructed. The edge weights ($w(r_{ij}^p, r_{ij+1}^q)$) in E represent similarity measures between all pairs of proposals from both the frames as the convex combination of their spatial overlapping along with the appearance resemblance as follows:

$$\begin{aligned} w(r_{ij}^p, r_{ij+1}^q) = \delta \, \frac{R(r_{ij}^p) \cap R(r_{ij+1}^q)}{R(r_{ij}^p) \cup R(r_{ij+1}^q)} + \\ (1-\delta) \, \exp(-\gamma ||r(r_{ij}^p) - r(r_{ij+1}^q)||_2^2) \end{aligned} \tag{4.1}$$

Where first part of RHS denotes the spatial overlap between r_{ij}^p and r_{ij+1}^q, defined in terms of their respective spatial regions $R(r_{ij}^p)$ and $R(r_{ij+1}^q)$.

The second part of RHS in Eq. 4.1 represents the appearance similarity between r_{ij}^p and r_{ij+1}^q in terms of the Gaussian Radial Basis Function (RBF) kernel of their CNN features. We explicitly consider $w = 0$ for non-overlapping proposals (IoU $= 0$) in order to omit the construction of irrelevant chains and remove the respective edges in E. Since the movement of a given body part is consistent both in spatial arrangement and appearance, $w(p, q)$ is high if both p and q refer to unswerving regions in both the frames. Given the bipartite graph G, a matching is defined as a set of the edges chosen in such a way that no two edges share a common endpoint.

Remark 2. Since we aim at finding a bijective mapping between the proposals of both frames based on high similarity and consistency, we can alternatively find the maximum weighted matching from G which refers to our desired solution. We employ an equivalent approximation based on integer linear programming (ILP) for the maximum weighted graph matching problem for the bipartite graph G.

Maximum weighted bipartite matching. Given boolean variables Δ for the edges in E such that $\Delta_e = 1, (e \in E)$ for a matched edge and 0 otherwise, the maximum weighted matching problem can be formulated as:

$$\begin{aligned} \text{maximize} \quad & \sum_e w_e \, \Delta_e \\ \text{subject to} \quad & \forall v \in V, \sum_{e \sim v} \Delta_e \leq 1 \\ & \forall e \in E, \Delta \in \{0, 1\}. \end{aligned} \tag{4.2}$$

$e \sim v$ indicates that $e \in E$ is adjacent to the vertex $v \in V$. Optimal solution to this problem is the assignment vector of the maximum weighted matching. Since it is impractical to solve Eq. 4.2. in polynomial time as ILP is NP-complete, we relax the integrability constraint on Δ ($0 \leq \Delta \leq 1$) so as to convert Eq. 4.2 into a standard linear programming problem (LP). The solution to the ILP problem is consistent to that of the LP one which provides an upper bound on the solution of Eq. 4.2. We solve the LP problem in terms of the classical Hungarian primal-dual algorithm [Sch02] given that the minimum vertex cover is dual to the maximum matching problem. In particular, the Hungarian algorithm maintains a cover and iteratively reduces the weight of the cover by finding maximum matchings on tight edges of the corresponding equality graph (perfect bipartite graph corresponding to G with auxiliary edges) (Kuhn-Munkres theorem) [Sch02]. Once such matchings are obtained for all pairs of selected frames of X_i, we link common proposals for two consecutive matchings in order to further expand the chain graphs (Fig. 4.2).

Remark 3. Such chain graphs represent the characteristics of the video sub-volumes which contribute to the recognition of the underlying human activity. We opt for the proposed bipartite graph matching based framework in contrast to simple region matching between the proposals since such naive matching in the Euclidean space is prone to false region correspondence which may affect the efficacy of the classifier. Since the chains can be of different lengths, we further employ a generic time-series feature pooling strategy to encode the appearance evolution of the underlying region proposals into fixed length vectors.

Remark 4. It is worth noting that the Hungarian algorithm is yet to be explored extensively in computer vision. However, it proves to have significant applications in areas involving assignment problems.

Time series pooling and concept generation. Specifically, the considered time-series pooling [RRM15] separately tracks the positive and negative slopes for each of the dimensions of the feature gradient vectors. The gradient is defined as the difference between the CNN features of a pair adjacent proposals (nodes) present in the chain. The positive and negative components are further averaged to obtain two 4096×1 dimensional descriptors. Each chain is finally represented by a vector of size 8192×1 by concatenating them. We rely on the time-series pooling since unlike the traditional max/average pooling, it respects the ordering of the frame-level contents. Further, considering the dimensions of the gradient vectors as one dimensional signals, it is possible to quantify the advancement (absence/presence) of abstract concepts over the frames with this feature representation. We further term such extracted descriptors as *concepts* (Fig. 4.1).

Fig. 4.3 provides a visual description of the mid level feature extraction process from a sample HMDB51 video as discussed here. In this figure, (a) shows the region proposals after initial subsampling as discussed in the main submission. (b) shows the way each region proposal in one frame is treated as a node and edges are formed between region proposals of 2 consecutive frame pairs to form a bipartite graph. Maximum weighted bipartite graph matching provides optimal solution for region

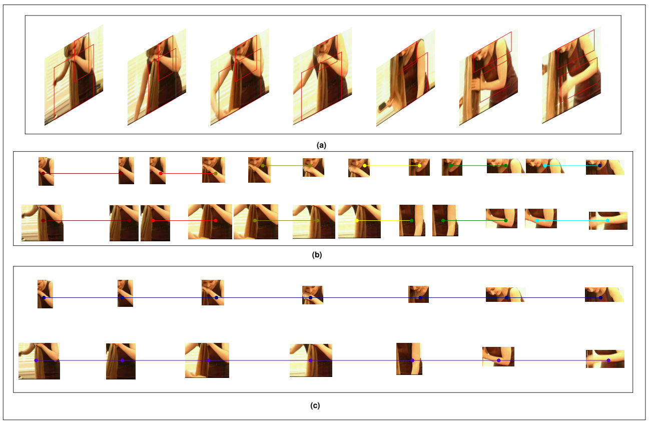


Figure 4.3: chain graph formation example for HMDB51. (a) Initial sets of motion salient region proposals. (b) pairwise bipartite graph matching. (c) chain graph formation.

correspondence. Connecting the solutions from each pair give the chain graph in (c). Fig. 4.3 validates visually our region proposal subsampling process. The resultant region proposals are not highly overlapping and generally contain a certain body part. Hence chain graphs formed subsequently follow the movements of specific parts of body. For example, in Fig. 4.3, region proposals in each frame correspond to hand, upper body legs respectively. Three chain graphs formed essentially follow the movements of these body parts. Similarly, region proposals corresponding to head and hand can be observed in different frames in Fig. 4.3. Also, maximum weight graph matching ((b) in Fig 4.3) based optimization greatly reduces chances of region mismatch, utilizing both CNN feature extracted from region proposals as well as their position in frames (for finding overlap). Fig. 4.4 is the visual description for the times series pooling used in our work. We pool on chains formed on local spatial regions across the frames. Also, by incorporating positive and negative gradient information, we encode local information across temporal dimension as well. This is in contrast with general convention of global pooling schemes such as max pooling, average pooling etc [WLSS16].

Remark 5. Differently from [ZNH$^+$15], concepts are automatically discovered without any assumption, and multiple concepts ($\{\widehat{c_i^l}\}_{l=1}^{\nu_i}$) are extracted from X_i. Further, the extracted concepts are highly non-redundant since they neither contain any common proposal nor refer to background contents. We also propose a simple dictionary learning strategy to encode the videos with respect to the concepts.

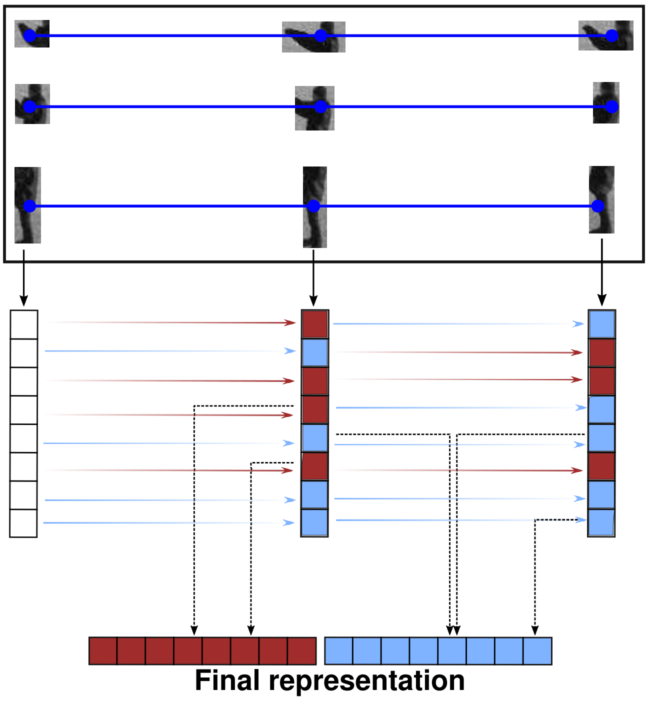


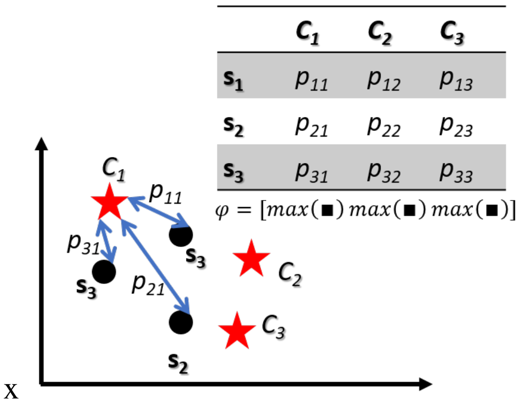
Figure 4.4: Example of time series pooling from chain graph (*concept*). Positive gradient is denoted by brown colour. negative gradient is denoted by blue colour. For a 4096 dimensional vector (in our work), positive and negative gradients are placed in their respective dimension (position in a vector). Concatenation of these two provides final 8192 dimensional vector.

4.3 Dictionary learning with probabilistic embedding

Dictionary construction. It is expected that for a given action, a certain body part repeats similar manoeuvres multiple times which further reappears in videos from the same category. It is worth noting that the extracted concepts mainly have two characteristics: 1) action specific and 2) shared since we primarily eliminate the presence of irrelevant concepts in the pre-processing stage (Sec. 4.1). In order to identify clusters of class oriented concepts, k-means with sufficiently large k on all concepts across videos of every category is separately performed and all the cluster centroids are concatenated to create a dictionary of size $Q = k \times C$. We further introduce a simple probabilistic embedding scheme to project the video level concepts in the space spanned by the dictionary codewords. The distance (ρ) between a given concept $\widehat{c}_i^l$ of X_i to the q^{th} codeword μ^q is computed in terms of the Gaussian RBF distance as:

$$\rho_i^{lq} = \exp(-\gamma ||\widehat{c}_i^l - \mu^q||_2^2) \tag{4.3}$$

where ρ is inversely proportional to the distance between $\widehat{c}_i^l$ and μ^q. We calculate the similarities of



	C_1	C_2	C_3
s_1	p_{11}	p_{12}	p_{13}
s_2	p_{21}	p_{22}	p_{23}
s_3	p_{31}	p_{32}	p_{33}

$\varphi = [max(■)\; max(■)\; max(■)]$

Figure 4.5: Example of video embedding into concept space: C_1, C_2 and C_3 are codewords (concept templates). s_1,s_2 and s_3 are interest points from a video. Similarities are calculated according to equation 1. Column-wise max-pooling operation gives the embedding ϕ of the corresponding video.

Eq. 4.3 between all the concepts of the video X_i and the set of codewords.

Proposed embedding. We retrieve an embedding ϕ_i^q of X_i for the q^{th} $(1 \leq q \leq Q)$ dimension by max-pooling the similarity measures between all the concepts of X_i and the q^{th} codeword as follows:

$$\phi_i^q = \max_l(\rho_i^{lq}) \tag{4.4}$$

Essentially, $\phi_i \in \mathbb{R}^Q$ represents a discriminative embedding of the video X_i; a vector of best similarity scores between the concepts and each of the Q codewords.

Remark 6. Popular encoding techniques such as VLAD or fisher vector (FV) are super-vector encoding and depending on the size of codebook, the size of the obtained encodings is extensively large. Given the small size of the datasets considered, the ratio between the number of samples and the data dimensionality is very less, which subsequently causes the overfitting of the classifier. The proposed encoding is devoid of such a problem since there is no hyper-parameter involved and we always obtain a fixed length feature vector of size 8192. Other work in [RBM17c] has demonstrated the effectiveness of this encoding technique even for low level features.

Remark 7. It should be noted that unsupervised k-means is used to select the initial set of codewords per category. However, further introduce a discriminative probabilistic embedding which ensures concentration of the videos belonging to each class. On the other hand, the construction of a single dictionary considering k-means on the videos of all the classes does not possess enough discrimination unless external supervision is deployed.

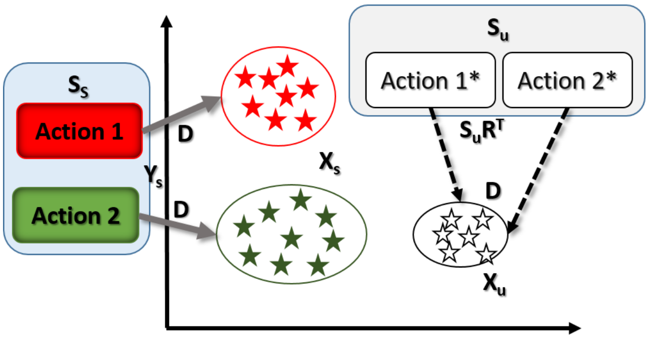


Figure 4.6: Example of ZSL. action 1^* and action 2^* are unseen class labels. R (along with D), determines the class assignment of elements of X_u either to action 1^* or to action 2^*. See also text.

Remark 8. In contrast to the exemplar classifier based embedding generally adopted for mid-level descriptors [ZNH$^+$15], the proposed encoding is simple, discriminative and does not require to model the responses of such classifiers. As an outcome, we obtain a Q dimensional embedding $(\phi_i^1, \phi_i^2, \ldots, \phi_i^Q)$ of video X_i . We use metric learning based multi-class SVM for classification and the proposed semi-supervised clustering based ZSL technique is mentioned below.

4.4 Action classification using zero-shot learning

For the task of ZSL, we consider that the training classes C_s ($C_s \subset \{1, 2, \ldots, C\}$) and test classes C_u ($C_u \subset \{1, 2, \ldots, C\}$) are mutually non-overlapping ($C_s \cap C_u = \emptyset$). Furthermore, the available seen and unseen instances for ZSL are of the form $\{(X_\alpha, Y_\alpha)\}_{\alpha=1}^{N_S}$ and $\{(X_\beta)\}_{\beta=1}^{N_u}$, respectively.

In order to learn a mapping (D) from label to visual embedding space jointly for both the seen and unseen data, we pose the ZSL problem as a semi-supervised clustering problem [SB16], where seen and unseen data are clustered together, utilizing available annotations of seen instances. The visual embedding (ϕ) of the previous step is considered for semantic embedding and the unseen classes are initially embedded based on the concepts extracted from the seen classes. Further, such embeddings are projected onto a metric space [Sug06] for better cluster discrimination. Given that the CNN features capture high-level discriminative abstractions, videos from a given class are expected to form a dense cluster in the metric space. Let $a_i \in \mathbb{R}^t$ be the signature of a seen class $Y_i \in C_s$ and the matrix $S_s \in \mathbb{R}^{t \times |C_s|}$ stores the signatures into columns: $S_s = [a_1, a_2, ..., a_{|C_s|}]$. Similarly, $S_u \in \mathbb{R}^{t \times |C_u|}$ represents the same for unseen classes (Fig. 4.6).

We construct an optimization function based on ridge regression for the underlying semi-supervised clustering problem. In particular, we intend to learn a low-rank transformation preserving the least square error bounds on seen classes. For samples belonging to unseen categories, we ideally seek cluster assignments (R) such that a generic cluster validity index is also optimized. Formally, we aspire to optimize the following cost function jointly on all the samples:

$$\min_{R,D} ||X_s - DY_s||_F^2 + \lambda_1 ||X_u - DS_u R^T||_F^2 + \lambda_2 ||D||_F^2 \tag{4.5}$$

where $X_s = [X_1, X_2, \ldots, X_{N_s}]$, $Y_s = [Y_1, Y_2, \ldots, Y_{N_s}] \in \mathbb{R}^{t \times N_s}$ and $R \in \mathbb{R}^{N_u \times |C_u|}$ is the class indicators matrix for unseen data. Eq. 4.5 signifies that we seek a class assignment for samples of unseen classes in such a way that linear transformation of both seen and unseen class signatures provide good representations for the samples.

Since Eq. 4.5 is not convex jointly on $\{R, D\}$, the function is optimized by an alternate descent technique which iteratively finds D and R keeping the other fixed. Given R, the problem becomes a simple multi-task ridge regression and there is a closed-form solution for D as follows:

$$D = (X_s Y_s^T + \lambda_1 X_u R S_u^T)(Y_s Y_s^T + \lambda_1 S_u R^T R S_u^T + \lambda_2 I)^{-1} \tag{4.6}$$

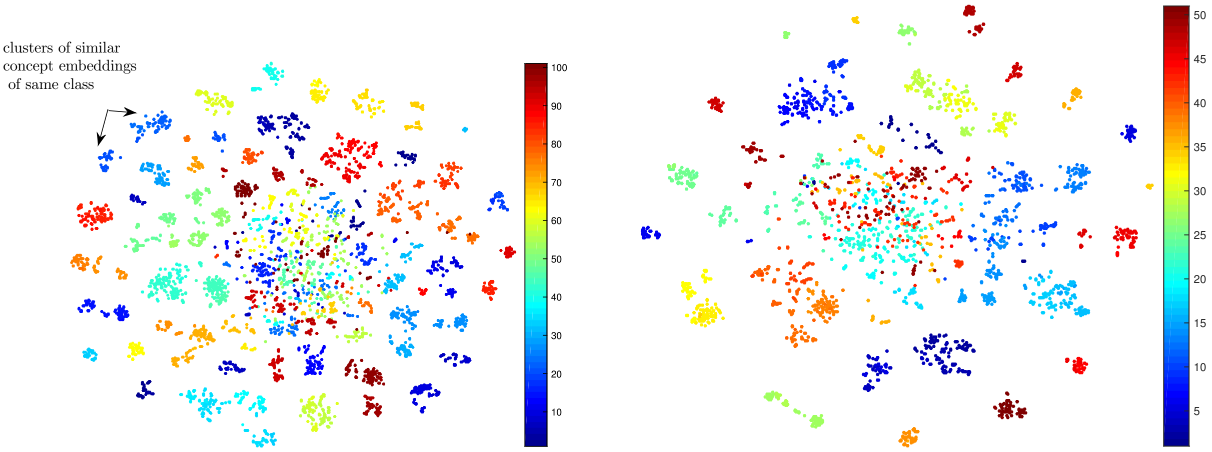
Fixing D, R can be solved by assigning each unseen sample to its nearest class. i.e. for a sample X_β from unseen class we find:

$$k' = \arg\min_k ||X_\beta - DS_{u(k)}|| \tag{4.7}$$

and subsequently assign $R_{\beta k'} = 1$. We use difference in the goodness of clustering (Dunn's index) [HVB00] for unseen instances between two consecutive iterations as the stopping criteria . The whole algorithm is shown in Algorithm 3.

Remark 9. Since we essentially learn an embedding (D) jointly from seen and unseen data, the framework is thus capable of inherently capturing any potential domain mismatch between both the source and target domains.

```
Algorithm 3: Zero shot learning
Input: $X_s$,$Y_s$,$X_u$, $S_s$, $S_u$
Output: $R$,$D$
Initialize $R$,$D$ ;
while $err \leq \eta$ do
    update $D$ using Eq. 4.6 for fixed $R$;
    update $R$ using Eq. 4.7 for fixed $D$;
    $err$ = difference between the goodness of target domain clustering for two iterations;
```



(a) UCF-101 (b) HMDB-51

Figure 4.7: 2D t-SNE visualization of video data after probabilistic embedding. Colorbar corresponds to the cluster labels. Best viewed in color.

4.5 Experiments

In this section, we primarily evaluate the robustness of the proposed descriptors and the encoding technique, respectively in terms of fully-supervised action recognition.

Datasets. We evaluate the performance of the proposed framework on four challenging datasets: KTH dataset [SLC04] consists of 599 videos of 6 actions performed by 25 subjects in 4 different scenarios including indoor, outdoor, changes in clothing and scale variations. UCF-101 [SZS12] contains 13320 videos related to 101 human action categories. They represent single action with substantial background clutter. HMDB-51 [KJG$^+$11] comprises of 6766 videos from 51 human action categories. They are primarily collected from realistic videos from movies, Youtube and other public repositories. Finally, Hollywood Human Action (HOHA) [LMSR08] consists of 430 videos from 8 action classes of realistic human actions in unconstrained videos such as in feature films, sitcoms, or news segments.

Implementation details. We use the RPN used in faster RCNN [RHGS15] for the initial region

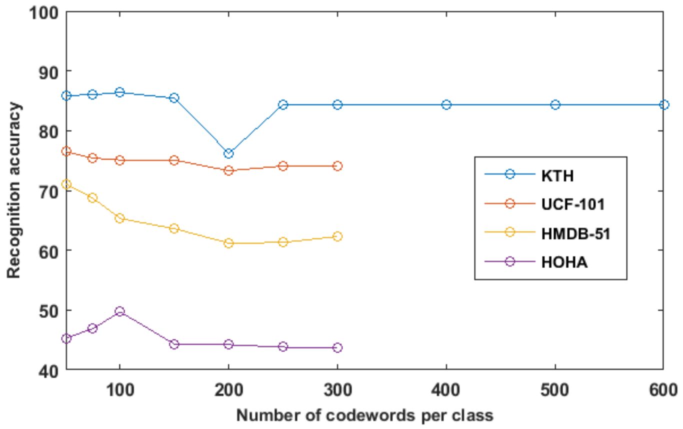


Figure 4.8: Trade-off between the number of selected codewords and classification accuracy (without metric learning).

proposals from video frames, approximately obtaining $5 - 50$ proposals from each frame after pre-processing (Sec. 4.1). For initial class specific codebook generation (Sec. 4.3), we select 100 code-words per class for KTH and HOHA, 50 words per class for UCF-101 and HMDB-51 datasets (Fig. 4.8). These values are selected by empirically. Probabilistic embedding therefore provides encodings of size $\mathbb{R}^{600}$,$\mathbb{R}^{5050}$, $\mathbb{R}^{2550}$ and $\mathbb{R}^{800}$ for KTH, UCF-101, HMDB-51 and HOHA respectively. Next, for standard multi-class action classification, we rely on the Local Fisher Discriminant Analysis (LFDA) [Sug06] metric learning based SVM framework for its better performance (discussed later in detail) and scalability. Gaussian RBF kernel is chosen in conjunction with SVM and both kernel hyper-parameter ($\approx 0.05 - 0.005$) and SVM trade-off parameter (≈ 1000) are fixed by 10-fold cross-validation.

For ZSL, both λ_1 and λ_2 (Eq. 4.5) are fixed by cross-validating on the seen classes. Proper convergence of the alternate optimization is largely dependent on the initialization of R. For optimal initialization of R and D, we obtain a projection matrix (D) solely based on the samples from the seen classes by solving the standard ridge regression problem. We subsequently cluster the unseen samples after projecting them on the new space (D) using k-means to initialize R accordingly.

Parameter tuning. Our pipeline is indeed a combination of several subtasks. Each of these subtasks are carefully chosen in order to fit our requirements, i.e. the ultimate goal of mining discriminative *concepts*. A brief description of parameter selection strategy is given below:

- *Motion salient region proposal detection:* In this task, the number of region proposals to extract is a crucial parameter. We tune this number to satisfy two criteria: a) Avoid redundancy b)

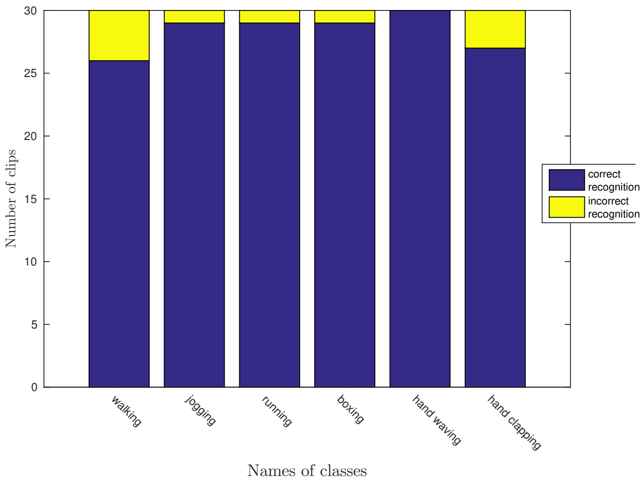


Figure 4.9: Bar graph for recognition for KTH. Notations: Yellow → wrongly classified, Purple → rightly classified

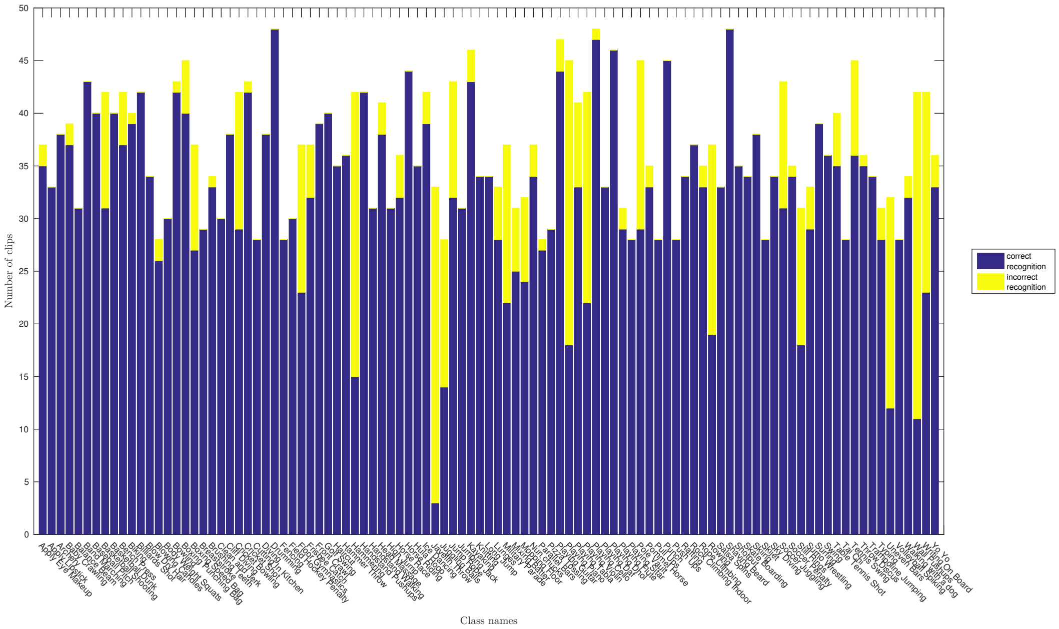


Figure 4.10: Bar graph for recognition for UCF101 (split 3). Notations: Yellow → wrongly classified, Purple → rightly classified

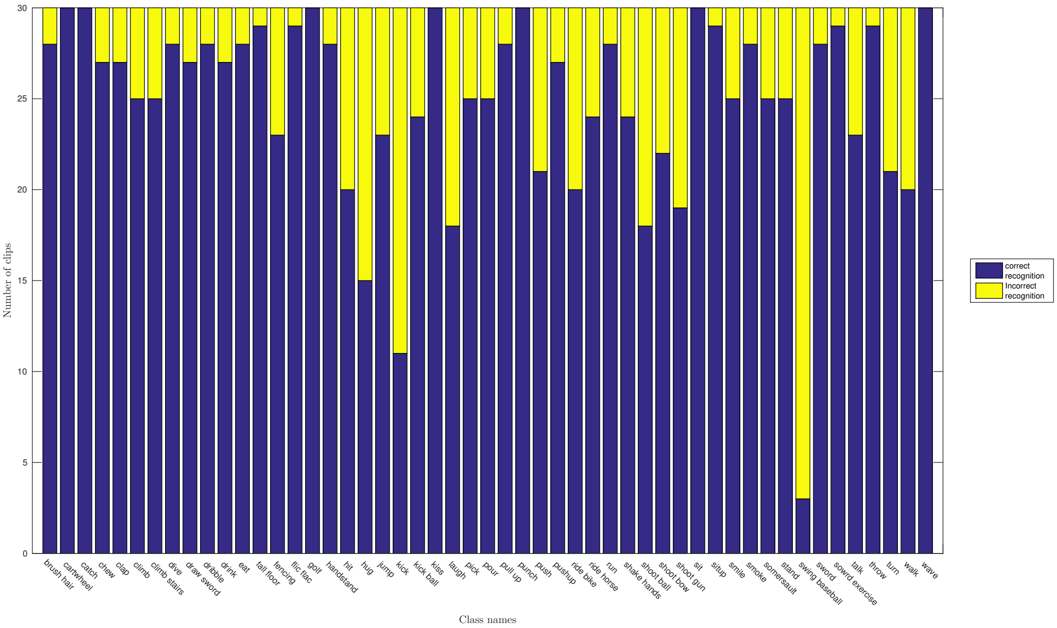


Figure 4.11: Bar graph for recognition for HMDB51 (split 3). Notations: Yellow → wrongly classified, Purple → rightly classified

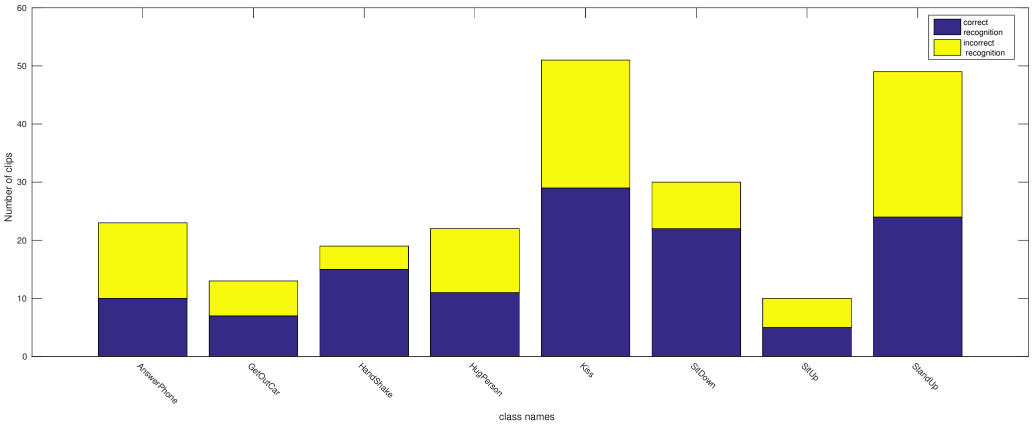


Figure 4.12: Bar graph for recognition for HOHA. Notations: Yellow → wrongly classified, Purple → rightly classified

Capture salient regions. In order to satisfy a) we do not extract too many region proposals (RP) so that the same region does not come under a large number RPs. In order to satisfy b) we visually observe on a number of (consecutive) image(s), the minimum number of RPs that capture the regions responsible for motion (e.g. hands, legs etc.). It should be noted that initial number of RPs proposed from the faster R-CNN is 2000. In order to come to a fixed optimal number, we utilize the notion of motion saliency. As discussed in section 3.1, we fix the saliency to be 0.6. This is done in order to make the process to be semi supervised (semi supervised because we fix the threshold of motion saliency yet do not necessarily rely on visual inspection to fix the number of RPs) and adaptive. We observe that number of RPs selected are different for different datasets. For example, for KTH there were 10 RPs/image in average over all the images but in HMDB51, this number was increased to 50.

- *Bipartite graph matching and time series pooling:* Our bipartite graph matching and time series pooling stages are hyper parameter free.

- *Number of code-words for probabilistic embedding:* For this, we fixed the other settings (RP selection, pooling, LFDA and SVM) and performed supervised classification on the validation sets of the respective datasets. Fig. 8 shows this. Please note that we could not experiment with high number of per class codewords since it would result in probabilistic embedding of really high dimensionality and consequently make the rest of the process rather cumbersome.

- *LFDA and SVM:*The output dimensionality is a key parameter which we fix by cross-validation together on the go with the parameters of SVM (RBF kernel hyperparameter and SVM regularization parameter). Please note for different datasets, the sets of parameters were different.

- *Zero shot learning:* We create the setup for our ZSL experiments by randomly splitting the dataset in 50%-50% for test-train sets. Further, we tune the parameters λ_1 and λ_2 by creating validation sets on the training set and applying algorithm 1. We try to maintain a ratio of 50%-50% in creating the validation set out of our training sets.For KTH, we have 3 training classes, hence we create a split of 2-1 class(es) for validation. For each split (since we have 10 random splits) we repeat this. This means for each random split we essentially have different values for λ_1 and λ_2.

At each subtask (except zero shot learning), the parameters are tuned keeping the other parts of the pipeline constant. Classification parameters were tuned using 10 fold cross-validation with (almost) 90%-10% split on the training set.

Visualization of the extracted descriptors. For visual confirmation of our results we use t-SNE [MH08]. For example, Fig. 4.14 depicts the 2-dimensional visualizations of the extracted concepts generated from t-SNE [MH08] for UCF-101 and HMDB-51. It can be observed that the concepts form locally dense clusters and concepts from different categories can easily be distributed. This essentially is due to high discriminative nature of the proposed encoding obtained based on the concepts.

Effectiveness of bipartite matching technique. We carried out experiments to compare the effectiveness of the sub-graph matching algorithm we used in our pipeline with two other commonly used matching techniques:

1. Ford-Fulkerson Algorithm [EK72]
2. HopcroftKarp Algorithm [Gar85]

Ford-Fulkerson algorithm is primarily used in case of maximum flow problem and can be extended to solve the maximum bipartite matching problem. On the other hand, Hopcroft-Karp algorithm solves bipartite matching problem using a maximum matching, a matching of maximum size (maximum number of edges). In order to compute the efficiency of our matching strategy against the ones mentioned above, we compute the percentage of correct matching in order to create meaningful concepts. Ideally, if similar region proposals (part of body) is tracked in consecutive frames, the difference of CNN feature values will be small, whereas if the matching is faulty (i.e. dissimilar body parts are matched) the difference will rather be high. We first calculate the difference of the consecutive region proposals in a chain of region proposals (concepts). We average them over number of region proposals in the chain. We take the average of all such values over all possible chains in a video. Table 4.1 shows a comparison of the methods on all 4 datasets. The reported values are mean over all the videos in each of the datasets.

From the table it is evident that our matching provides the least amount of error in creating the chain of region proposals which in turn proves the effectiveness of our matching approach. Especially

Table 4.1: Comparison of our bipartite matching approach with other methods.

Method	**KTH**	**UCF-101**	**HMDB-51**	**HOHA**
Ford-Fulkerson Algorithm	10400	25650	34800	100750
Hopcroft-Karp Algorithm	9800	21500	37600	98750
Ours	**1050**	**9700**	**13200**	**34050**

in HMDB-51 and HOHA, our matching performs considerably better than both other algorithms compared.

Table 4.2: Comparison of our probabilistic embedding technique with other embedding techniques (in %).

Method	**KTH**	**UCF-101**	**HMDB-51**	**HOHA**
VLAD	78.9	65.5	53.3	39.8
FV	82.1	69.2	55.1	41.4
LLC	80.6	65.1	54	39.8
Probabilistic embedding	**86.4**	**76.5**	**71.1**	**49.7**

4.5.1 Action recognition

The goal of this experiment is to show the efficiency of our mid level encoding through supervised action classification performance.

Experimental protocol. For the KTH dataset, 25-fold leave-one-group-out is used whereas for the remaining datasets, the available train-test splits are considered. We report the performance of our framework in terms of mean classification accuracy.

Evaluation. First, we compare the proposed probabilistic embedding to three standard feature encoding techniques used in conjunction with the extracted concepts: Vector of Linearly Aggregated Descriptors (VLAD), Fisher's vector (FV) [CLVZ11], and sparse locality constrained linear coding (LLC) [WYY$^+$10]. We find that the proposed encoding sharply outperforms the others by a substantial margin in terms of classification accuracy. For example, Fisher's encoding with 100 Gaussians provides best classification performance among the above mentioned three encoding techniques for

Table 4.3: Comparison of our supervised classification accuracies with that of the recent literature (in %).

Type	Method	KTH	UCF-101	HMDB-51	HOHA
Local	STIP+BoW [Lap05]	91.8	43.9	20.2	34.8
	DT [WKSL13]	94.1	-	46.6	28.8
	IDT [WS13]	-	85.9	57.2	41.8
Mid-level	Graph MIL [YL16]	-	-	-	56.51
	Action parts [RKS12]	-	-	-	40.1
CNN	3D CNN [KTS^{+}14]	-	63.3	-	-
	C3D (1 net) [TBF^{+}15]	-	82.3	-	-
	Spatial CNN [SZ14a]	-	73.0	40.5	-
	Temporal CNN [SZ14a]	-	83.7	54.3	-
	Two Stream Net [SZ14a]	-	**86.9**	58.0	-
	CNN+LSTM [VLS16]	-	83	57	-
Pooling (motion)	Average pooling [WLSS16]	-	80	50.9	-
	Max pooling [WLSS16]	-	80.2	50.6	-
	Temporal PP [WLSS16]	-	81.6	54.7	-
	Order aware [WLSS16]	-	82.1	55.0	-
Ours	SVM	*86.4*	*76.5*	*71.1*	*49.7*
	LFDA + SVM	**94.3**	*85.1*	***82.1***	***57.35***

Table 4.4: Recognition accuracies for the three splits of UCF101 and HMDB51.

UCF-101				
method	**split 1**	**split 2**	**split 3**	**average**
SVM	73	77.9	78.6	76.5
SVM+LFDA	77.7	88.7	88.9	85.1
HMDB-51				
method	**split 1**	**split 2**	**split 3**	**average**
SVM	70.3	68.9	74.1	71.1
SVM+LFDA	82.1	81.5	82.4	82.1

the KTH, UCF-101, HMDB-51 and HOHA with 82%, 69%, 55% and 41%, respectively, which are considerably lower than ours (Table 4.2). Another point to consider is the fact that all the encodings compared here, considers supervectors of high dimensionality, given the initial size of our feature vectors ($\mathbb{R}^{8192}$) and it is computationally more expensive to handle these encodings. Apart from that, evolution of the classification performance for different number of codewords is shown in Fig. 4.8. It can be observed that for both UCF-101 and HMDB-51, a small number of codewords produce better classification outcomes, whereas the performance gradually degrades for a larger number of codewords. This can be attributed to the fact that concepts are spread across the feature space, hence selecting in more number implies these class specific code words are also more spread across the feature space. Thus class-secificity of these code words are diluted and they become rather ambiguous

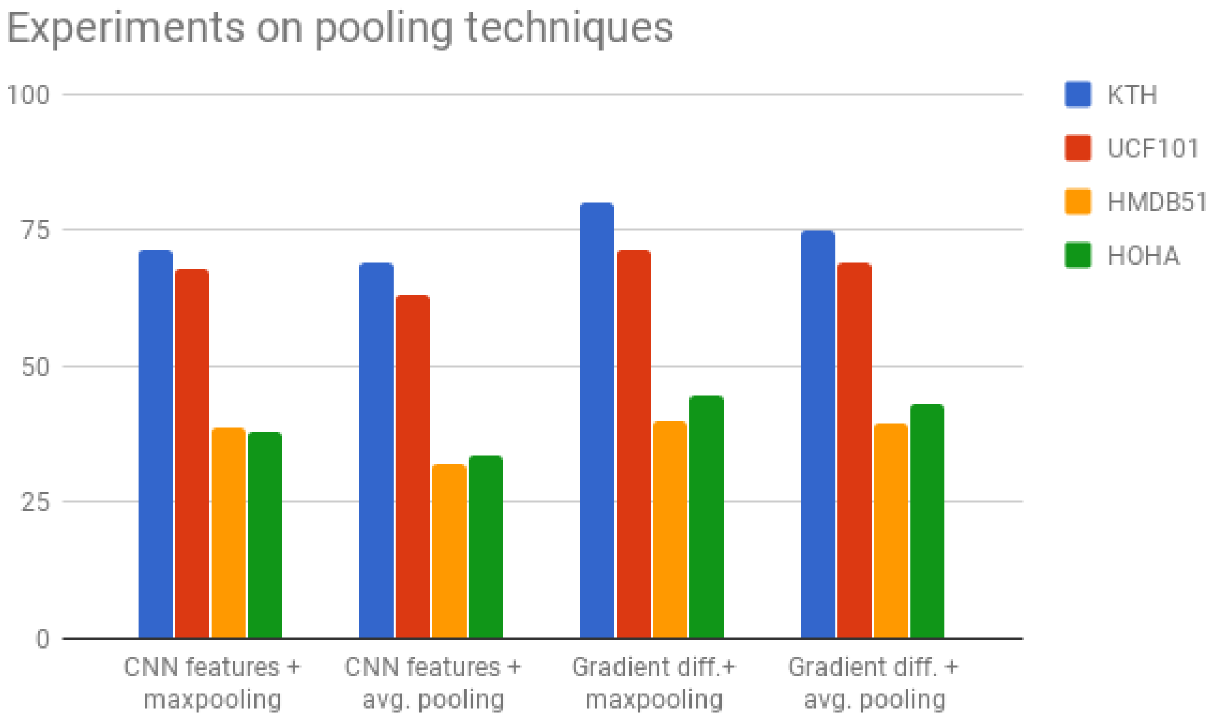


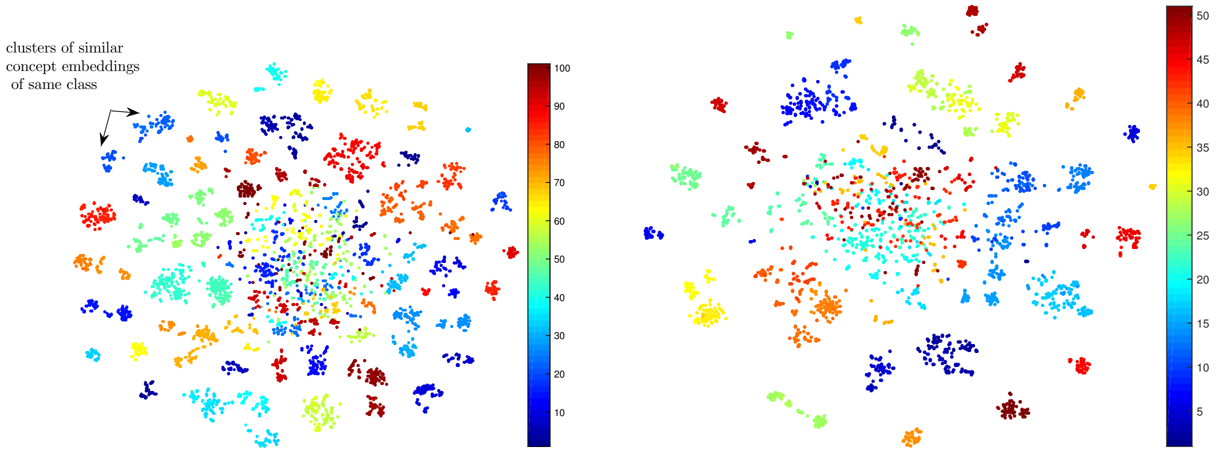
Figure 4.13: Experiments on pooling techniques.

(Fig. 4.14).

Effect of time series pooling. In order to validate the effectiveness of our pipeline as compared to just the effect of deep features, we conduct the following experiment. We extract CNN features per frame and the video level descriptors are calculated by max and average pooling the :

- frame level CNN descriptors
- gradient features between each pair of frames.

In both the cases, the obtained accuracies are substantially less for all 4 datasets (Fig. 4.13). Experiments reveal that maxpooling fares better average pooling for both frame level CNN and gradient features. But even in that case, time series pooling performs much better than max pooling. For example, for KTH dataset, frame level CNN feature + maxpooling gives an accuracy of 71.3% and gradient feature+maxpooling gives 80.1% accuracy. Whereas time series pooling gives 86.4% accuracy (Table 1). **Effect of metric learning (LFDA).** Fig. 4.14 shows that class specific concepts are distributed throughout feature space in a locally dense manner. LFDA is known for its ability to



(a) UCF-101 (b) HMDB-51

Figure 4.14: 2D t-SNE visualization of video data after probabilistic embedding. Colorbar corresponds to the cluster labels. Best viewed in color.

maximize between-class separability while preserving within-class local structure [Sug06]. Hence, LFDA, being a fast, robust, local metric learning technique, can be adopted to our setup effectively. Consequently, given the discriminative nature of our encoding, which highlights class-specific concepts, supervised LFDA further enhances the class separation significantly, thus providing a high leap in accuracy. Our system exhibits classification performance of $86.4\%, 76.5\%, 71.1\%$; and 49.7% and $94.3\%, 85.1\%$, 82.1% and 57.35% for KTH, UCF-101, HMDB-51 and HOHA, without and with the use of metric learning (Table 4.3) respectively.

Comparison to the literature. Table 4.3 summarizes our performance with respect to recent literature. We compare our results with low level local features, mid level features, as well as some high level CNN based models. We further perform a comparative analysis with other recent deep feature pooling techniques used in action recognition. Amongst local approaches, the improved dense trajectory based features with HOG, HOF an MBH descriptors [WS13] produce recognition accuracies of 85.9%, 57.2% and 41.2% for UCF-101, HMDB-51 and HOHA, respectively. On the other hand, `3D-CNN` [KTS$^+$14] trained on 1-million sports videos produces 63.3% accuracy for UCF-101 after fine-tuning. The popular two-stream networks [SZ14a] also exhibit competitive performance for UCF-101 and HMDB-51 with 86.9% and 58%, respectively. Amongst different pooling strategies, order-aware pooling [WLSS16] is found to provide best performance, i.e. 82.1% and 55% for UCF-101 and HMDB-51, respectively by incorporating ranking paradigm in CNN training.

We also provide the bar graph showing per class accuracies for KTH, UCF101, HMDB51 and HOHA

Table 4.5: Comparison of recognition accuracies of the ZSL techniques (in %).

Method	**KTH**	**UCF-101**	**HMDB-51**	**HOHA**
Random guess [XHG15]	-	2.0	4.0	-
DAP [LNH14a]	-	2.2±0.5	-	-
IDP [LNH14a]	-	6.9±1.1	-	-
NN Self-Training [XHG15]	-	**18.6±2.0**	**21.2±3.0**	-
Sparse code + ZSL [KXFG15b]	-	14.0±1.8	-	-
Ours	*40 ± 4.0*	*17.8±0.8*	*20.1±0.6*	8.0 ±3.4

in Fig. 4.9,4.10, 4.11 and 4.12 respectively. We obtain state of the art recognition accuracy in HMDB51. Although for most of the classes recognition is commendable (we got 100% class accuracy in 7 classes- ’cartwheel’, ’catch’,’golf’, ’punch’,’sit’, ’wave’), for the classes ’ swing baseball’ and ’kick’, recognition is comparatively much lower. 27 out of 30 clips in ’swing baseball’ (90% of clips) and 19 clips from ’kick’ (63%) are misclassified.

Our model exhibits consistently impressive performance for vastly different action types. In particular, for the challenging HMDB51 (daily activity), where the action videos include substantial human object interaction as well as cluttered backgrounds, we report state of the art performance, outperforming the best accuracy by more than 20% (71.1% & 82.1% vs 58%). Given the semantic richness of extracted concepts, which is attributed to the motion salient region proposals, our feature descriptors implicitly encode complex human object interactions. Similarly for UCF 101 (sports centric), our results are comparable to the standard two-stream net [SZ14a] and CNN+LSTM [SMS15], thus emphasizing the discriminative nature of our concepts without the need of costly training of deep network.In particular, Two-stream nets trains deep networks to encode appearance and motion, whereas we encode both at the same time without any extra deep training overhead.

In comparison to other mid level approaches [RKS12, YL16], which require clustering of large volume of trajectories (often redundant and irrelevant), our detected concepts are non overlapping, focussed on action specific components, as exhibited by the performance on the HOHA, dominated by camera and background motion (see Table 4.3). Overall, these results on datasets of varied nature and size establish the robustness as well as the scalability of our proposed pipeline for .

4.5.2 Zero shot classification

The aim of this experiment is to demonstrate the ability of the concepts as a replacement for computationally expensive semantic attributes for the action classes. A proper way to highlight this is through a ZSL setup where there is a need for semantic understanding of the related classes.

Experimental protocol. For ZSL, we randomly split all the available classes into two groups (seen and unseen group) where each of the groups consists of half of the available classes. We consider 10 such random splits for each dataset and report the average classification performance.

Comparison to the literature. We compare the proposed semi-supervised clustering based ZSL technique with that of the traditional attribute based [LNH14a] and semantic embedding based techniques in Table 4.5. Attributes are an important factor in ZSL give the fact it forms the basis for semantic space. They are either predetermined from prior experience or learnt through explicit supervised learning. Automatic learning of attributes is an expensive process and does not always guarantee proper understanding of semanticity related to a class label. The performances of both direct and indirect attribute prediction (DAP, IDP [LNH14a]) is very little for the UCF-101 data when 51 classes are considered to be seen and the remaining 50 classes are used for testing (2.2% and 6.9%, for DAP and IDP respectively). On the contrary, the proposed clustering based approach does not require the application of manually annotated attributes or extensive training of deep models for word embedding [GL14]. Our results closely resemble with that of the costly semantic embedding based approach of [XHG15] for both UCF-101 and HMDB-51 (only about 1% less). Another important aspect of experiments related to ZSL is the stability, i.e. how consistent the process is when the sets of known and unknown classes are changed. This is generally expressed in terms of the standard deviation as measure of tolerance. In this regard, we exhibit more stability over different splits (standard deviation $\leq 1\%$) as compared to [XHG15] (standard deviation 2%-3%). This stability property establishes the global and invariant nature of the extracted concepts.

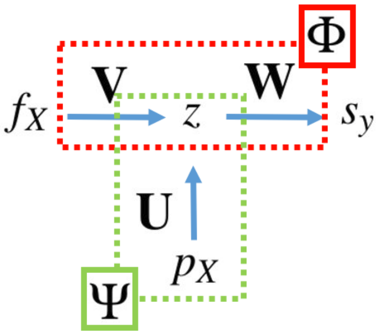
Chapter 5

Visually-Driven Semantic Augmentation for Zero Shot Learning

In spite of supirior performance of mid level features for traditional supervised as well as zero-shot scenarios, there is still progress needs to be made in order to achieve generality, which essentially guarantees success in situations with small number of data. To that regard, zero-shot learning demands special attention. In particular, we concentrate on the topic of zero-shot image recognition. Different from the contemporary works that learn an embedding from visual to semantic spaces and carry out zero shot recognition based on the semantic representation of categories, we posit that semantic and visual spaces are equally important, providing complementary information. In this chapter, we present a novel optimization pipeline, called Visually-driven Semantic Augmentation (VdSA), to augment semantic embeddings by means of visual cues extracted from soft labels. Before discussing the details of the method, some notations are as follows.

Let X denote a generic instance data to be classified (in this work, an image). In ZSL, the task is to train a model with full supervision using instances belonging to a given set of seen classes $\mathcal{Y}_{\text{seen}}$. During testing, such model is transferred on a *different* set of unseen classes $\mathcal{Y}_{\text{unseen}}$. As usually done in ZSL [XSA17], one assumes that seen and unseen classes are disjoint sets, that is, $\mathcal{Y}_{\text{seen}} \cap \mathcal{Y}_{\text{unseen}} = \emptyset$.

For each instance X, we compute its visual signature $f_X \in \mathbb{R}^m$ in a visual embedding space whereas,



10.3

Figure 5.1: Overview of the proposed ZSL method. In the first stage, the visual embedding f_X is first mapped into the latent attribute space z and, afterwards into the semantic embedding s_y: this mapping is performed by Φ, which depends by parameters $\mathbf{V}$ and $\mathbf{W}$. The second stage is accomplished by the auxiliary mapping Ψ, which depends by parameters $\mathbf{U}$.

for each class y, we compute a semantic representation $s_y \in \mathbb{R}^n$ in semantic embedding space. In other words, one can think of f_X as a deep feature from a CNN [KSH12b, SZ14b], and s_y can be a distributed word embedding (such as word2vec [MSC+13]).

As a learning process, ZSL can be framed in stages. In training, we *only* use the seen classes $\mathcal{Y}_{\text{seen}}$ to learn a transformation Φ taking visual signature f_X as input and outputs a semantic representation $\Phi(f_X)$. In testing, we predict the unseen class of a testing instance $\widetilde{X}$ by selecting $\widetilde{y} \in \mathcal{Y}_{\text{unseen}}$ according to the criterion

$$\widetilde{y} = \arg \min_{y \in \mathcal{Y}_{\text{unseen}}} \|\Phi(f_{\widetilde{X}}) - s_y\|_2, \tag{5.1}$$

where $\|\cdot\|_2$ stands for the Euclidean norm.

5.1 Method

Our proposed Visually-driven Semantic Augmentation (VdSA) builds upon the previous setup of learning a transformation Φ from visual to semantic embeddings (that is, from f, computed from X, to s, computed from y).

Our approach is to augment semantic information through the representation p_X extracted from the softmax output vector of a deep neural network (trained for image classification tasks on the seen classes only), fed with X. Let us note that since we are learning a mapping $\Phi\colon f \to s$ where the semantic representation s is the output (and not the input), one cannot simply concatenate soft labels to attributes/DWEs features and learn a (supposedly) better function Φ. Therefore, the way of inserting soft labels in a ZSL pipeline is not trivial a priori.

In this work, we tackle this issue and propose a novel ZSL pipeline. The key idea consists of introducing a intermediate layer z represented by latent attributes, instead of having a direct mapping Φ from f_X to s_y [ZS16a, PTX$^+$16, XHG16, JWS$^+$17b, KXG17, HMS17]. The rationale is that we take advantage of z in order to do a compression while fusing the visual embedding f_X and soft-labels p_X, ultimately integrating the semantic cues which are extracted by softmax operators from visual data directly.

Formally, let F_{seen} denote the visual data, i.e., $m \times d$ matrix which stacks by columns all the visual features f_{X_k} computed from training data instances X_k, $k = 1, \ldots, d$, belonging to the set of seen classes . Similarly, let S_{seen} be the $n \times d$ matrix whose k^{th} column gives the semantic representation s_{y_k} relative to the seen training class y_k to which X_k belongs to. As a baseline, we consider the following model based on latent attributes [ZS16a, PTX$^+$16, XHG16, JWS$^+$17b, KXG17, HMS17]:

$$\min_{\mathbf{W},\mathbf{V},Z} \|S_{\text{seen}} - \mathbf{W}Z\|_F^2 + \alpha\|Z - \mathbf{V}F_{\text{seen}}\|_F^2, \tag{5.2}$$

where $\alpha > 0$, $\|\cdot\|_F$ stands for the Frobenius norm, Z stacks by columns all the latent attributes, and the two sets of parameters $\mathbf{W}$ and $\mathbf{V}$ represent the mapping Φ (see Figure 5.1)[1].

In order to make latent attributes aware of semantic information distilled from visual data by means of soft labels, we introduce the auxiliary function Ψ which enriches z using p_X. Therefore, our proposed ZSL framework rewrites as follows

$$\boxed{\min_{\mathbf{W},\mathbf{V},\mathbf{U},Z} \|S_{\text{seen}} - \mathbf{W}Z\|_F^2 + \alpha\|Z - \mathbf{V}F_{\text{seen}}\|_F^2 + \beta\|Z - \mathbf{U}P_{\text{seen}}\|_F^2} \tag{5.3}$$

where $\alpha, \beta > 0$ and P_{seen} stacks all soft labels by columns. In (5.3), the usage of the auxiliary mapping Ψ helps Φ in optimizing z as to 1) map visual into semantic embeddings, and 2) take advantage of the auxiliary semantic information extracted by means of soft labels. This constitutes a novel approach, never investigated in previous ZSL methods [LNH09, FEHF09, MSN11, WM10, WJ13, PTX$^+$16,

[1] Although in principle the map Φ can be arbitrary, here we consider it to be a composition of linear functions as commonly done in several mainstream ZSL approaches such as [FCS$^+$13, RPT15a, ARW$^+$15, JWS$^+$17b, KXG17].

Dataset	No. of instances	No. of attributes	No. of seen/unseen classes
aP&Y [FEHF09]	15339	64	20/12
AwA [LNH09]	30475	85	40/10
CUB-200 [WBW+11]	11788	312	150/50

Table 5.1: Description of the ZSL benchmark datasets used in experiments

ZS16a, JWS+17b], in which semantic information driven by visual data is actually disregarded. Since we are using P_{seen}, soft label information from a discriminative network, this also contributes towards equation 5.3 to be discriminative.

Optimization. The objective function (5.3) is not jointly convex with respect to all the variables $\mathbf{W}, \mathbf{V}, \mathbf{U}, Z$. But it becomes convex as long as one optimizes over one variable while fixing the others. In fact, if one uses alternating optimization to solve (5.3), when either optimizing over $\mathbf{W}$ (resp. $\mathbf{V}$ or $\mathbf{U}$), only the first (resp. second or third) term in (5.3) is considered. More importantly, solving for $\mathbf{W}, \mathbf{V}$ and $\mathbf{U}$ separately is a least square fitting for which a closed-form solution exists (due to the usage of the Frobenius norm, one can use normal equations [Bis06b]).

Similarly, the optimization for Z can be done in closed-form by the following change of variables: Z can be found by minimizing the objective $\|A - BZ\|_F^2$ (while freezing $\mathbf{W}, \mathbf{V}$ and $\mathbf{U}$) where the matrices A and B are given as the 3×1 block-column matrices composed of $S_{\text{seen}}, \mathbf{V}F_{\text{seen}}, \mathbf{U}P_{\text{seen}}$ and $\mathbf{W}, I, I$, respectively (I denotes the identity matrix of suitable size). We impose a normalization on our trainable parameters; each column in $\mathbf{W}, \mathbf{V}, \mathbf{U}$ or Z has $\|\cdot\|_2$-norm upper bounded by 1. Even with such constraint, we still achieve closed-form solution in our alternated optimization thanks to Lagrangian multipliers [Rud76]. In the inference stage, predictions are done using eq. (5.1).

5.2 Experimental Results

In this Section, we empirically demonstrate the effectiveness of the VdSA approach. More precisely, after providing technical details for reproducibility (in §5.2.1), we provide an ablation study to assess the effect of soft labels (in §5.2.2) and, finally, we report a comparison with the state-of-the-art methods on three benchmark datasets (in §5.2.3).

5.2.1 Details for reproducing the experiments

We validate our proposed semantic augmentation by performing experiments on three ZSL benchmark datasets for the task of object recognition and image classification: aPascal & aYahoo (aP&Y) [FEHF09], Animals with Attributes (AwA) [LNH09], Caltech-UCSD Birds-200-2011 (CUB-200) [WBW+11]. Details of these datasets are shown in Table 5.1 in terms of number of samples/attributes. To partition seen and unseen classes, we adopt the splitting criteria commonly used in the literature [XSA17].

Visual Embedding. As done in [JWS+17a, QLSH17, ZS15], we encode each image with the 4096-dimensional `fc7` feature vector extracted from a VGG-19 model [SZ14b] pre-trained on ImageNet [DDS+09].

Semantic Embedding. We consider binary attributes - manually annotated - provided with each dataset. In addition, we also use continuous distributed word embeddings (DWEs). To this end, we use word2vec [MSC+13], exploiting a pre-trained model[2] to cast each class name into a 300-dimensional vectorial representation.

Soft labels. We generate soft labels by extracting softmax outputs generated after fine-tuning AlexNet [KSH12b]. To do so, we run ADAM optimizer for 5000 iterations with a fixed learning rate of 0.001 and dropout regularization in the AlexNet fully connected layers (with a dropout rate of 0.5). In each dataset, we setup soft labels for both seen and unseen classes, but, in order to follow a fair ZSL protocol, we supervise back-propagation on the soft labels *only* for seen classes with *only* seen class data. Therefore the entries of soft labels which correspond to unseen classes are not directly optimized with supervision but, rather, we expect that the network itself will populate them implicitly. In this way, the network will mine the similarities among different classes by itself, ultimately facilitating the transfer of knowledge (more details in §5.2.2).

Latent attributes. For the alternate optimization of our objective function (5.3), we used a uniform random initialization (in the range $[-12, 12]$) for all parameters. We did not notice any remarkable difference in the results of the optimization depending on the order with which variables are optimized

[2] https://github.com/chrisjmccormick/word2vec_matlab

Binary attributes annotated by humans			
Dataset	baseline (5.2) + **A**	VdSA + **A** + **H**	VdSA + **A** + **S**
aP&Y	50.6	51.1	**51.7**
AwA	78.2	**81.0**	78.4
CUB-200	55.0	55.1	**56.7**

Continuous distributed word embeddings (DWEs) learnt from a text corpus			
Dataset	baseline (5.2) + **W**	VdSA + **W** + **H**	VdSA + **W** + **S**
aP&Y	41.7	42.4	**43.2**
AwA	51.6	54.1	**56.7**
CUB-200	29.8	33.9	**34.8**

Combination of binary attributes and continuous DWEs			
Dataset	baseline (5.2) + **A** + **W**	VdSA + **A** + **W** + **H**	VdSA + **A** + **W** + **S**
aP&Y	49.1	49.7	**53.6**
AwA	76.1	79.8	**80.6**
CUB-200	54.6	56.7	**59.7**

Table 5.2: Results of our ablation study. We present the multi-class classification accuracies (in percentage %) for the unseen classes used in testing - best result for each row in boldface. We compare a baseline latent attribute model (5.2) with our proposed model (5.3) for visually-driven semantic augmentation. In both cases, we evaluate with different semantic embeddings: either binary manually annotate attributes (**A**) or distributed word embeddings (**W**). Also, we compare augmentation by exploiting both hard labels (**H**) - given as ground truth - and soft-labels (**S**) - estimated from the softmax operator of VGG-19 [SZ14b].

- and therefore we optimized in the order $\mathbf{W}$, $\mathbf{V}$, $\mathbf{U}$, Z. For the latent attributes, we fixed their number to be 300 for AwA and 500 for CUB-200 and aP&Y respectively. The values of α and β as well as the number of latent attributes are determined after five fold cross validation, using seen classes only.

5.2.2 Ablation Study

In this Section, we present an ablation study to evaluate the effect of our visually driven semantic augmentation. To do so, we compare our latent attribute augmentation (5.3) with the baseline (5.2), without augmentation [HMS17, XHG16]. We consider different semantic embeddings: binary attributes - **A**, word2vec distributed word embeddings - **W**, as well as a concatenation of the two (**A** + **W**). In addition, we also compare our proposed visual augmentation based on soft labels with another approach which, instead of the prediction given by softmax operators, directly takes into account ground truth labels in the form of one-hot encodings.

The results of our ablation study are reported in Table 5.2.

Discussion. We register a common trend in all three datasets when using either **A**, **W** or **A** + **W**. That is, the ZSL testing accuracies always increase when switching from the baseline (5.2) - second column - to our proposed visually driven semantic augmentation (5.3) using soft labels **S** - fourth column. For instance, $+4\%$ on CUB-200 when using **W** and $+5\%$ in the **A+W** case. This clearly states that our proposed augmentation 1) extracts semantic patterns from visual data and 2) combines it with the semantic information of attributes/DWEs.

Hard vs. soft labels. In principle, one can expect that hard labels **H** are better than soft ones **S** since those one-hot vectors are given as ground truth annotations (for the seen classes). On the contrary, soft labels **S** are just predictions and therefore they can be wrong. On the contrary, when switching from hard to soft labels - Table 5.2, third and fourth columns respectively, accuracies grow systematically. This can be explained by the fact that soft labels are more informative with respect to hard ones since they convey the confidence with which a given instance is estimated to belong to each class [TSS07]. In our work, we build upon this idea to show that such concept can be favourably embodied in ZSL: soft labels, even if trained on the seen classes only, implicitly learns similarity patterns between seen and unseen classes and ultimately boost the transfer in between.

Combining various semantic information. From the results in Table 5.2, one can see how combining attributes and DWEs in (5.2) is not straightforward. This is clear from the fact that DWEs (such as word2vec) are surrogates for the attributes used to conveniently circumvent manual annotation. However, in terms of performance, **A** is arguably better than **W** and our experimental findings confirm that. Moreover, a concatenation of the two does not enrich the semantic information exploited by the ZSL model to transfer from seen to unseen classes. On the contrary, the performance systematically deteriorates: when switching from **A** to **A+W**, the baseline (5.2) drops from 50.6% to 49.1% on aP&Y and (5.3) drops from 81.0% to 79.8% on AwA.

Remarkably, our proposed semantic augmentation based on soft labels shows a completely different behaviour: when concatenating **A** to **W**, we sharply improve with respect to using **A** only. Precisely, in the 4^{th} column of Table 5.2, our method achieves an improvement on $+1.9\%$ on aP&Y, $+2.2\%$ on AwA and $+3\%$ on CUB-200.

As the consequence of the solid potential showed by our method in this analysis, in the next Section,

Method	**Semantic Embedding**	**Datasets**		
		AwA	**CUB-200**	**aP&Y**
DAP [LNH14b]	**A**	57.2	-	38.16
DeVise [FCS$^+$13]	**A**	56.7	33.5	-
SJE [ARW$^+$15]	**A**	66.7	50.1	-
Kodirov et al. [KXFG15a]	**A**	73.2	39.5	-
ESZSL [RPT15a]	**A**	75.3	47.2	24.2
SSE [ZS15]	**A**	76.3	30.4	46.2
MTL [YH15]	**A**	63.7	32.3	-
SynC [CCGS16]	**A**	72.9	54.7	-
Bucher et al. [BHJ16]	**A**	77.3	43.3	**53.2**
JLSE [ZS16a]	**A**	80.4	42.1	50.4
LAD [JWS$^+$17b]	**A**	81.0	55.1	51.1
JSLA [PTX$^+$16]	**A**	**82.8**	49.8	-
Visually-driven Semantic Augmentation, VdSA (ours)	**A**	*78.4*	***56.7***	*51.7*
DeVise [FCS$^+$13]	**W**	50.4	-	-
MTL [YH15]	**W**	55.3	-	-
ConSE [NMB$^+$13]	**W**	46.8	23.1	21.8
SynC [CCGS16]	**W**	56.7	21.5	28.5
LatEm [XAS$^+$16]	**W**	50.8	16.5	19.8
VAWE [QLSH17]	**W**	**61.2**	27.4	35.2
Visually-driven Semantic Augmentation, VdSA (ours)	**W**	*56.7*	***34.8***	***43.2***
Fu et al. [FXKG15]	**A + W**	66.0	-	-
SJE [ARW$^+$15]	**A + W**	73.9	51.7	-
Kodirov et al. [KXFG15a]	**A + W**	75.6	40.6	-
Visually-driven Semantic Augmentation, VdSA (ours)	**A + W**	***80.6***	***59.7***	***53.6***

Table 5.3: Benchmarking our proposed Visually-driven Semantic Augmentation (***VdSA***) with the state-of-the-art in ZSL. We report classification performance (measured in percentage %) obtained on the unseen classes, following the usual training/testing splits (see Table 5.1). The performance of our method is in italic, for each of the semantic embeddings (attributes **A**, word2vec **W** and the concatenation **A+W**) we highlight in bold the best performance. Globally, the highest classification scores in the Table are underlined.

we will take advantage of it to challenge the state-of-the-art in ZSL.

5.2.3 Comparison with the state-of-the-art in ZSL

To evaluate our proposed visually-driven semantic augmentation against the state of the art while making comparisons fair, we split methods on the basis of the adopted semantic embedding, either 1) attributes **A**, 2) DWE (word2vec) **W**, or 3) a combination of the two **A+W**.

1) Among the methods which use annotated attributes, we compare with the probabilistic graphical model of Direct Attribute Prediction (DAP) [LNH09]. We also consider DeVise [FCS$^+$13], SJE

[ARW$^+$15] and Embarrassing Simple ZSL (ESZSL) [RPT15a] that use bi-linear compatibility functions to map visual into semantic information. We include the Similarity Semantic Embedding (SSE) [ZS15], and the dictionary learning-based method of Kodirov et al. [KXFG15a], which both apply unsupervised domain adaptation methods to ZSL. We also consider the neural network based approach of [YH15] based on multi-task learning (MTL) and the metric-learning paradigm by Bucher et al. [BHJ16]. We also compare against the synthetic classifier (SynC) [CCGS16] which expresses images and semantic class embeddings as a mixture of seen class proportions. Finally, we evaluate our proposed augmentation scheme against the methods JLSE [ZS16a], LAD [JWS$^+$17b], JSLA [PTX$^+$16] which are all based on latent attributes.

2) In the case of DWEs (here word2vec [MSC$^+$13]), we additionally consider the convex combination of nearest neighbours predictors ConSE [NMB$^+$13], the latent embedding framework LatEm which takes advantage of a structured-output support vector machine [XAS$^+$16], and the VAWE approach[3] [QLSH17] that re-aligns the topology of the semantic embeddings by using the one of the visual embedding.

3) Finally, we also consider the geometrical method proposed by Fu et al. [FXKG15], which deploys ZSL on a manifold with geodesic distances.

Let us clarify that, we do not compare against transductive approaches [YG17, ZS16b] since those methods use unseen classes for training, while we do not. Moreover, we do not compare with methods such Zhang et al. [ZXG$^+$17] or Kodirov et al. [KXG17] since they take advantage of end-to-end learning to explicitly train a deep feature representation for ZSL, whereas, we use pre-computed visual features to codify the input images[4].

We report the quantitative results of our comparison with state-of-the-art methods in Table 5.3 and we discuss the results in the rest of this Section .

Discussion. When using **A**, we outperform ESZSL on AwA, CUB-200 and aP&Y by $+3\%$,$+7\%$ and $+27\%$, respectively, and SJE in CUB-200 by $+6\%$. With respect to DAP, we achieve a 21% improve-

[3] Since VAWE is not technically a full method but just a pruning techniques for DWEs, we reported the combination of WAVE with ConSE as published in qiao2017visually.

[4] In spite of that, it's worth saying that, when using **A + W**, our visual augmentation is still able to improve [ZXG$^+$17] by $+1.4\%$ on CUB-200.

ment in AwA, and 13% improvement in aP&Y. We improve over SSE and SynC by $+2\%$ in AwA and CUB-200 and $+5\%$ in aP&Y. We outperform MTL by more than 15% in AwA and by more than 26% in CUB-200.

In general, on aP&Y and CUB-200, our method is superior to all reported methods. The only exception is Bucher et al. [BHJ16]: on aP&Y, in fact metric learning [BHJ16] seems better than attribute augmentation - but, on either AwA and CUB-200, our approach is still superior. Finally, apart from AwA, our method is able to systematically improve other latent-attribute-based methods: we improve LAD by $+1.6\%$ on CUB-200 and by $+0.6\%$ on aP&Y, we outperform JSLA by $+6.9\%$ on CUB-200 and we are $+1.3\%$ and $+14.6\%$ better than JLSE on aP&Y and CUB-200, respectively. This empirically proves the claim that augmenting is preferable to modify semantic representations.

When comparing with methods that exploit **W**, our augmentation scheme scores again a solid performance: it is on par with respect to SynC on AwA (56.7%) and it outperforms DeVise, MTL, ConSE and LatEm by large margin. For instance, $+18.3\%$ on CUB-200 with respect to LatEm and $+21.4\%$ on aP&Y with respect to ConSE.

VAWE is worth a dedicated discussion. Despite in AwA our method is gapped by -4.5%, in either CUB-200 or aP&Y, our approach improves over the performance of VAWE by $+7.4\%$ and $+8.0\%$, respectively. Overall, this is an empirical evidence that (in most cases) visually-driven augmentation is preferable to visually-driven re-alignment.

Moreover, when combining **A + W**, our method improves Fu et al. [FXKG15], SJE and Kodirov et al. [KXFG15a] by $+5.0\%$ on AwA and by $+8.0\%$ on CUB-200 - in either Table 5.2 or 5.3, this is the most favorable setup for our augmentation.

In general, our method performs extremely well on CUB-200 which is a fine-grained dataset of birds (as opposed to AwA and aP&Y which tackle regular object recognition). Thus, our experimental findings seem to suggest that visual semantics augmentation is particularly effective in the case of fine-grained ZSL.

Chapter 6

Conclusion

This thesis presents approaches in computer vision tasks such as recognition of action and object categories in two starkly opposite data driven scenarios: traditional supervised learning with ample data and zero-shot supervised learning. First, we propose novel dictionary learning approaches that are suitable for traditional supervised learning. With ample labled data available for training, training data available, dictionary learning becomes detrimental in the success of a learning system. We come up with novel dictionary learning technique applicable to action recognition in still images and videos. More precisely, we take advantage of the available training samples to adaptively rank local features which are both robust and discriminative. Further, we cluster the local features at the entity and category levels to eliminate the effects of features corresponding to non-recurrent or background locations.

Then we propose mid-level representation for action recognition in videos which is robust in a traditional supervised learning setup as well as zero shot learning setup. We devise novel algorithms both for detecting and describing such mid-level elements which can detect local variabilities in human body-parts, in addition to capturing rich semantic information. Such mid-level elements, when used in conjunction with a novel and simple yet discriminative dictionary learning, exhibit superlative performance. More importantly, such features are transferable in the way that similar local (sub-)movements of a specific body part are shared by more than one action, which is helpful in obtaining some ’prior knowledge’ about previously unseen classes as in zero-shot learning.

Finally, we propose a novel optimization approach dedicated to zero-shot setup. We augment the semantic information conveyed by attributes/ distributed word embeddings with visual patterns which are distilled from data by means of soft labels. Consecutively, we analyse the benefits of casting such augmentation in the form of a learning pipeline where latent attributes do bottleneck compression to fuse multiple sources of semantic information.

This thesis essentially demonstrates the need for application of data driven approaches under varying scenarios to solve computer vision tasks. We develop novel approaches that can cope with these scenarios. We show efficiency of our proposed approaches in wide ranges of problems such as action and object recognition from images and videos. Although other areas in computer vision such as regression, semantic segmentation in images and videos can also also be explored.

Bibliography

[ADF12] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. Measuring the objectness of image windows. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(11):2189–2202, 2012.

[AHS15] Ziad Al-Halah and Rainer Stiefelhagen. How to transfer? zero-shot object recognition via hierarchical transfer of semantic attributes. In *Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on*, pages 837–843. IEEE, 2015.

[APHS13] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for attribute-based classification. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 819–826. IEEE, 2013.

[ARW$^+$15] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 2927–2936. IEEE, 2015.

[BHJ16] Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. Improving semantic embedding consistency by metric learning for zero-shot classiffication. In *European Conference on Computer Vision*, pages 730–746. Springer, 2016.

[Bis06a] Christopher M Bishop. Pattern recognition. *Machine Learning*, 128, 2006.

[Bis06b] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[BMB$^+$13] Andreas Baak, Meinard Müller, Gaurav Bharaj, Hans-Peter Seidel, and Christian Theobalt. A data-driven approach for real-time full body pose reconstruction from a depth camera. In *Consumer Depth Cameras for Computer Vision*, pages 71–98. Springer, 2013.

[CCGS16] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5327–5336, 2016.

[CLVZ11] Ken Chatfield, Victor S Lempitsky, Andrea Vedaldi, and Andrew Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *BMVC*, volume 2, page 8, 2011.

[CM02] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5):603–619, 2002.

[CSVZ14] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

[CWS$^+$15] Guangchun Cheng, Yiwen Wan, Abdullah N Saudagar, Kamesh Namuduri, and Bill P Buckles. Advances in human action recognition: a survey. *arXiv preprint arXiv:1501.05964*, 2015.

[DDS$^+$09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[DTS06] Navneet Dalal, Bill Triggs, and Cordelia Schmid. Human detection using oriented histograms of flow and appearance. In *European conference on computer vision*, pages 428–441. Springer, 2006.

[DWW15] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015.

[EK72] Jack Edmonds and Richard M Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM (JACM)*, 19(2):248–264, 1972.

[FCS$^+$13] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Tomas Mikolov, et al. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pages 2121–2129, 2013.

[FEHF09] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1778–1785. IEEE, 2009.

[FF$^+$03] Li Fe-Fei et al. A bayesian approach to unsupervised one-shot learning of object categories. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1134–1141. IEEE, 2003.

[FFFP06] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006.

[FGMR10] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2010.

[FGO$^+$17] Basura Fernando, Efstratios Gavves, José Oramas, Amir Ghodrati, and Tinne Tuytelaars. Rank pooling for action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(4):773–787, 2017.

[FXKG15] Zhenyong Fu, Tao Xiang, Elyor Kodirov, and Shaogang Gong. Zero-shot object recognition by semantic manifold distance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2635–2644, 2015.

[Gar85] Harold N Garbow. Scaling algorithms for network problems. *Journal of Computer and System Sciences*, 31(2):148–168, 1985.

[GL14] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

[HHP16] Samitha Herath, Mehrtash Harandi, and Fatih Porikli. Going deeper into action recognition: A survey. *arXiv preprint arXiv:1605.04988*, 2016.

[HKP06] Jonathan Harel, Christof Koch, and Pietro Perona. Graph-based visual saliency. In *Advances in neural information processing systems*, pages 545–552, 2006.

[HMS17] Amirhossein Habibian, Thomas Mensink, and Cees GM Snoek. Video2vec embeddings recognize events when examples are scarce. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (10):2089–2103, 2017.

[HVB00] Maria Halkidi, Michalis Vazirgiannis, and Yannis Batistakis. Quality scheme assessment in the clustering process. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 265–276. Springer, 2000.

[JLD11] Zhuolin Jiang, Zhe Lin, and Larry S Davis. Learning a discriminative dictionary for sparse coding via label consistent k-svd. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1697–1704. IEEE, 2011.

[JVGJ^{+}14] Mihir Jain, Jan Van Gemert, Hervé Jégou, Patrick Bouthemy, and Cees GM Snoek. Action localization with tubelets from motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 740–747, 2014.

[JVJZ13] Mayank Juneja, Andrea Vedaldi, CV Jawahar, and Andrew Zisserman. Blocks that shout: Distinctive parts for scene classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 923–930, 2013.

[JWS^{+}17a] Huajie Jiang, Ruiping Wang, Shiguang Shan, Yi Yang, and Xilin Chen. Learning discriminative latent attributes for zero-shot classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4223–4232, 2017.

[JWS^{+}17b] Huajie Jiang, Ruiping Wang, Shiguang Shan, Yi Yang, and Xilin Chen. Learning discriminative latent attributes for zero-shot classification. In *ICCV*, 2017.

[JXYY13] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):221–231, 2013.

[KJG+11] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563. IEEE, 2011.

[KSH12a] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[KSH12b] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*. 2012.

[KTS+14] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

[KXFG15a] E. Kodirov, T. Xiang, Z. Fu, and S. Gong. Unsupervised domain adaptation for zero-shot learning. In *ICCV*, 2015.

[KXFG15b] Elyor Kodirov, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Unsupervised domain adaptation for zero-shot learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2452–2460, 2015.

[KXG17] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. In *CVPR*, 2017.

[Lap05] Ivan Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005.

[LBRN06] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y Ng. Efficient sparse coding algorithms. In *Advances in neural information processing systems*, pages 801–808, 2006.

[LBSF$^+$15] Jimmy Lei Ba, Kevin Swersky, Sanja Fidler, et al. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4247–4255, 2015.

[LCHL07] Kang Hoon Lee, Myung Geol Choi, Qyoun Hong, and Jehee Lee. Group behavior from video: a data-driven approach to crowd simulation. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 109–118. Eurographics Association, 2007.

[LKS11] Jingen Liu, Benjamin Kuipers, and Silvio Savarese. Recognizing human actions by attributes. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3337–3344. IEEE, 2011.

[LMSR08] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[LNH09] CH. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009.

[LNH14a] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):453–465, 2014.

[LNH14b] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):453–465, 2014.

[Low04] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[LSFFX10] Li-Jia Li, Hao Su, Li Fei-Fei, and Eric P Xing. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *Advances in neural information processing systems*, pages 1378–1386, 2010.

[LSP06] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2169–2178. IEEE, 2006.

[LTCK16] Wei-Chao Lin, Chih-Fong Tsai, Zong-Yao Chen, and Shih-Wen Ke. Keypoint selection for efficient bag-of-words feature generation and effective image classification. *Information Sciences*, 329:33–51, 2016.

[MGS14] Thomas Mensink, Efstratios Gavves, and Cees G. M. Snoek. COSTA: co-occurrence statistics for zero-shot classification. In *CVPR*, 2014.

[MH08] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

[MPS$^+$09] Julien Mairal, Jean Ponce, Guillermo Sapiro, Andrew Zisserman, and Francis R Bach. Supervised dictionary learning. In *Advances in neural information processing systems*, pages 1033–1040, 2009.

[MSC$^+$13] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*. 2013.

[MSN11] Dhruv Mahajan, Sundararajan Sellamanickam, and Vinod Nair. A joint learning framework for attribute models and object descriptions. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1227–1234. IEEE, 2011.

[NJT06] Eric Nowak, Frédéric Jurie, and Bill Triggs. Sampling strategies for bag-of-features image classification. In *European conference on computer vision*, pages 490–503. Springer, 2006.

[NMB$^+$13] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013.

[OF97] Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

[OT06] Aude Oliva and Antonio Torralba. Building the gist of a scene: The role of global image features in recognition. *Progress in brain research*, 155:23–36, 2006.

[PP07] Massimiliano Pavan and Marcello Pelillo. Dominant sets and pairwise clustering. *IEEE transactions on pattern analysis and machine intelligence*, 29(1):167–172, 2007.

[PS13] Guillem Palou and Philippe Salembier. Hierarchical video representation with trajectory binary partition tree. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2099–2106, 2013.

[PSM10] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *European conference on computer vision*, pages 143–156. Springer, 2010.

[PSM14] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[PTX$^+$16] Peixi Peng, Yonghong Tian, Tao Xiang, Yaowei Wang, and Tiejun Huang. Joint learning of semantic and latent attributes. In *European Conference on Computer Vision*, pages 336–353. Springer, 2016.

[QLSH17] Ruizhi Qiao, Lingqiao Liu, Chunhua Shen, and Anton van den Hengel. Visually aligned word embeddings for improving zero-shot learning. *arXiv preprint arXiv:1707.05427*, 2017.

[RBM] Abhinaba Roy, Biplab Banerjee, and Vittorio Murino. Discriminative latent visual space for zero-shot object classification.

[RBM17a] Abhinaba Roy, Biplab Banerjee, and Vittorio Murino. Discriminative dictionary design for action classification in still images. In *International Conference on Image Analysis and Processing*, pages 160–170. Springer, 2017.

[RBM17b] Abhinaba Roy, Biplab Banerjee, and Vittorio Murino. A novel dictionary learning based multiple instance learning approach to action recognition from videos. In *Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM,*, pages 519–526. INSTICC, ScitePress, 2017.

[RBM17c] Abhinaba Roy, Biplab Banerjee, and Vittorio Murino. A novel dictionary learning based multiple instance learning approach to action recognition from videos. In *Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods, ICPRAM 2017, Porto, Portugal, February 24-26, 2017.*, pages 519–526, 2017.

[RBM18] Abhinaba Roy, Biplab Banerjee, and Vittorio Murino. Discriminative body part interaction mining for mid-level action representation and classification. *Journal of Visual Communication and Image Representation*, 55:829–840, 2018.

[RCM] Abhinaba Roy, Jacopo Cavazza, and Vittorio Murino. Visually-driven semantic augmentation for zero-shot learning.

[RHGS15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[RKS12] Michalis Raptis, Iasonas Kokkinos, and Stefano Soatto. Discovering discriminative action parts from mid-level video representations. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1242–1249. IEEE, 2012.

[RPE$^+$05] Liu Ren, Alton Patrick, Alexei A Efros, Jessica K Hodgins, and James M Rehg. A data-driven approach to quantifying natural human motion. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 1090–1097. ACM, 2005.

[RPT15a] Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *International Conference on Machine Learning*, pages 2152–2161, 2015.

[RPT15b] Bernardino Romera-Paredes and PHS Torr. An embarrassingly simple approach to zero-shot learning. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 2152–2161, 2015.

[RRM15] Michael S Ryoo, Brandon Rothrock, and Larry Matthies. Pooled motion features for first-person videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 896–904, 2015.

[RS10] Michalis Raptis and Stefano Soatto. Tracklet descriptors for action modeling and video analysis. In *European conference on computer vision*, pages 577–590. Springer, 2010.

[RS13] Kishore K Reddy and Mubarak Shah. Recognizing 50 human action categories of web videos. *Machine Vision and Applications*, 24(5):971–981, 2013.

[RSS11] Marcus Rohrbach, Michael Stark, and Bernt Schiele. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1641–1648. IEEE, 2011.

[Rud76] W. Rudin. *Principles of Mathematical Analysis*. International series in pure and applied mathematics. McGraw-Hill, 1976.

[SB16] Seyed Mohsen Shojaee and Mahdieh Soleymani Baghshah. Semi-supervised zero-shot learning by a clustering-based approach. *arXiv preprint arXiv:1605.09016*, 2016.

[SC12] Sreemanananth Sadanand and Jason J Corso. Action bank: A high-level representation of activity in video. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1234–1241. IEEE, 2012.

[Sch02] Alexander Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer Science & Business Media, 2002.

[SJ15] Ronan Sicre and Frédéric Jurie. Discriminative part model for visual recognition. *Computer Vision and Image Understanding*, 141:28–37, 2015.

[SJS13] Gaurav Sharma, Frédéric Jurie, and Cordelia Schmid. Expanded parts model for human attribute and action recognition in still images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–659, 2013.

[SLC04] Christian Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local svm approach. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 32–36. IEEE, 2004.

[SMS15] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised learning of video representations using lstms. *CoRR, abs/1502.04681*, 2, 2015.

[Sug06] Masashi Sugiyama. Local fisher discriminant analysis for supervised dimensionality reduction. In *Proceedings of the 23rd international conference on Machine learning*, pages 905–912. ACM, 2006.

[SZ14a] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014.

[SZ14b] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[SZ14c] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[SZS12] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[TBF$^+$15] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 4489–4497. IEEE, 2015.

[TSS07] Christian Thiel, Stefan Scherer, and Friedhelm Schwenker. Fuzzy-input fuzzy-output one-against-all support vector machines. In *International Conference on Knowledge-*

Based and Intelligent Information and Engineering Systems, pages 156–165. Springer, 2007.

[TT16] Du Tran and Lorenzo Torresani. Exmoves: Mid-level features for efficient action recognition and video analysis. *International Journal of Computer Vision*, pages 1–15, 2016.

[vGJGS15] J van Gemert, Mihir Jain, Ella Gati, and C Snoek. Apt: Action localization proposals from dense trajectories. In *BMVC*, volume 2, page 4, 2015.

[VLS16] Gül Varol, Ivan Laptev, and Cordelia Schmid. Long-term temporal convolutions for action recognition. *arXiv preprint arXiv:1604.04494*, 2016.

[WBW$^+$11] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[WJ13] Xiaoyang Wang and Qiang Ji. A unified probabilistic approach modeling relationships between attributes and objects. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2120–2127. IEEE, 2013.

[WKSL11] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3169–3176. IEEE, 2011.

[WKSL13] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*, 103(1):60–79, 2013.

[WLSS16] Peng Wang, Lingqiao Liu, Chunhua Shen, and Heng Tao Shen. Order-aware convolutional pooling for video based action recognition. *arXiv preprint arXiv:1602.00224*, 2016.

[WM10] Yang Wang and Greg Mori. A discriminative latent model of object classes and attributes. In *European Conference on Computer Vision*, pages 155–168. Springer, 2010.

[WQT13] LiMin Wang, Yu Qiao, and Xiaoou Tang. Mining motion atoms and phrases for complex action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2680–2687, 2013.

[WQT15] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4305–4314, 2015.

[WS13] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3551–3558, 2013.

[WTLF12] Yang Wang, Duan Tran, Zicheng Liao, and David Forsyth. Discriminative hierarchical part-based models for human parsing and action recognition. *Journal of Machine Learning Research*, 13(Oct):3075–3102, 2012.

[WTVG08] Geert Willems, Tinne Tuytelaars, and Luc Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *European conference on computer vision*, pages 650–663. Springer, 2008.

[WUK$^+$09] Heng Wang, Muhammad Muneeb Ullah, Alexander Klaser, Ivan Laptev, and Cordelia Schmid. Evaluation of local spatio-temporal features for action recognition. In *BMVC 2009-British Machine Vision Conference*, pages 124–1. BMVA Press, 2009.

[WXDL11] Xinxiao Wu, Dong Xu, Lixin Duan, and Jiebo Luo. Action recognition using context and appearance distribution features. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 489–496. IEEE, 2011.

[WYY$^+$10] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3360–3367. IEEE, 2010.

[XAS$^+$16] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 69–77, 2016.

[XHG15] Xun Xu, Timothy Hospedales, and Shaogang Gong. Semantic embedding space for zero-shot action recognition. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 63–67. IEEE, 2015.

[XHG16] Xun Xu, Timothy M Hospedales, and Shaogang Gong. Multi-task zero-shot action recognition with prioritised data augmentation. In *European Conference on Computer Vision*, pages 343–359. Springer, 2016.

[XSA17] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning - the good, the bad and the ugly. *CVPR*, 2017.

[YG17] Meng Ye and Yuhong Guo. Zero-shot classification with discriminative semantic representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7140–7148, 2017.

[YH15] Yongxin Yang and Timothy M Hospedales. A unified perspective on multi-domain and multi-task learning. 2015.

[YJK$^+$11] Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lai Lin, Leonidas Guibas, and Li Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *2011 International Conference on Computer Vision*, pages 1331–1338. IEEE, 2011.

[YL16] Yang Yi and Maoqing Lin. Human action recognition with graph-based multiple-instance learning. *Pattern Recognition*, 53:148–162, 2016.

[YWM10] Weilong Yang, Yang Wang, and Greg Mori. Recognizing human actions from still images with latent poses. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2030–2037. IEEE, 2010.

[YY15] Gang Yu and Junsong Yuan. Fast action proposals for human action detection and search. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1302–1311, 2015.

[ZG02] Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. 2002.

[ZLP$^+$16] Lei Zhang, Changxi Li, Peipei Peng, Xuezhi Xiang, and Jingkuan Song. Towards optimal vlad for human action recognition from still images. *Image and Vision Computing*, 2016.

[ZNH$^+$15] Yang Zhou, Bingbing Ni, Richang Hong, Meng Wang, and Qi Tian. Interaction part mining: A mid-level approach for fine-grained action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3323–3331, 2015.

[ZS15] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via semantic similarity embedding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4166–4174, 2015.

[ZS16a] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via joint latent similarity embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6034–6042, 2016.

[ZS16b] Ziming Zhang and Venkatesh Saligrama. Zero-shot recognition via structured prediction. In *European conference on computer vision*, pages 533–548. Springer, 2016.

[ZXG$^+$17] Li Zhang, Tao Xiang, Shaogang Gong, et al. Learning a deep embedding model for zero-shot learning. 2017.