Mid-level Representations for Action Recognition and Zero-shot Learning

Ruizhi Qiao

A thesis submitted for the degree of
DOCTOR OF PHILOSOPHY
School of Computer Science
The University of Adelaide

August 2017

To Yazhuo.

Declaration

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I give consent to this copy of my thesis, when deposited in the University Library, being made available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

Publications

This thesis is based on the content of the following journal and conference papers:

- **Ruizhi Qiao**, Lingqiao Liu, Chunhua Shen, Anton van den Hengel, "Learning discriminative trajectorylet detector sets for accurate skeleton-based action recognition", *Pattern Recognition* (PR), 2017.
- **Ruizhi Qiao**, Lingqiao Liu, Chunhua Shen, Anton van den Hengel, "Less is More: Zero-shot Learning from Online Textual Documents with Noise Suppression Mechanism" *Computer Vision and Pattern Recognition* (CVPR), 2016.
- **Ruizhi Qiao**, Lingqiao Liu, Chunhua Shen, Anton van den Hengel, "Visually Aligned Word Embeddings for Zero-shot Learning", *British Machine Vision Conference* (BMVC), 2017.

In addition, I have co-authored the below papers:

- Fayao Liu, Guosheng Lin, **Ruizhi Qiao**, Chunhua Shen, "Structured Learning of Tree Potentials in CRF for Image Segmentation," *IEEE Transactions on Neural Networks and Learning Systems* (TNNLS), 2017.
- Fayao Liu, **Ruizhi Qiao**, Chunhua Shen, Lei Luo, "Designing ensemble learning algorithms using kernel methods," *International Journal of Machine Intelligence and Sensory Signal Processing*, 2017.

Acknowledgments

First of all, I would like to thank my supervisors, Prof. Chunhua Shen and Prof. Anton van den Hengel, for their insightful guidance on my research. During my PhD study, they have given me lots of suggestions on the research topics that I am interested in. I also thank them for their helpful support whenever I was confronted with difficulties. It is with their supervision and support that I can solve one and another research problems and finish this thesis.

I would like to express a special gratitude to Dr. Lingqiao Liu. As my co-supervisor, he constantly helps me with polishing new ideas and implementing them. Without his collaboration and feedback, many insightful ideas of this thesis would not have been possible. I would also like to thank my former co-supervisor Dr. Peng Wang for his guidance over the first year of my PhD study.

I thank my supportive friends and colleagues at the University of Adelaide: Yuanzhouhan Cao, Yao Li, Hui Li, Teng Li, Bohan Zhuang, who not only shared their meaningful discussions on their research topics, but also spent a great time with me.

My thanks also go to the China Scholarship Council, which grants me financial support to complete my PhD program.

Last but not the least, I would like to thank my parents and my girlfriend for their love and support throughout the tough times during my PhD study.

Abstract

Compared with low-level features, mid-level representations of visual objects contain more discriminative and interpretable information and are beneficial for improving performance of classification and sharing learned information across object categories. These benefits draw tremendous attention of the computer vision communities and lots of breakthroughs have been made for various computer vision tasks with mid-level representations. In this thesis, we focus on the following problems regarding mid-level representations: 1) How to extract discriminative mid-level representations from local features? 2) How to suppress noisy components from mid-level representations? 3) And how to address the issue of visual-semantic discrepancy in mid-level representations? We deal with the first problem in the task of action recognition and the other two problems in the task of zero-shot learning.

For the first problem, we devise a representation suitable for characterising human actions on the basis of a sequence of pose estimates generated by an RGB-D sensor. We show that discriminate sequence of poses typically occur over a short time window, and thus we propose a simple-but-effective local descriptor called a trajectorylet to capture the static and kinematic information within this interval. We also show that state of the art recognition results can be achieved by encoding each trajectorylet using a discriminative trajectorylet detector set which is selected from a large number of candidate detectors trained through exemplar-SVMs. The mid-level representation is obtained by pooling trajectorylet encodings.

For the second problem, we follow the attractive research topic zero-shot learning and focus on classifying a visual concept merely from its associated online textual source, such as a Wikipedia article. We go further to consider one important factor: the textual representation as a mid-level representation is usually too noisy for the zero-shot learning tasks. We design a simple yet effective zero-shot learning method that is capable of suppressing noise in the text. Specifically, we propose an $l_{2,1}$-norm based objective function which can simultaneously suppress the noisy signal in the text and learn a function to match the text document and visual features. We also develop an optimization algorithm to efficiently solve the resulting problem.

For the third problem, we observe that distributed word embeddings, which become a popular mid-level representation for zero-shot learning due to their easy accessibility, are designed to reflect semantic similarity rather than visual similarity and thus using them in zero-shot learning often leads to inferior performance.

To overcome this visual-semantic discrepancy, we here re-align the distributed word embedding with visual information by learning a neural network to map it into a new representation called the visually aligned word embedding (VAWE). We further design an objective function to encourage the neighbourhood structure of VAWEs to mirror that in the visual domain. This strategy gives more freedom in learning the mapping function and allows the learned mapping function to generalize to zero-shot learning methods and different visual features.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Classification or categorization lays the foundation of computer vision and pattern recognition. The task is to classify an object into one or more categories by drawing the connection from the features of the object to the definitions of categories. One of the key issues in classification is the feature representation of the objects. Given appropriately represented features, classification tasks can be performed on any kind of objects: images (pixel values, CNN activations, etc.), videos (pixel values, CNN activations, etc., over time), actions (optical flow, skeleton joints coordinates, etc., over time), speeches (auditory signals), etc.. Despite significant recognition results these low-level representations produced over the past 20 years, they often lack meaningful and interpretable descriptions for human beings to understand the objects. Being the global and explicit descriptors of objects, low-level representations make strong assumptions on the training samples, e.g. well-aligned faces or even-paced actions. However, in real-world applications those assumptions are mostly violated and this limits the generalization ability of low-level representations.

Attributes (Ferrari and Zisserman [2008]; Farhadi et al. [2009]), on the other hand, are mid-level semantic descriptions of objects that are close to human interpretation. Compared conventional category-level classification, the attributes are adjectives that describe the nouns (category). In some cases such as attribute discovery (Rastegari et al. [2012]; Yu et al. [2013]), however, attributes do not need be explicitly semantic, but rather be some discriminative mid-level representations. Typical attributes can be either binary (presence or absence of a property) or continuous (strength of a

property) vectors, but other general structures (text descriptions, wordnet hierarchy, learned discriminative patterns) can still serve as the purpose of attributes. Because attributes co-occur across different categories, they are beneficial for improving performance of classification and for transferring learned information between object categories. Given a sufficiently rich dataset of learned adjectives, new categories of objects can be recognized simply from a verbal description consisting of a list of the attributes with just a few or even no training examples.

With their two major benifits, attributes are usually applied in two fields: 1) improving classfication results as more discriminative mid-level representations (Torresani et al. [2010]; Rastegari et al. [2012]; Yu et al. [2013]); 2) enabling zero/few-shot learning as informative media that are transferable between seen and unseen categories (Lampert et al. [2009]; Farhadi et al. [2009]). Both approaches are made possible by the connection from low-level representations to attributes and the connection from attributes to categories. Although the recent use of attributes has led to exciting advances in both areas (Zhang and Saligrama [2015]; Escorcia et al. [2015]), the annotation cost of attribute constrains its further applications large-scale environments since extensive human labours are required to define the attribute vector for each category.

To bypass the limitation of human-defined attributes, most recent works have explored alternative mid-level representations which are learned from data other than hand-crafted. These include automatically mined mid-level representations from an auxiliary source (semantic attribute) or the training data itself (discriminative patterns) and learning category-level representations (word embeddings, document, etc.) from auxiliary sources.

1.1 Problem Formulation

In this thesis we focus on two topics heavily related to the alternatives of hand-crafted attributes in two applications: action recognition and zero-shot learning.

1.1.1 Action Recognition with Discovered Discriminative Mid-level Representation

The advent of low-cost RGB-D sensors, and their ability to rapidly capture sequences of human pose estimates, has promoted a large amount of research interest in skeleton-based human action recognition. Intuitively, a temporal sequence of 3D skeleton joint locations captures sufficient information to distinguish between actions, but recording such skeleton sequences was previously very expensive with the traditional motion capture technology Moeslund et al. [2006]. Recently, the advent low-cost of RGB-D cameras such as Microsoft Kinect Han et al. [2013] and their ability to rapidly capture sequences of human pose estimates, has promoted a large amount of research interest in skeleton-based human action recognition Shotton et al. [2011]. This advance has promoted the development of a range of skeleton-based action recognition approaches (Vemulapalli et al. [2014]; Gowayyed et al. [2013]; Wu and Shao [2014b]). Devising a representation suitable for characterising human actions on the basis of a sequence of pose estimates generated by an RGBD sensor remains a research challenge.

In contrast to the previous approaches which either represent an action with the whole sequence or extract local features at the frame level, we argue that the discriminative information regarding an action is better captured by a short interval of trajectories. We here provide two insights into this challenge. First, we show that discriminative sequence of poses typically occur over a short time window, and thus we propose a simple-but-effective local descriptor called a trajectorylet to capture the static and kinematic information within this interval. Second, we design the mid-level representation by encoding each trajectorylet using a discriminative trajectorylet detector set which is selected from a large number of candidate detectors trained through exemplar-SVMs. The action-level representation is obtained by pooling trajectorylet encodings.

1.1.2 Zero-shot Learning with alternatives of Attributes

Unlike traditional object classification tasks in which the training and test categories are identical, zero-shot learning aims to recognize objects from classes not seen at the training stage. It is recognized as an effective way for large scale visual classification since it alleviates the burden of collecting sufficient training data for every possible class. The key component ensuring the success of zero-shot learning is to find an intermediate semantic representation to bridge the gap between seen and unseen classes. In a nutshell, with this semantic representation we can first learn its connection with image features and then transfer this connection to unseen classes. So once the semantic representation of an unseen class is given, one can easily classify the image through the learned connection.

Attributes, which essentially represent the discriminative properties shared among both seen and unseen categories, have become the most popular semantic representation in zero-shot learning. Although the recent use of attributes has led to exciting advances in zero-shot learning, the creation of attributes still relies on much human labour. This is inevitably discouraging since the motivation for zero-shot learning is to free large-scale recognition tasks from cumbersome annotation requirements.

1.1.2.1 Zero-shot Learning with On-line Documents

To remedy this drawback and move towards the goal of fully automatic zero-shot learning, one possible choice is to directly use online textual documents, e.g., those found in Wikipedia, to build such a representation (Elhoseiny et al. [2013]; Ba et al. [2015]). This is promising because online text documents can be easily obtained and contain rich information about the object. To conduct zero-shot learning with textual documents, existing works (Akata et al. [2015]; Fu et al. [2015]) develop various ways to measure the similarity between text and visual features. Our work is also based on this idea. We take a step further, however, to consider one additional important factor: the document representation is much more noisy than the human specified

semantic representation and negligence of this fact would inevitably lead to inferior performance. For example, when the bag-of-words model is adopted as the document representation, the occurrence of every word in a document will trigger a signal in one dimension of the document representation. However, it is clear that most words in a document are not directly relevant for identifying the object category. Thus it is necessary to design a noise suppression mechanism to down weight the importance of those less relevant words for zero-shot learning.

1.1.2.2 Zero-shot Learning with Distributed Word Embeddings

Recently, several works have explored to use distributed word embeddings (DWE) (Mikolov et al. [2013]; Pennington et al. [2014]) as the alternative to attributes in zero-shot learning (Frome et al. [2013]; Norouzi et al. [2014]). In contrast to human annotated attributes, DWEs are learned from a large-scale text corpus in an unsupervised fashion, which requires little or no human labour to collect.

Different to existing work, the method proposed in this thesis directly learns a neural network to map the semantic embedding to a space in which the mapped semantic embeddings preserves a similar neighbourhood structure as their visual counterparts. In other words, we do not require the mapped semantic embeddings to be comparable to visual features but only impose constraints on their structure. This gives more freedom in learning the mapping function, and this could potentially enhance its generalizability. Moreover, since our approach is not tied to a particular zero-shot learning method, the learned mapping can be applied to any zero-shot learning algorithm.

1.2 Main Contribution

We propose several novel algorithms that are applied to the different tasks (action recognition and zero-shot learning) as introduced in the previous section and conduct in-depth analyses on them. These works serve as the main contributions of this

thesis and they are listed as follows:

- We design a novel local descriptor called a trajectorylet to capture the static and dynamic pose information within the short interval of joint trajectories. A novel framework is proposed to generate robust and discriminative representation for action instances from a set of learned template trajectorylet detectors. The action-level representation is obtained by pooling trajectorylet encodings. Evaluating on standard datasets acquired from the Kinect sensor, it is demonstrated that our method obtains superior results over existing approaches under various experimental setups.

- We consider one important factor for zero-shot learning with online documents: the textual representation is usually too noisy for the zero-shot learning application. This observation motivates us to design a simple yet effective zero-shot learning method that is capable of suppressing noise in the text. Specifically, we propose an $l_{2,1}$-norm based objective function which can simultaneously suppress the noisy signal in the text and learn a function to match the text document and visual features. We also develop an optimization algorithm to efficiently solve the resulting problem. By conducting experiments on two large datasets, we demonstrate that the proposed method significantly outperforms those competing methods which rely on online information sources but with no explicit noise suppression. Furthermore, we make an in-depth analysis of the proposed method and provide insight as to what kind of information in documents is useful for zero-shot learning.

- Compared with human defined attributes, distributed word embeddings (DWEs) are more scalable and easier to obtain. However, they are designed to reflect semantic similarity rather than visual similarity and thus using them in ZSL often leads to inferior performance. To overcome this visual-semantic discrepancy, this work proposes an objective function to re-align the distributed word embeddings with visual information by learning a neural network to map it

into a new representation called visually aligned word embedding (VAWE). Thus the neighbourhood structure of VAWEs becomes similar to that in the visual domain. Note that in this work we do not design a ZSL method that projects the visual features and semantic embeddings onto a shared space but just impose a requirement on the structure of the mapped word embeddings. This strategy allows the learned VAWE to generalize to various ZSL methods and visual features. As evaluated via four state-of-the-art ZSL methods on four benchmark datasets, the VAWE exhibit consistent performance improvement.

1.3 Thesis Outline

The strucutre of this thesis is outlined as follows.

In chapter 1, we give a brief overview and background on object classification and attributes. We introduce two main topics of this thesis, and summarise its main contributions.

In Chapter 2, we give a literature review of the background of our study. This involves action recognition methods, attribute-based methods for object classification, zero-shot learning, and alternatives of hand-crafted attributes for zero-shot learning.

In Chatper 3, We propose an action recognition approach with discriminative mid-level representations. A novel local descriptor called a trajectorylet is designed to capture the static and dynamic pose information. And a novel framework is proposed to generate robust and discriminative representation for action instances from a set of learned template trajectorylet detectors.

In Chapter 4, we propose a zero-shot learning method which particularly caters for the need for noise suppression of text documents as a substitution for attributes. We also develop an optimization algorithm to efficiently solve the resulting problem. We then conduct an in-depth analysis of the proposed method which provides an insight as to what kinds of information within a document are useful for zero-shot learning.

In Chapter 5, we provide a meta-method that aids the performance of word vector based zero-shot learning methods. We first empirically demonstrate that the inferior ZSL performance of DWE is caused by the discrepancy between visual features and semantic representations. We align the distributed word embedding with visual information by learning a neural network to map it into a new representation called the visually aligned word embedding (VAWE). We further design an objective function to encourage the neighbourhood structure of VAWEs to mirror that in the visual domain.

In Chapter 6, we conclude this thesis and discuss the potential research direction in the future.

Chapter 2

Literature Review

In this chapter, we will firstly review the conventional approaches to action recognition. Then, we will introduce the attributes and their applications to object classification and zero-shot learning. Finally, we will look beyond attributes and review some alternatives of attributes in zero-shot learning.

2.1 Approaches for Action Recognition

2.1.1 Overview

Traditionally, the object to be identified in action recognition tasks resides in a set of video clips. As a result, in video-based action recognition, multiple tracking points of an action performer needs to be estimated first before the feature representation and classification process. With the advance of technology, motion capture sensors and RGB-D cameras provide the explicit 3D coordinates of the space-time evolution of marked or estimated skeletal joints of the action performer. This review will cover the input sources of action recognition as well as methods for building action representation.

2.1.2 Input Sources

2.1.2.1 Video clips

Video clips record raw information of pixel intensity variations in space-time. However, mere pixel intensity is not able to provide enough information to understand

and identify the action performed in the video. Therefore low-level features need to be extracted firstly before further steps and here we briefly describe three popular choices.

- **Optical flow** Beauchemin and Barron [1995]: Optical flow defined as the apparent motion of individual pixels on the image plane. It often serves as a good approximation of the true physical motion of objects that is projected onto the image plane. The methods to determine optical flow try to calculate the motion between consecutive image frames at each pixel position. The accuracy of estimated optical flow is affected by noise and illumination changes.

- **Trajectories of interest points** Sethi and Jain [1987]: Trajectories of moving objects have popularly been used as features to infer the activity of the object. The trajectory in 2D image plane is not very useful as it is sensitive to translations, rotations and scale changes. Alternative representations such as trajectory velocities, trajectory speeds, spatio-temporal curvature, relative-motion etc. have been proposed that are invariant to some of these variabilities.

- **Silhouettes** Bobick and Davis [2001]: The shape of the human silhouette plays a very important role in recognizing human actions, and it can extracted from background modelling techniques Elgammal et al. [2000]. Visual features based on global, boundary and skeletal descriptors have been proposed to model the silhouette actions.

2.1.2.2 Skeleton Squence

As video-based action recognition is conducted on a 2D image plane, it suffers from loss of depth information. Skeleton-based action recognition, on the other hand, benefits from the direct modelling of moving 3D points of performers. Here we briefly introduce two major sources to acquire such moving skeleton sequences.

- **Motion capture** Bruderlin and Williams [1995]: Motion capture (or mocap) data create accurate and realistic motion estimation in the form of 3D points of skeleton joints in space-time. The motions are performed by live actors, captured by a digital mocap system, and finally mapped to an animated skeleton. There are various ways to generate motion capture data using, e. g., mechanical, magnetic, or optical systems. Although the mocap data have the advantage of high precision and low latency, the mocap devices are expensive and demand specific environment to record the actions. This limits their application in wider range.

- **RGB-D camera** Li et al. [2010]: RGB-D cameras such as Microsoft Kinect capture both RGB (red, green and blue) video and depth (D) information. This allows to predict 3D positions of body joints Shotton et al. [2011] from depth images. In Shotton et al. [2011], a single input depth image is segmented into a dense probabilistic body part labelling, with the parts defined to be spatially localized near skeletal joints of interest. The labelled body parts are then projected into world space as skeletal joints. Compared with mocap data, the acquisition of 3D skeleton data is much easier, faster, and more practical, but less accurate, since the estimated 3D locations of skeletal joints can be degraded by occlusion and illumination. Despite the disadvantages, RGB-D data are widely used in skeleton-based recognition because they are more suitable for real-world applications.

2.1.3 Action Representation

In both video-based and skeleton-based action recognition, the key challenge is how to construct the action representation from a sequence of frames in space-time. The representations are usually depicted with holistic or local spatial-temporal features. The former uses low-level features directly to depict the whole action, while the latter mines informative local features to form mid-level representations of action.

2.1.3.1 Holistic representation

The most straightforward way in action representation is to model the action holistically, either by extracting statistics from the sequence or modelling its generative process. In Gowayyed et al. [2013], a histogram records the displacements of joint orientations over the whole trajectory. In Ohn-bar and Trivedi [2013], the action is modelled with the pairwise affinities trajectories of joint angles. In Xia et al. [2012], the action sequence is modelled by the Hidden Markov Model with quantized histogram of spherical coordinates of joint locations as frame-level feature. In Wu and Shao [2014a] and Wu and Shao [2014b], deep neural networks such as deep belief networks and 3D convolutional neural networks are adopted for spatio-temporal feature extraction from skeletal and depth data, and then a Hidden Markov Model is used to infer the action class with the learned representation. In Vemulapalli et al. [2014], 3D geometric relationships between various body parts are modelled with a Lie group to represent the whole action.

2.1.3.2 Mid-level Representation from local parts

Besides direct modelling the holistic representation, a natural observation is that only a small portion of the action is distinctive pattern for the classification, either spatially or temporally. Mid-level representations can be mined from a collection of discriminative local spatial-temporal parts.

Regarding spatial dimensions, researchers find that not all body parts can be activated from static pose during the action, and a compact representation formed by the activated body parts only can be constructed. In Ofli et al. [2012] a subset of most informative joints is selected according to criteria such as mean or variance of joint angles. In Wang et al. [2012], joints are grouped into actionlets, and the most discriminative collection of them are mined via a multiple kernel approach. In Chaudhry et al. [2013], a subset of joints during a short-time interval is extracted according to the spatio-temporal hierarchy of moving skeleton, and a linear combination of them

are learned via a discriminative metric learning approach. In Du et al. [2015], the skeleton is divided sub-parts and they are fed into a deep RNN architecture, which learns the action representation at the fully connected layer. In Moussa et al. [2017], a technique is proposed to distinguish between different actions using features learned from global variation in the visual appearance of the subject body.

Temporally, as most of the frames in an action sequence describe non-distinctive static poses, features at a small number of discriminative temporal locations are informative enough to represent the action. In video-based action recognition, a number of key frame selection approaches have been proposed. In Zhao and Elgammal [2008], key frames are selected by ranking the conditional entropy of the codewords assigned to the frames. In Raptis and Sigal [2013], key frames are encoded as latent variables and computed by dynamic programming for each action instance. In the recent work on skeleton-based action recognition, distinctive canonical poses Ellis et al. [2013] are learned via logistic regression, and discriminative frames Zanfir et al. [2013] determined by their approximated confidence on specific action classes. In Yang and Tian [2012], distinctiveness of each frame is calculated by a measurement of accumulated motion energy.

2.2 Attributes

2.2.1 Definition

Attribute is a special type of mid-level representations. Given a set of attributes $\mathcal{A} = \{a_1, \cdots, a_m\}$, where each attribute depicts a single semantic or discriminative associated to the object, and image features $\mathcal{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\} \in \mathcal{R}^d$, the assignment of attributes to an image is a multi-label classification problem $f : \mathcal{R}^d \rightarrow \mathcal{Y}$, where $\mathcal{Y} = \{0, 1\}^m$ indicates the presence of the attributes in the image. As a result, the traditional definition of attributes stays in the form of a m-dimensional binary vector $\mathbf{A} = (a_1, \cdots, a_m) \in \{0, 1\}$ and attribute classifiers are learned to find attributes of

images. Besides binary attribute, continuous attributes $\mathbf{A} = (a_1, \cdots, a_m) \in \mathcal{R}^m$ marks the strength of association for each attribute.

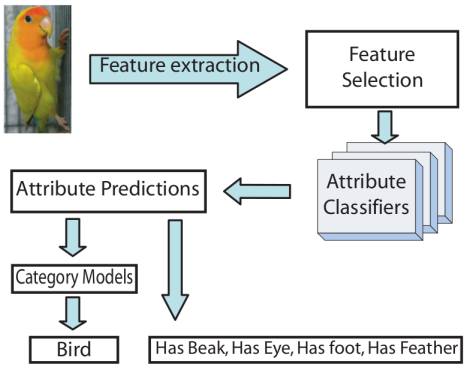

Figure 2.1: A typical framework of attribute learning in Farhadi et al. [2009]: the predicted attributes serve as mid-level representation for describing known objects and identifying unkown objects

2.2.2 Attribute learning

Because of their rich representation ability, attributes provide deeper understandings of images than other mid-level representations. The earliest work of predicting attributes for images is by Ferrari and Zisserman [2008]. They use a generative model to learn simple attributes defined by segments, such as “stripe”, and “ dots". But these patterns are difficult to learn in natural images, as they assume near-perfect segmentation of the pattern.

Torresani et al. [2010] are among the first works to define attributes as descriptors (classmes, in their term) for classification tasks. They learn attribute classifiers with LP-β kernel combiner, each kernel made of a feature type. The multi-label real-valued classifier output vector is directly used as descriptor.

Farhadi et al. [2009] are among the first to propose to learn visual attributes to identify familiar objects, and to describe unfamiliar objects when new images are provided. They predefine the semantic attributes and generate discriminative attribute by comparison of random splits of images in different categories. To remove

irrelevant low-level features, they design a heuristic to select the relevant feature for specific attribute classifiers by using L1-regularized logistic regression trained for each attribute within each class, and then pool selected features over all classes.

In the early work of zero-shot learning for novel category classification, Lampert et al. [2009] introduce another way of learning attribute: instead of learning attribute labels from the low level features, they first train the multi-class classifiers, and then estimate the posterior distribution of the training class labels over attributes. This induces a distribution over the labels of unseen classes by means of the estimated class-attribute relationship.

Instead of training a classifier for each attribute independently, Rastegari et al. [2013] argue that training conjunctions of related attributes as whole is more efficient and effective. Their intuition is when images that contain both attributes occupy a tight region in the feature space and have enough margin to images that have only one of the attributes, the merging of the two attributes are more learnable. To implement this idea, they define a gain function for pairs of attributes to investigate their joint learnability. And they design an algorithm to recursively combine attribute pairs with positive gains. To measure the margin and tightness, they map images to binary spaces and calculate the Humming distances.

Vedaldi et al. [2014] investigate attribute learning in fine-grained part of objects by introducing Objects in Detail (OID), intended as describing an object and its parts with a rich set of semantic attributes. They use manually mined attributes, and focus on a single object category to collect significantly deeper annotations. The attribute classifiers are learned by restricting visual information to specific parts of the objects detected by DPM or BOW.

At pixel level, Zheng et al. [2014] formulate the problem of joint visual attribute and object class image segmentation as a dense multi-labelling problem, where each pixel in an image can be associated with both an object-class and a set of visual attribute labels. They develop a hierarchical CRF model in which both objects and

attributes are labelled at two levels, pixels and region. Their results show that the joint training with attributes and objects helps semantic image segmentation for both object classes and attributes.

2.2.3 Attribute discovery

The above section lists works that use predefined fixed number of attributes to aid computer vision tasks. However, human efforts are usually involved in the attribute labelling process, making the representation costly to obtain. And the fixed number of predefined attributes limits the generalization ability of the learned model.

Berg et al. [2010] are the first to investigating on automatically identifying attribute vocabularies without hand labelled training data. Their approach starts with collecting images and associated text descriptions from the web. This forms a set of potential/candidate attributes from the noisy web text. They then rank the potential attributes using the learned classifiers by measuring average labelling precision on the validation data. Some redundant attributes are merged into attribute synsets. They also learn to localize an attribute with MILBoost in an image. This work, however, only focuses on semantic attribute labels from a finite candidate set.

Different from the semantic approach, Rastegari et al. [2012] present a method that discover arbitrary amount of discriminative attributes as binary code for images. The codes are learned from category labels on a per-image basis. Each bit in the codes corresponds to a hyperplane in the feature space space and can be thought of as a visual attribute whose name is not known. The attributes (hyperplane) are jointly learned in an SVM formulation to satisfy large inter-class margins and small intra-class variations. Some of the learned attributes are visually interpretable.

Similar to Rastegari et al. [2012], Feng et al. [2014] propose learn a binary encoding of objects as the presence of attributes, which are firstly learned as a dictionary of basic visual patterns. The image features are binary combinations of the attributes in a dictionary. The discriminative dictionary, the binary encoding and the classifica-

tion weights are jointly learned by alternating optimization. The objective function are designed based on that related samples should have similar attribute representations and attribute representations from different categories should be separated with a large margin.

Another work on arbitrary attributes discovery by Yu et al. [2013] proposes to automatically design category-level attributes encoded by a compact category-attribute matrix $A \in \mathcal{R}^{K \times L}$, where A_{kl} corresponds to the output of the l-th attribute classifier on the k-th category. The encoding A is learned in an algorithm which greedily adds new column with criteria to ensure category-separability, attribute-learnability, and small redundancy. The category-separability is implemented by a proximity matrix S of different categories, built upon kernels of different categories.

Chen and Grauman [2014], however, argue that category-sensitive attributes are beneficial to classification by training attributes with an importance-weighted SVM for in-class and out-class samples. But training all category-specific attribute classifiers is impractical. So they first train a sparse collection of category-sensitive attributes and construct learned weights into a 3D tensor $W \in \mathcal{R}^{M \times n \times D}$, where W_{kld} indicates the d-th dimension of the weights of the n-th attribute classifier for the m-th category, and a large amount of m-n pairs remain empty. They then use a tensor factorization method to infer the latent factors. These factors are used to generate analogous category-attributes pairs.

Shankar et al. [2015] propose discovery to visual attributes in a weakly supervised setting with deep CNN. The framework trains with objects of a single attribute label (ground-truth has multiple labels), and predicts with multiple labels (weakly supervised scenario). The goal is that the trained feature maps should only fire at specific attributes so that they are disentangled. They modify the AlexNet architecture by providing the net with pseudo-labels after some training step. The motivation behind the pseudo-labels is that CNN already learns a set of reasonably disentangled feature maps during initial stages of training and they start to get befuddled in later

stages of training due to lack of all correct labels. The generated pseudo-labels to analyse the responses of the feature maps after every fixed number of iterations, and the net eventually carves itself for attribute-specific feature maps. The pseudo-label probabilities are empirically assigned when the chances of co-occurrence of the missing attributes in the feature map are significantly high. The limitation of this work is that it can only discover predefined attributes in images.

2.2.4 Relative attributes

So far, the learned and discovered attributes in previous sections only depicts a single object or category. The relative/comparative attribute is defined by Parikh *et al.* Parikh and Grauman [2011] to capture richer semantic relationships between objects or categories, indicating the strength of an attribute in an image with respect to other images. They achieve this goal by a ranking function of the m-th attribute $r_m(\cdot)$:

$$\begin{aligned} r_m(\mathbf{x}_i) &> r_m(\mathbf{x}_j), \forall (i,j) \in \mathcal{O}_m \\ r_m(\mathbf{x}_i) &= r_m(\mathbf{x}_j), \forall (i,j) \in \mathcal{S}_m \end{aligned} \tag{2.1}$$

where $\mathcal{O}_m$ and $\mathcal{S}_m$ are ordered and unordered sets for the m-th attribute. The linear weights $r_m(\mathbf{x}_i) = \boldsymbol{w}_m^\top \mathbf{x}_i$ are learned in a rank margin maximization formulation, and the training samples are ordered and unordered object pairs in $\mathcal{O}_m$ and $\mathcal{S}_m$. The learned relative attributes are proved to aid zero-shot learning and image descriptions.

Shrivastava et al. [2012] are the first to combine binary attributes and comparative attributes to aid semi-supervised learning. The bootstrapping in semi-supervised learning is constrained by attributes. The classifiers are learned using seed labelled examples but are updated at each iteration using new labelled data. At each iteration, the large unlabelled images are jointly labelled under the attribute constraints. The most confident ones are added as new labelled data.

Following Parikh and Grauman [2011], Chen *et al.* Chen et al. [2014] extends its original formulation to multi-task learning framework with group lasso penalties on the components of weight matrix $\mathbf{W} = \mathbf{P} + \mathbf{Q}$, to capture shared features (rows of $\mathbf{Q}$) among the attributes and outlier attributes (columns of $\mathbf{P}$). It is solved by introducing slack variable$\tilde{\mathbf{W}}$ that is close to $\mathbf{W}$ and alternatively optimized in a 2-step approach. The results show the modified learning process produces better ranking accuracy and zero-shot learning performance.

As obtaining training pairs in relative attributes are more costly than in binary attributes, Liang and Grauman [2014] explore an active learning strategy for training relative attribute ranking functions, requesting human comparisons only where they are most informative. From a pool of unlabelled visual data, the proposed system requests a set of image samples that is both ambiguous/informative for the system and visually diverse for human annotator to rank. Human provides informative labels as the active labelling feedback, and the system updates current ranking function with the new information and selects a new set of comparisons (setwise, instead of pairwise).The samples with low ranking margins (high uncertainties) subjected to different cluster (high visual diversity) are chosen, and the set of comparisons are made of one sample from a cluster. By repeating the process, informative sets of comparisons are discovered.

2.3 Zero-shot Learning

2.3.1 Overview

In conventional object classification tasks, the categories in training and testing sets are identical, leaving the trained model unable to identify novel classes not present in the training set. Zero-shot learning (ZSL) is thus proposed to overcome this limitation by bridging the mid-level semantic represention between seen and unseen classes. Specifically, zero-shot learning is implemented in three steps: 1) assign mid-level

semantic to each class label; 2) learn the connection from image features to the mid-level semantic represent ions of seen classes; 3) transfer this learned connection to the mid-level semantic represent ions of unseen classes. As a result, given the semantic representation of a novel class, the model is able identify the novel object without having it in the training classes. Zero-shot learning mimics the process of human cognition, as humans are able to learn a visual concept by connecting the texual description from known concepts to the unseen one without ever seeing its actual appearance. It is recognized as an effective way for large scale visual classification since it alleviates the burden of collecting sufficient training data for every possible class.

In this section, we firstly define a general formulation of zero-shot learning, and then review some influential ZSL methods using various sources of mid-level representations, including attributes, documents, distributed word embeddings (word vectors), etc..

2.3.2 General Formulation

Given a set of class labels $\mathcal{W}_s$ and $\mathcal{W}_u$ for objects from seen and unseen classes, where $\mathcal{W}_s \cap \mathcal{W}_u = \varnothing$, and the training set $\mathcal{S}_t = \{(\mathbf{x}_n, y_n), n = 1, \cdots, N\}$, where the object descriptors $\mathbf{x}_n \in \mathcal{R}^d$ and labels $y_n \in \mathcal{W}_s$, the general formulation of zero-shot learning tasks is to learn a model $s : \mathcal{R}^d \rightarrow \mathcal{Y}$:

$$s(\mathbf{x}; \theta) = \underset{y \in \mathcal{Y}}{\operatorname{argmax}}\ F(\mathbf{x}, \phi(y); \theta), \tag{2.2}$$

where $F(\mathbf{x}, \phi(y))$ is a compatibility function for the visual feature x and the mid-level representation $\phi(\cdot)$ of class y, and θ are the model parameters. During the training phase, where $\mathcal{Y} = \mathcal{W}_s$, $F(\cdot, \cdot)$ is learned to measure the compatibility between x and $\phi(y)$. During the testing phase, the learned $F(\mathbf{x}, \phi(y))$ is applied to measure the compatibility between novel classes $y \in \mathcal{W}_u$ and testing visual samples $x \in \mathcal{X}_{unseen}$.

Finally, $s(\mathbf{x};\theta)$ assign label for x which has the highest compatibility score with it.

As can be seen, the key component ensuring the success of zero-shot learning is the mid-level semantic representation $\phi(y)$ that can describe seen and unseen classes accurately.

2.3.3 Zero-shot Learning with Attributes

Attribute is the first kind of mid-level semantic representation utilized for zero-shot learning and they are easily transferable from seen to unseen classes. There two ways to use attributes in zero-shot learning methods: attribute classifiers and label embedding. The former explicitly builds the connection from visual features to each attribute and the latter treats the attribute vectors as the general embeddings for class labels.

2.3.3.1 Attribute Classifiers

The most intuitive way for using attributes in zero-shot learning is to directly predict the attributes of unseen classes and infer the class label from predicted attributes. In Lampert et al. [2009], DIP(Direct Attribute Prediction) learns a set of probabilistic attribute classifiers from the visual features, $p(a_m|\mathbf{x})$, and predicts the novel object as the unknown class with a Bayesian probabilistic inference framework:

$$s(\mathbf{x}) = \underset{y\in\mathcal{W}_u}{\operatorname{argmax}} \prod_{m=1,\cdots,M} \frac{p(a_m^y|\mathbf{x})}{p(a_m^y)}, \tag{2.3}$$

where $p(a_m^y)$ is the pre-defined attribute prior for class y and the M is the number of attributes.

Indirect attribute prediction (IAP) is also prposed by Lampert et al. [2009], which trains the probabilistic classifiers from visual feature to seen class labels $p(y_k|\mathbf{x}), y_k \in \mathcal{W}_s$, and $p(a_m|\mathbf{x})$ is estimated by the posterior distribution of the training class labels over attributes:

$$p(a_m|\mathbf{x}) = \sum_{y_k \in \mathcal{W}_s} p(a_m|y_k)p(y_k|\mathbf{x}). \tag{2.4}$$

The estimated attribute prediction is used in the same way as in (2.3) of DAP.

Based on DAP and IAP, several works are proposed improve the way of learning the connection between attributes and object categories. Rohrbach et al. [2010] improve the quality of class-attribute association by mining from the linguistic knowledge bases. Jayaraman and Grauman [2014] propose a random forest approach to explicitly accounts for the address the unreliability of attribute predictions.

The above methods are closely related to the topics of attribute learning and attribute discovery, and more of them are described in Section 2.2.2.

2.3.3.2 Label Embedding

In last section, attribute learning and class inference are two separate processes. This makes attribute prediction suboptimal for zero-shot learning tasks because it does not account for the distribution difference for seen objects and unseen objects. To overcome this issue, recent works re-define attribute vectors as semantic embeddings for class labels and learn a model that maximizes the compatibility between the semantic embeddings and the visual embeddings (features). Typically, the matching function $F(\mathbf{x}, \phi(y))$ in (2.2) can be modelled as a bi-linear function which projects the semantic embeddings and the visual embeddings into a shared space:

$$F(\mathbf{x}, \phi(y); \Theta) = \mathbf{x}^T \Theta \phi(y), \tag{2.5}$$

where $\Theta \in \mathcal{R}^{d \times \hat{d}}$, and $\hat{d}$ is the dimensionality of the semantic embedding $\phi(y)$.

Attribute Label Embedding (ALE) Akata et al. [2013] is one of the first methods to utilize this formulation, in which each class is embedded in the space of attribute

vectors. It uses weighted approximately ranking loss Usunier et al. [2009] to ensure the correct classes rank higher than the incorrect ones. In Akata et al. [2015], Structured Joint Embedding (SJE) learns Θ with structured SVM hinge loss.

Romera-Paredes and Torr [2015] design a new loss function:

$$\min_{\mathbf{V}} \|\mathbf{X}^\top \mathbf{V}\mathbf{S} - \mathbf{Y}\|_F^2 + \lambda \|\mathbf{V}\mathbf{S}\|_F^2 + \gamma \|\mathbf{X}^\top \mathbf{V}\|_F^2 + \lambda\gamma \|\mathbf{V}\|_F^2 \tag{2.6}$$

where $\mathbf{S}$ denotes the semantic attribute matrix, $\mathbf{X}$ denotes the visual feature matrix, $\mathbf{V}$ is the parameter matrix to be learned and $\|\cdot\|_F^2$ is the Frobenius norm. This simple formulation allows efficient implementation and regularizes the projections of visual embedding $\mathbf{X}^\top \mathbf{V}$ and semantic embedding $\mathbf{V}\mathbf{S}$ independently.

Bisides bi-linear function, $F(\mathbf{x}, \phi(y))$ has other implementations. In Xian et al. [2016], a piece-wise linear compatibility function is proposed:

$$F(\mathbf{x}, \phi(y); \Theta_i) = \max_{i=1,\cdots,K} \mathbf{x}^T \Theta_i \phi(y), \tag{2.7}$$

where each projection module Θ_i is a latent variable for the current image-class pair. These latent variables factorize different visual aspects. The model is trained with a ranking based objective function which penalizes incorrect rankings of the true class for a given image.

In Zhang and Saligrama [2015], Semantic Similarity Embedding (SSE) views the visual and semantic embeddings as a mixture of seen class proportions and predicts objects of unseen class by the similarity of the mixture patterns. The similarity function is learned by sparse coding within a max-margin framework, which aligns visual embeddings of seen classes with the semantic embeddings of their corresponding labels.

In Changpinyo et al. [2016], a mapping function between the semantic label em-

bedding space and a model space is learned by constraining consistent neighbour relationship in both two spaces. In the model space, training classes and a set of phantom classes form a weighted bipartite graph. Those phantom classes form convex combinations of the coordinates of the real classes in the model space and are learned by a structured SVM hinge loss. The phantom classes builds the connection from seen classes to unseen classes.

2.3.4 Beyond Attributes

Although attributes remains the best choice for achieving the state-of-the-art performance of ZSL (Akata et al. [2015]; Zhang and Saligrama [2016]), its good performance, however, is obtained at the cost of extensive human labour to label these attributes. Since attribute vector is merely one type of semantic embedding as noted in Section 2.3.3.2, many works have explored to apply alternative semantic embeddings in zero-shot learning. In fact, as long as those mid-level representations contain shared information of different categories and thus reflect cross-category relationships, they are applicable to zero-shot learning. Among them, online documents and distributed word embeddings are mostly widely used since they are much easier to obtain than attributes.

2.3.4.1 Online documents

Online documents, such as Wikipedia articles, directly describe the concept with human-interpretable textual information. They can be viewed as an unstructured form of attributes. So early works such as Berg et al. [2010] attempt to discover the attribute representation from online document sources. Their main idea is to rank visual-ness scores of attribute candidates. The visual-ness scores are measured by labelling precision on the validation images of the learned attribute classifiers. Those approaches fall into the field of attribute discovery, and they are not applicable to zero-shot learning since the attribute discovery process requires both images and

texts for all classes, where the images are not available for unseen classes.

To deal with this issue, recent works propose to directly use online documents as mid-level representations for each class. Elhoseiny et al. [2013] is one of the first works to propose a zero-shot learning approach that only uses the textual description of categories as alternatives to attributes. The textual features are extracted as tf-idf vectors of the textual document, followed by a dimension reduction step. They first learn a domain transfer function that captures the correlation between the textual and visual domains. Then a set of classifiers on the seen classes and a regressor which maps the textual features to the learned classifiers are jointly learned. Given the textual descriptions of an unknown class, the new classifier of the unknown class is inferred from the learned domain transfer function.

Ba et al. [2015] integrate text features into a deep convolutional neural network (CNN) and use them to predict the output weights of both the convolutional and the fully connected layers. A filter layer is learned jointly with the the CNN parameters using the text features and images of the seen classes. This generates a convolutional classifier that convolves the visual feature map with a filter predicted by the text description. The classification score is generated by global pooling after convolution. Given the text description of unseen classes, the filtered CNN is able to perform zero-shot learning tasks.

Reed et al. [2016] propose to train a neural language model from scratch with raw texts for zero-shot learning. The texts describing categories are collected Amazon Mechanical Turk and are encoded using word-based or character-based RNN/CNN. The compatibility function of text-visual matching score is defined as the inner product of the texual encodings and the image features.The formulation can be factorized into two prediction models, and can be learned by minimizing both the errors of assigning image features to the training labels and the errors of assigning texual encodings to training labels.

2.3.4.2 Distributed Word Embeddings

Distributed word embeddings or word vectors are continuous vector representations of words learned from a large-scale text corpus in an unsupervised fashion. DWEs are firstly introduced in natural language processing tasks as they reflect the relationships among words in the corpus. In a word embedding space, the distances of the embeddings of similar words are closer than the distances of the embeddings of irrelevant words. For example, assuming $\mathbf{vec}(\cdot)$ is a learned embedding mapping, one can observe $\|\mathbf{vec}(dog) - \mathbf{vec}(wolf)\| < \|\mathbf{vec}(dog) - \mathbf{vec}(car)\|$. Arithmetic relationships can also be found in word embeddings, such as $\mathbf{vec}(Paris) - \mathbf{vec}(France) + \mathbf{vec}(Italy) \approx \mathbf{vec}(Rome)$. There are two most notable types of DWEs:

- **word2vec** Mikolov et al. [2013]: word2vec is a two-layer feed-forward neural network which learns to predict the relationship of a target word and its context words. The word vector is obtained from the hidden layer parameters of the predictive model. There are two variations of word2vec models: Continuous Bag-of-Words and Continuous Skip-gram. The former predicts the target word with its contexts while the latter predicts the contexts with the target word.

- **GloVe** Pennington et al. [2014]: Global Vectors for Word Representation is a count-based model that learns word vectors from aggregated global word co-occurrence statistics. The approach first constructs a large matrix of co-occurrence information of word-context pairs in a large corpus. This matrix is then factorized to yield a lower-dimensional matrix, where each row now yields a vector representation for each word. The objective is to minimize a reconstruction loss in the lower-dimensional representations which can explain most of the variance in the high-dimensional data.

The property of reflecting word relationships equips DWEs with potential towards fully automatic zero-shot learning since their unsupervised training process does not involve any human intervention. One of the earliest works using DWEs

is Socher et al. [2013], which propose a regression model that maps image feature into a semantic space of word embeddings. The regression model is learned with a neural network with *tanh* non-linearity:

$$\sum_{y \in \mathbf{W}_s} \sum_{\mathbf{x} \in \mathcal{X}_s} \|\phi(y) - W_1 \tanh(W_2 \mathbf{x})\|_2, \tag{2.8}$$

where W_1 and W_2 are the model parameters. A novelty detection mechanism is proposed with thresholds that determine whether the input images belong to the seen classes or the outliers. To identify objects of unseen classes, an isometric Gaussian distribution around each of the unseen class word vectors is assumed and class labels are assigned based on the likelihood of the mapped embeddings.

Frome et al. [2013] learn an bi-linear compatibility function as in (2.7) for image features and word embeddings with a pairwise hinge rank loss:

$$loss(x, y) = \sum_{\hat{y} \in \mathbf{W}_s - \{y\}} [margin - F(\mathbf{x}, \phi(y); \theta) + F(\mathbf{x}, \phi(\hat{y}); \theta)]_+, \tag{2.9}$$

This visual-semantic embedding model (DeViSE) is initialized from two pre-trained neural network models: a deep CNN on the visual side and a word2vec model on the semantic side. During the early stage of training only the parameters of the bi-linear projection mapping are learned. In the later stages of training the derivative of the loss function was back-propagated into the deep CNN to fine-tune visual feature output while the language model is kept fixed.

In contrast to Frome et al. [2013] and Socher et al. [2013] which casts zero-shot learning as a regression problem from the visual space to the semantic embedding space, Norouzi et al. [2014] propose Convex combination of semantic embeddings (ConSE) model, which firstly learns a classifier from training inputs to seen classes, and transfers the probabilistic predictions of the classifier beyond the seen classes,

to a set of unseen classes. Assuming $p(y|\mathbf{x})$ as the probability of an unseen image $\mathbf{x}$ assigning to seen label $y \in \mathbf{W}_s$, one can denote $y(\mathbf{x}, t)$ as the t-th most likely training label for $\mathbf{x}$ according to $p(y|\mathbf{x})$. Then the convex combination of semantic embeddings $c(\mathbf{x})$ is used to predict the image to an unseen class:

$$c(\mathbf{x}) = \frac{1}{Z} \sum_{i=1,\cdots,T} p(y(\mathbf{x}, t)|\mathbf{x}) \phi(y(\mathbf{x}, t)), \tag{2.10}$$

where Z is a normalization factor and T is the maximum number of word vectors. Given the predicted embedding $c(\mathbf{x})$, zero-shot classification is performed by finding the unseen class labels with embeddings nearest to $c(\mathbf{x})$ in the semantic space.

In Fu and Sigal [2016], a max-margin framework is proposed with similar an objecive to Socher et al. [2013], but it also incorporates distance constraints on the projected embeddings of image features, ensuring that labelled samples are projected closest to their the word embeddings of their correct labels than to others.

Chapter 3

Action Recognition with Discriminative Mid-level Representations

3.1 Introduction

Intuitively, a temporal sequence of 3D skeleton joint locations captures sufficient information to distinguish between actions, but recording such skeleton sequences was previously very expensive with the traditional motion capture technology, which limits its application. Recently, the introduction of RGB-D cameras such as Microsoft Kinect Han et al. [2013], has made the acquisition process of 3D skeleton data much easier, faster, and more practical Shotton et al. [2011], but less accurate. This advance has promoted the development of a range of skeleton-based action recognition approaches (Vemulapalli et al. [2014]; Gowayyed et al. [2013]; Wu and Shao [2014b]). The key challenge faced by these approaches has been how to extract discriminative features from the inevitably noisy sequences of pose estimates.

The trajectories of skeletal joints in space-time are a direct representation of the classes of human action in which we are interested. Earlier works (Wu et al. [2008]; Shao and Li [2013]) model human action trajectory descriptors of variable-lengths and classify them based on similarity matching between trajectories. In Gowayyed et al. [2013], for instance,an action representation is encoded in a histogram of the

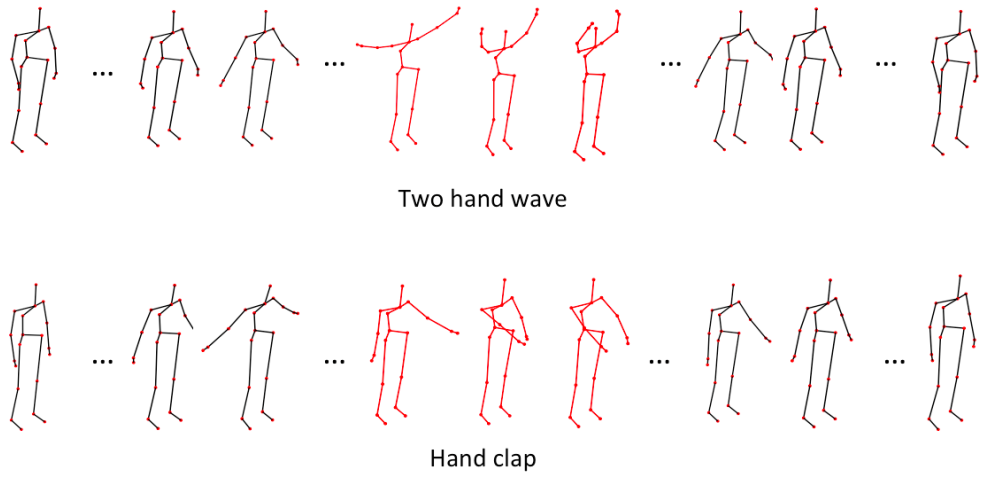

Figure 3.1: Skeleton sequences from two action classes. Only the red skeletons show significant differences between the two sequences. In this example, less than 20% of the frames are required to tell whether the skeleton is clapping or waving.

displacements of joint trajectories with respect to their orientations. Under this approach a global feature is extracted from the whole trajectory. However, only a short section of the trajectory is actually distinctive and can provide usable information about the action being undertaken. For example, as is illustrated by the two real sequences shown in Figure 3.1, the only distinctive poses are contained in the small segments when the subject moves their hands. The vast majority of the poses in the sequences are irrelevant and potentially distracting. The abundant non-informative local patterns may cause large, but irrelevant, variations between global trajectories, obfuscating the action in both training and testing.

More recent work (Yang and Tian [2012]; Zanfir et al. [2013]) has focussed on identifying discriminative patterns to create local descriptors at the frame-level. Single pose, frame-level descriptors are indeed robust for the subset of actions for which there exists a single identifying pose, but this is a small subset of the actions that are of practical interest. General action recognition, in contrast, requires analysis of the temporal relationships between individually ambiguous poses.

In contrast to the above-mentioned approaches which either represent an action

with the whole sequence or extract local features at the frame level, we argue that the discriminative information regarding an action is better captured by a short interval of trajectories. This interval usually consists of several frames. In other words, its temporal range is longer than a single frame but much shorter than the whole skeleton sequence. To extract features at this trajectory interval, we make our first contribution by designing a novel local descriptor called a trajectorylet to capture the static and dynamic pose information within the short interval.

Furthermore, as we have observed above, not all trajectorylets in a sequence are equally important for classification, and recognition performance generally benefits from focusing on discriminative sub-sequences. In skeleton-based action recognition, recent works (Zanfir et al. [2013]; Ellis et al. [2013]) directly identify (single) discriminative frames from the training set. Our approach, in contrast, does not explicitly look for the discriminative trajectorylets, but rather provides a method for creating a set of detectors that fire on specific template trajectorylets. Our approach firstly applies exemplar-SVM Malisiewicz et al. [2011] to learn a large number of candidate detectors and then selects detectors according to their discriminative performance over the trajectorylets in the training set. We further cluster detectors into multiple clusters, and remove the redundancy between the learned detectors by selecting one representative detector from each cluster. The selected detectors form a template detector set and their detection scores on a trajectorylet form the coding vector for that trajectorylet. An action level representation is then obtained by pooling all trajectorylet coding vectors. Temporal pyramid pooling can also be incorporated to capture long range temporal information within the action sequence. In extensive experiments, this framework brings significant performance improvement over state-of-the-art approaches for skeleton-based action recognition.

In summary, our *first contribution* of this chapter is the trajectorylet, a novel local descriptor that captures static and dynamic information in a short interval of joint trajectories. In our *second contribution* of this chapter, a novel framework is proposed

to generate robust and discriminative representation for action instances from a set of learned template trajectorylet detectors.

Following briefly reviewing related literature in Section 3.2, we propose the design of our local feature and detector learning method in the first half of Section 3.3, and present the action-level representation of an action instance in the second half of Section 3.3. Our framework is experimentally evaluated in Section 3.4 and summarized in Section 3.5.

3.2 Background

The key challenge in skeleton-based action recognition is how to construct the action representation from a sequence of skeletal joint locations. Some video-based methods (Messing et al. [2009]; Wang et al. [2011]) extract trajectories of multiple tracking points, and compute descriptors along them, such as HOG, HOF and MBH. For skeleton-based methods, trajectories are directly obtained from the space-time evolution of skeletal joint locations. The most straightforward way is to model the trajectory holistically, either by extracting statistics from the sequence or modelling its generative process. In Gowayyed et al. [2013], a histogram records the displacements of joint orientations over the whole trajectory. In Ohn-bar and Trivedi [2013], the action is modelled with the pairwise affinities trajectories of joint angles. In Xia et al. [2012], the action sequence is modelled by the Hidden Markov Model with quantized histogram of spherical coordinates of joint locations as frame-level feature. In Wu and Shao [2014a] and Wu and Shao [2014b], deep neural networks such as deep belief networks and 3D convolutional neural networks are adopted for spatio-temporal feature extraction from skeletal and depth data, and then a Hidden Markov Model is used to infer the action class with the learned representation. In Vemulapalli et al. [2014], 3D geometric relationships between various body parts are modelled with a Lie group to represent the whole action.

Besides directly modelling the trajectory holistically, it has also been noted that

only a small fraction of patterns of a skeletal sequence are actually distinctive and thus many approaches have been proposed to identify those discriminative patterns, whether these patterns are defined spatially or temporally.

It has been found that not all skeletal joints are informative in distinguishing one action from another, therefore it is beneficial to select a subset of joints. Ofli Ofli et al. [2012] select a subset of the most informative joints according to criteria such as mean or variance of joint angles. In Wang et al. [2012], joints are grouped into actionlets, and the most discriminative collection of such are mined via the multiple kernel learning approach. In Chaudhry et al. [2013], a subset of joints within a short-time interval is extracted according to the spatio-temporal hierarchy of the moving skeleton, and a linear combination of them is learned via a discriminative metric learning approach. In Wang et al. [2013], the distinctive set of body parts are mined from their co-occurring spatial and temporal configurations. In Chaaraoui et al. [2014], an evolution algorithm is employed to select an optimal subset of joints for action representation and classification is performed by using DTW-based sequence matching. In Du et al. [2015], the skeleton is divided sub-parts and they are fed into a deep RNN architecture, which learns the action representation at the fully connected layer.

As most of the frames in an action sequence typically represent non-distinctive static poses, features at a few discriminative temporal locations are often informative enough to represent an action. In video-based action recognition, a number of key frame selection approaches have been proposed. In Zhao and Elgammal [2008], key frames are selected by ranking the conditional entropy of the codewords assigned to the frames. In Raptis and Sigal [2013], the locations of key frames are modelled as latent variables and estimated for each action instance by dynamic programming. In recent works on skeleton-based action recognition, distinctive canonical poses Ellis et al. [2013] are learned via logistic regression, and discriminative frames Zanfir et al. [2013] are identified by their approximated confidence of belonging to a specific

action class. In Yang and Tian [2012], distinctiveness of each frame is calculated by a measurement of accumulated motion energy.

3.3 The proposed action representation

Our model utilizes the relationships between the positions of the J skeletal joints $\mathbf{j}_j = (x_j, y_j, z_j) \in \mathbb{R}^3, j = 1 \cdots J$ in the current and preceding frames to form a local trajectorylet. Because human skeleton size varies from different action instances, we perform a skeleton size normalization on the raw skeletal joints according to Zanfir et al. [2013]. We also subtract the position of the hip center $\mathbf{j}_{hip}$ from each joint and concatenate them to form a feature column: $\mathbf{j} = [\mathbf{j}_1 - \mathbf{j}_{hip}, \cdots, \mathbf{j}_J - \mathbf{j}_{hip}] \in \mathbb{R}^{3J}$, making $\mathbf{j}_{hip}$ the origin point of the coordinate system across all frames and subjects.

3.3.1 Trajectorylet

Although holistic trajectories of joints depict the movement of human body, distinctive patterns are usually overwhelmed by common ones. For example, in long-term actions such as draw circle and draw tick, only the last moment of drawing movement distinguishes them, before which both trajectories share the same movement of raising up hand for a long time. On the other hand, as depicted in Figure 3.2, frame-level local descriptors record current poses and some local dynamics, but they fail to capture the movement that spans a long temporal range. To distinguish walk from run, for instance, we need to examine the displacement and speed of the joints within a sufficient period of time, rather than the static poses. Based on these observations, we propose our trajectorylet local descriptor, which captures the static and dynamic information of trajectories in a short period of time. Compared with frame-level descriptors, trajectorylet depicts richer dynamic information. On the other hand, its temporal range is much smaller than the whole trajectory sequence and therefore it is less affected by potentially irrelevant frames.

More specifically, considering a trajectorylet of length L starting from frame t_0,

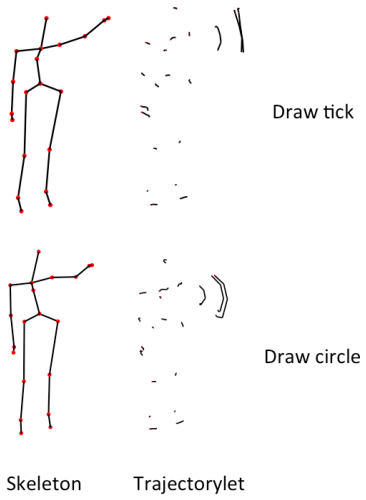

Figure 3.2: The joint coordinate information at frame-level may provide little information to distinguish between some action classes, such as the above drawing actions. One of the advantages of trajectorylets is their ability to focus on the dynamics of distinctive sections of individual actions.

we extract the static positions of the joints from each frame occurring before time $t_0 + L$:

$$\mathbf{x}_0^{t_0} = [\mathbf{j}^{t_0\top}, \mathbf{j}^{t_0+1\top}, \cdots, \mathbf{j}^{t_0+L-1\top}]^\top \in \mathbb{R}^{(L\times 3J)}. \tag{3.1}$$

In order to retrieve the dynamic information within this interval, we inspect multiple levels of temporal dynamics such as displacement and velocity.

$$\mathbf{x}_1^{t_0} = [\Delta\mathbf{j}^{t_0+1\top}, \cdots, \Delta\mathbf{j}^{t_0+L-1\top}]^\top \in \mathbb{R}^{((L-1)\times 3J)}, \tag{3.2}$$

$$\Delta\mathbf{j}^{t_0+i} = \mathbf{j}^{t_0+i} - \mathbf{j}^{t_0},\ i = 1, \cdots, L-1.$$

$$\mathbf{x}_2^{t_0} = [\Delta^2\mathbf{j}^{t_0+2\top}, \cdots, \Delta^2\mathbf{j}^{t_0+L-1\top}]^\top \in \mathbb{R}^{((L-2)\times 3J)}, \tag{3.3}$$

$$\Delta^2\mathbf{j}^{t_0+i} = \Delta\mathbf{j}^{t_0+i} - \Delta\mathbf{j}^{t_0+i-1},\ i = 2, \cdots, L-1.$$

where $\Delta\mathbf{j}^{t_0+i}$ indicates the relative joint displacements of frame $t_0 + i$ from the first frame; $\Delta^2\mathbf{j}^{t_0+i}$ indicates the joint velocities of frame $t_0 + i$ from its previous frame within the trajectorylet. The static positions of $\mathbf{x}_0^t$ store the absolute spatial location

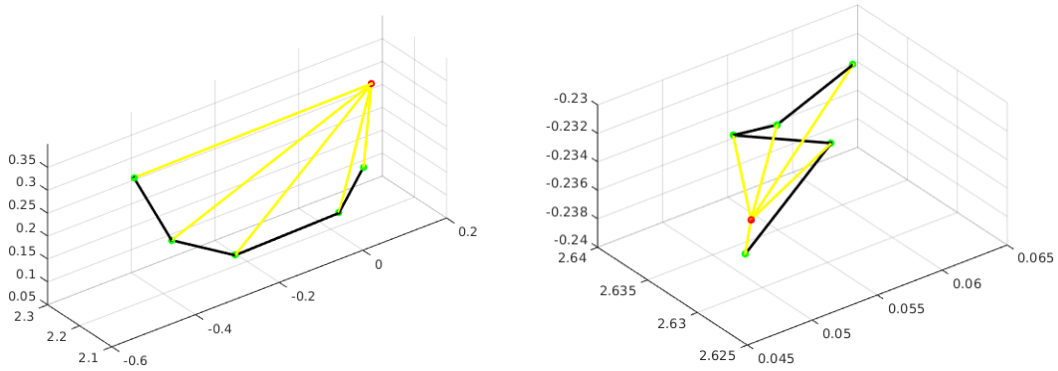

Figure 3.3: Visualization of trajetorylet of length 5 at a single joint (left hand). The red point is the position at the starting frame, and the green points are its positions at succeeding frames in this interval. The yellow segments are joint displacements from the first frame. The black segments are joint velocities at each frame. The left trajetorylet is part of *drawing circle* and the right trajetorylet is part of *high waving*. The differences between them are clearly distinguished by their positions, displacements and velocities over a short period of time.

of the trajectorylet. The temporal dynamics $\mathbf{x}_1^t$ and $\mathbf{x}_2^t$ approximate the relativel kinematic evolution within this short time interval. Combining both static and dynamic information we define the t-th trajectorylet for an action instance with F frames as

$$\mathbf{x}^{(t)} = (\mathbf{x}_0^{t\top}, \mathbf{x}_1^{t\top}, \mathbf{x}_2^{t\top})^\top \in \mathbb{R}^{(3L-3)3J}. \tag{3.4}$$

where $t = 1, \cdots, F - L$.

PCA is applied on trajectorylets to reduce the their dimension for our detector learning module. We still denote the final descriptor as $\mathbf{x}^{(t)} \in \mathbb{R}^d$, $d \leq (3L-3)3J$. Figure 3.3 visualizes components in a trajectorylet , including one static component and two dynamic components.

3.3.2 Learning candidate detectors of discriminative trajetorylet using ESVM

As we have previously discussed, only a small fraction of the trajectorylets from a sequence contain the information required to identify the associated action. Most of

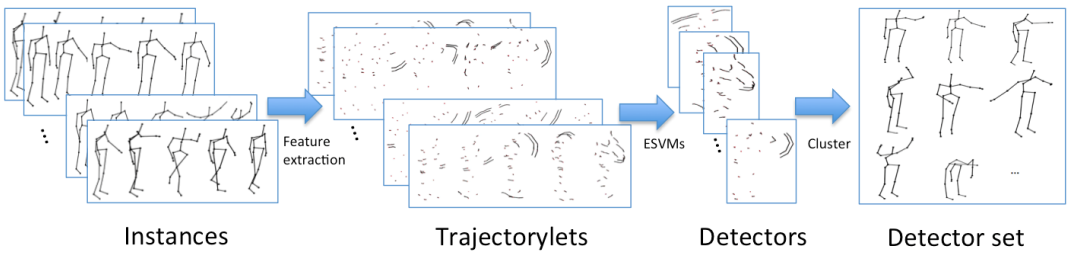

Figure 3.4: Overview of our feature learning framework.

the trajectorylets, especially those that describe static postures, are shared by multiple action classes. Our aim is to learn a set of detectors that fire on the distinctive trajectorylets. To this end, we firstly resort to exemplar-SVM (ESVM) Malisiewicz et al. [2011] to learn a large set of detectors for a large number of sampled trajectorylets, one for each sampled trajectorylet. Then for each action instance we select a few discriminative trajectorylet detectors as the candidate detectors of discriminative trajectorylet.

An ESVM learns a decision boundary that achieves the largest possible margin between an exemplar sample and a set of negative examples. If we take each trajectorylet as a positive exemplar $\mathbf{x}_E$ of its associated class c, $c = 1, \cdots C$, and trajectorylets that belong to other action classes as the negative examples, we can train an exemplar-SVM for it and formally this can be formulated as:

$$\arg\min_{\mathbf{w}_E, b_E} ||\mathbf{w}_E||^2 + \lambda_1 h(\mathbf{w}_E^\top \mathbf{x}_E + b_E) + \lambda_2 \sum_{\mathbf{x} \in \mathcal{N}_c} h(-\mathbf{w}_E^\top \mathbf{x} - b_E) \tag{3.5}$$

where $h(x) = \max(0, 1 - x)$ is the hinge loss function, and $\mathcal{N}_c$ is the negative set of trajectorylets that do not belong to class c. λ_1 and λ_2 denote the weights for the losses corresponding to the positive and negative samples respectively, and $\lambda_1 > \lambda_2$ ensures that a greater penalty will be applied to the incorrectly classified positive exemplars.

For each ESVM, the trained detector $f(\mathbf{x}) = \mathbf{w}_E^\top \mathbf{x} + b_E$ returns higher scores on

trajectorylets that are most similar to $\mathbf{x}_E$. If the current exemplar trajectorylet is common in multiple action classes, the returned trajectorylets will be abundant in multiple classes. On the contrary, if the current exemplar trajectorylet is unique for a single class, most returned trajectorylets belong to the same class as the current exemplar trajectorylet. Thus we can exploit the distribution of action classes of the returned trajectorylets to estimate the discriminative power of one detector.

Given an action instance A, we extract F_A trajectorylet descriptors $\mathbf{x}^{(t)}$, $t = 1, \cdots F_A$, and train the associated detectors $(\mathbf{w}_E^{(t)}, b_E^{(t)})$, $t = 1, \cdots F_A$. A selection method is implemented to find the most discriminative trajectorylet detector among the candidates. More specifically, we apply each detector $(\mathbf{w}_E^{(t)}, b_E^{(t)})$ to the trajectorylets $\mathbf{x}^{(i)}$, $i = 1, \cdots N$ sampled from the *whole training set* and compute the detection scores $r_{ti} = \mathbf{w}_E^{(t)\top}\mathbf{x}^{(i)} + b_E^{(t)}$. In order to measure the scores on the same scale, we adjust the trained parameters with unit norm before computing the scores. From $\mathcal{R}_t = \{r_{ti}\}_{i=1,\cdots N}$ we choose a subset $\mathcal{R}_t'$, with the top N_A detection scores, corresponding to the trajectorylets that are most compatible with current detector $(\mathbf{w}_E^{(t)}, b_E^{(t)})$. For the N_A trajectorylets detected by $(\mathbf{w}_E^{(t)}, b_E^{(t)})$, we denote $h_t^{(c)}$ as the number of trajectorylets belonging to action class c . The histogram $H_t = [h_t^{(1)}, \cdots, h_t^{(C)}]^\top \in \mathbb{R}^C$ gives a clear view of the distinctiveness of detector $(\mathbf{w}_E^{(t)}, b_E^{(t)})$.

If H_t is flat across many classes, $\mathbf{x}^{(t)}$ is a common pattern shared by many classes and its detector is therefore not distinctive. If the H_t is centered mostly at the correct class, trajectorylet $\mathbf{x}^{(t)}$ is a distinctive pattern for this class and hence $(\mathbf{w}_E^{(t)}, b_E^{(t)})$ is an effective detector of this distinctive pattern. Figure 3.5 visualizes the two typical cases of H_t and their associated trajectorylets. The pattern of the distinctive trajectorylet clearly matches the human intuition of *drawing a circle* while the pattern of the non-distinctive trajectorylet is ambiguous as it can also be seen as a part of *drawing a tick*, *boxing* or many other action classes in the dataset. More examples of representative trajectorylets are shown in Figure 3.10. In practice, if the correct class corresponding to $(\mathbf{w}_E^{(t)}, b_E^{(t)})$ is c, we denote $P_t = h_t^{(c)}/N_A$ as the ratio of correctly

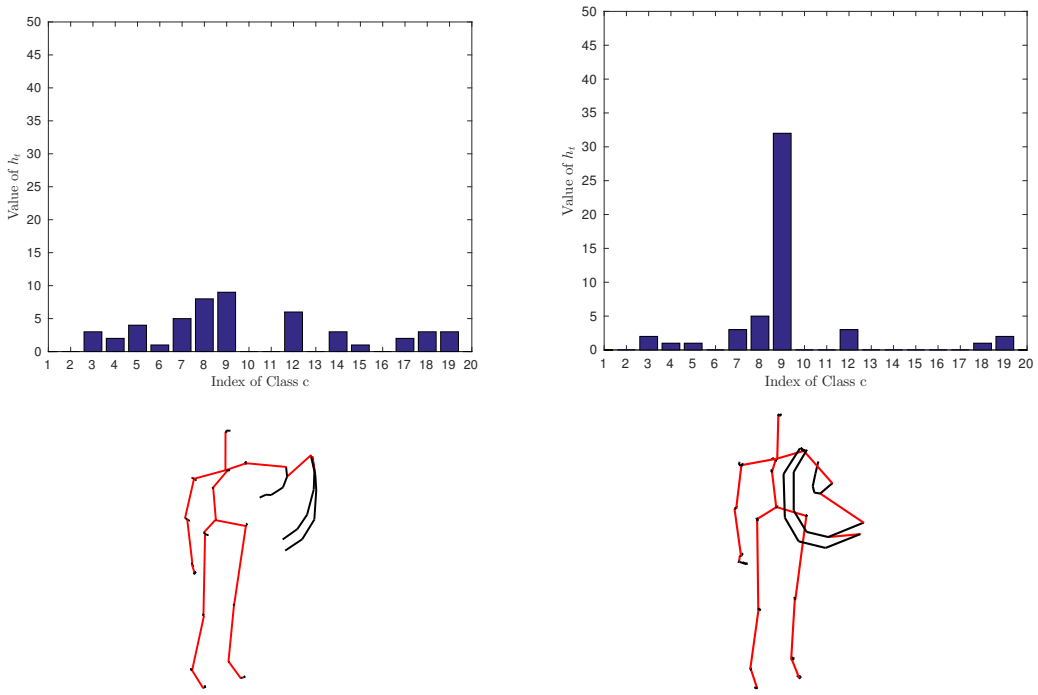

Figure 3.5: Example of the histogram of class distribution of trajectorylets detected by $(\mathbf{w}_E^{(t)}, b_E^{(t)})$. In this case we build the histogram from top $N_A = 50$ trajectorylets fired by $(\mathbf{w}_E^{(t)}, b_E^{(t)})$ in MSR Action dataset, and the total class number is 20. Upper Left: the trajectorylet is not distinctive as its detector also fires on most trajectorylets of other classes. Upper Right: a trajectorylet detector fires mostly at Class 9, *drawing a circle*, indicating the associated trajectorylet is a distinctive pattern for Class 9. Bottom Left: the trajectorylet that corresponds to the above non-distinctive detector. Bottom Right: the trajectorylet associated to the above distinctive detector.

detected trajectorylets and a detector with higher P_t is selected because it fires primarily on trajectorylets with the same class of it, verifying the distinctiveness of this detector. We summarize this approach in Algorithm 1.

3.3.3 Template detector set

As the detectors are discovered from every action instance, the size of the detector set grows with the number of training instances, which will lead to a very high-dimensional action representation and make the computation intractable. On the other hand, the above method might select similar distinctive detectors multiple

Algorithm 1 Find discriminative detectors for an action instance

Input: Training action instance A of class c, trajectorylets within it $\{\mathbf{x}^{(t)}\}_{t=1\cdots F_A}$; sampled training trajectorylets $\mathcal{X} = \{\mathbf{x}^{(i)}\}_{i=1,\cdots N}$; number of trajectorylets to retain: N_A; maximum number of detectors to be selected for the instance: M_A.
Initialize: Set of discriminative detectors for instance A: $\mathcal{D}_A = ;$ number of discriminative detectors selected for the instance $m_A = 0$.
for $t = 1 \cdots F_A$ **do**
  · Solve ESVM $\rightarrow (\mathbf{w}_E^{(t)}, b_E^{(t)})$.
  · Compute detection scores on sampled trajectorylet set
  · Compute H_t from the top N_A scored samples.
  · Compute the ratio of correctness P_t of H_t.
end
· Sort P_t by magnitude, storing the resulting (sorted) indeces in **s**.
for t *in* **s do**
  · $\mathcal{D}_A = \mathcal{D}_A \cup (\mathbf{w}_E^{(t)}, b_E^{(t)})$.
  · $m_A = m_A + 1$.
  if $m_A \geq M_A$ **then**
    · Break.
  end
end
Output: Discriminative detectors for instance A: $\mathcal{D}_A$.

times, resulting in a highly redundant detector set. To control the size of detector set and remove the redundancy of candidate detector set, we perform spectral clustering on candidate detectors and then select one detector from each cluster as the final detector set used for trajectorylet encoding. To build the affinity graph for spectral clustering, we need to specify the similarity measurement between two detectors. Here we measure this similarity by considering the "active detection scores" of two detectors which refer to the detection scores with positive values. We evaluate it by firstly calculating detection scores on N sampled trajectorylets and setting negative detection scores to zero. This process gives a N dimensional active detection score vector $\mathbf{r}_d$ for each detector and the similarity between two detectors are measured as follows:

$$q_{dd'} = \frac{\mathbf{r}_d^\top \mathbf{r}_{d'}}{||\mathbf{r}_d|| \cdot ||\mathbf{r}_{d'}||} \tag{3.6}$$

where $\|\cdot\|$ represents the l^2 norm, and $\mathbf{r}_d$ and $\mathbf{r}'_{d'}$ denote the active detection score vectors for the two compared detectors. The value $q_{dd'}$ measures the similarity between two detectors and is used to build the affinity matrix $\mathbf{Q}$ for the detector set $\mathcal{D}$, that is, $\mathbf{Q} = [q_{dd'}]_{d,d'=1,\ldots,D}$. We apply spectral clustering to $\mathbf{Q}$ and obtain $K < D$ clus-

ters of detectors. The detectors within the same cluster fire on similar trajectorylets. From each cluster, we select a representative detector that produces the highest score on the sampled trajectorylets. In practice, given a sufficient large K, the collection of representative detectors can cover all discriminative trajectorylets. We call this collection the template trajectorylet detector set.

torylet and max-pool those detection scores to obtain the action representation. Formally, let $\mathbf{x}_i^j \in \mathbb{R}^n$ be the j-th trajectorylet of the i-th action, and $(\mathbf{w}_k, b_k)$ be the k-th detector in the template detector set. We define the action representation for the i-th action $\mathbf{\Phi}(\mathbf{x}_i) = [\Phi_k(\mathbf{x}_i)]_{k=1,\dots,K}$ as:

$$\Phi_k(\mathbf{x}_i) = \max_j \left(\mathbf{w}_k^\top \mathbf{x}_i^j + b_k\right),\ k = 1, \cdots, K. \tag{3.7}$$

We use a one-versus-all SVM to classify actions among the C action classes $y_i \in \{1, \cdots, C\}$.

The learned feature mapping $\mathbf{\Phi}(\cdot)$ governed by the template detector set serves as a global descriptor of the action instance. It maps temporally continuous trajectorylets into a higher-level representation. Also, $\mathbf{\Phi}(\cdot)$ can not only map a complete sequence of action, but also works for a temporal sub-sequence. This allows us to build a temporal pyramid representation of the action instance. For a 3-level temporal pyramid, the sub-sequences are $F^{(p)}, p = 1, \cdots, 7$, and the k-th dimension of subfeature $\mathbf{\Phi}^{(p)}(\mathbf{x}_i)$ for sub-sequence p is

$$\Phi_k^{(p)}(\mathbf{x}_i) = \max_{j \in F^{(p)}} \left(\mathbf{w}_k^\top \mathbf{x}_i^j + b_k\right). \tag{3.8}$$

The concatenated $\mathbf{\Psi}(\cdot) = [\mathbf{\Phi}^{(1)}(\cdot)^\top, \cdots, \mathbf{\Phi}^{(7)}(\cdot)^\top]^\top$ incorporates the temporal information of the skeleton sequence. Therefore we are able to train a one-versus-all SVM with this feature that takes into account the global temporal information of the whole action sequence.

3.4 Experiments

We organize the experimental evaluation in four parts. We first compare our proposed method against other state-of-the-art methods on three standard datasets obtained from the Kinect sensor and one less noisy dataset obtained from motion capture. Then we analyze the performance of our method under different parameter settings. Since our method consists of two modules, the trajectorylet descriptor and the template detector learning based middle-level feature representation, we conduct two experiments to separately evaluate their impacts on the classification performance. To examine the first module, we compare our descriptor against the descriptor of Zanfir et al. [2013], which is most related to our trajectorylet descriptor, by keeping the other settings of the recognition system the same. We also compare our descriptor with its several alternative variants. To examine the second module, we compare our method with alternative way to obtain constructed from three state-of-the-art middle-level feature representation methods: VLAD Jegou et al. [2010], LSC Liu et al. [2011], and LLC Wang et al. [2010].

Implementation details: The ESVMs are implemented by liblinear Fan et al. [2008], which produces about 10 candidate detectors per second on an Intel Core i7 CPU at 3.40GHz. We set the regularization parameters as $\lambda_1 = 10$ and $\lambda_2 = 0.01$ for all ESVMs. The training time of all ESVMs depends on the number of trajectorylets in each dataset, which varies from 0.5 hour to 5 hours. On average, our unoptimized MATLAB code trains an ESVM within 0.1 second. The training and inference of global descriptors take around 0.09 to 0.04 second and 0.01 to 0.02 second respectively. Our unoptimized MATLAB implementation consumes 1.2 GB memory. As seen, the most time consuming component of our algorithm is the training of all ESVMs, but it should also be noted that the training process of each ESVM is completely independent. This means that it is possible to develop a parallel training scheme to train many trajectorylets simultaneously and reduce the training time tremendously. We leave this parallel scheme to future work. The dimensionality of

AS1	AS2	AS3
Horizontal arm wave	High arm wave	High throw
Hammer	Hand catch	Forward kick
Forward punch	Draw x	Side kick
High throw	Draw tick	Jogging
Hand clap	Draw circle	Tennis swing
Bend	Two hand wave	Tennis serve
Tennis serve	Forward kick	Golf swing
Pickup & throw	Side boxing	Pickup & throw

Table 3.1: The classes in the three action subsets of the MSR Action3D dataset.

Protocol of Li et al. [2010]	AS1	AS2	AS3	Average
3DBag Li et al. [2010]	72.9	71.9	79.2	74.7
HO3DJ Xia et al. [2012]	88.0	85.5	63.5	79.0
EigenJoints Yang and Tian [2012]	74.5	76.1	96.4	82.3
Skeletal QuadsEvangelidis et al. [2014]	88.4	86.6	94.6	89.9
Cov3DJ Hussein et al. [2013]	88.0	89.3	94.3	90.5
HOD Gowayyed et al. [2013]	92.4	90.1	91.4	91.2
Lie Group Vemulapalli et al. [2014]	95.3	83.9	98.2	92.5
EJS Chaaraoui et al. [2014]	91.6	90.8	97.3	93.2
HBRNN Du et al. [2015]	93.3	94.6	95.5	94.5
Moving Pose Zanfir et al. [2013]	96.4	91.6	99.1	95.7
Ours	96.4	97.5	100.0	97.9

Table 3.2: Results on 3 subsets of the MSR Action3D dataset.

trajectorylets is reduced to 50% percent of it by PCA. As the testing data will not be known in advance, the PCA coefficients μ and covariance matrix are learned from the training data only. Unless indicated otherwise, the length of trajectory descriptor is set to $L = 5$. The regularization parameter for the final one-versus-all SVM is determined by a five-fold cross-validation. We apply a 3-level temporal pyramid on MSR DailyActivity3D only, because it contains complex actions which involves several sub-actions and the long-range temporal information can be useful in such a case.

3.4.1 MSR Action3D

The MSR Action3D dataset consists of human actions expressed with skeletons composed of 20 3D body joint positions in each frame. The 20 joints are connected by 19 limbs. There are 20 action classes performed by 10 subjects for 2 or 3 times each, making up 567 action instances. Each action instance contains a temporal sequence of a moving skeleton, usually in 30-50 frames. As in Wang et al. [2012] and Zanfir et al. [2013], we drop 10 instances because they contain erroneous data. The experiment setup is that of a cross-subject test Li et al. [2010], i.e. instances of half of the subjects are used for training and instances of the other half subjects are used for testing. We construct H_t with top responding $N_A = 50$ trajectorylets, and select $M_A = 10$ best detectors for each training instance. We use the clustering method of section 3.3.3 to obtain the template trajectorylet detector set. The final number of template detectors is set to $K = 500$.

In Table 3.2, we compare our approach with other state-of-the-art methods using the protocol of Li et al. [2010], by which the 20 action classes are grouped into 3 action subsets AS1, AS2, and AS3. The training and testing is performed on each action set separately. AS1 and AS2 group actions with similar movements while AS3 group complex actions. The action classes of each action subset are listed in Table 3.1. On average, our proposed method is more accurate than all other methods. On AS2, all other methods get moderate accuracy and in contrast our method outperforms the second best by 5.9%. Note that in Table 3.2, we use the code of Zanfir et al. [2013] to obtain this result, as the original work did not report the results according to the protocol of Li et al. [2010]. On AS3, our method achieves perfect recognition.

In Table 3.3, a more challenging protocol of Wang et al. [2012] is used. Here the model is trained and tested over all 20 action classes. The results show that our method still obtains a highly accurate recognition rate, outperforming the current best state-of-the-art by a margin of 4.2%. The confusion matrix of our method on this dataset under the second protocol is displayed in Figure 3.6, where 16 of 20 action

Protocol of Wang et al. [2012]	Accuracy
Recurrent Neural Network Martens and Sutskever [2011]	42.5
Dynamic Temporal Warping Müller and Röder [2006]	54.0
Canonical Poses Ellis et al. [2013]	65.7
DBN+HMM Wu and Shao [2014b]	82.0
JAS (skeleton data only) Ohn-bar and Trivedi [2013]	83.5
Actionlet Ensemble Wang et al. [2012]	88.2
HON4D Oreifej and Liu [2013]	88.9
Lie Group Vemulapalli et al. [2014]	89.5
LDS Chaudhry et al. [2013]	90.0
Pose based Wang et al. [2013]	90.2
Moving Pose Zanfir et al. [2013]	91.7
Ours	95.9

Table 3.3: Results on the entire MSR Action3D dataset.

classes are perfectly classified. The only highly misclassified class is *hammer*, because its distinctive pattern involves human-object interaction, which is not captured by the skeleton data.

3.4.2 MSR DailyActivity3D

In MSR DailyActivity3D, there are 16 action classes performed by 20 subjects twice, making up 320 action instances. Each subject performs an action class in two variants (e.g. sitting versus standing, or in front of versus behind an object). This dataset has longer sequences, usually in 100-300 frames. We still follow the cross-subject test in protocol of Wang et al. [2012] and Zanfir et al. [2013], where training and testing are conducted over all action classes. Because this dataset contains more local information than MSR Action3D, we construct H_t with top responding $N_A = 50$ trajectorylets, select $M_A = 15$ best detectors for each training instance, and reduce the final number of clustered detectors to $K = 500$.

We compare our approach with other state-of-the-art methods in Table 3.4. As the purpose of this experiment is to address skeleton-based action recognition, some best reported results (Oreifej and Liu [2013]; Wang et al. [2012]) on this dataset using additional RGB-D data are not comparable to our method, and therefore we cite the result

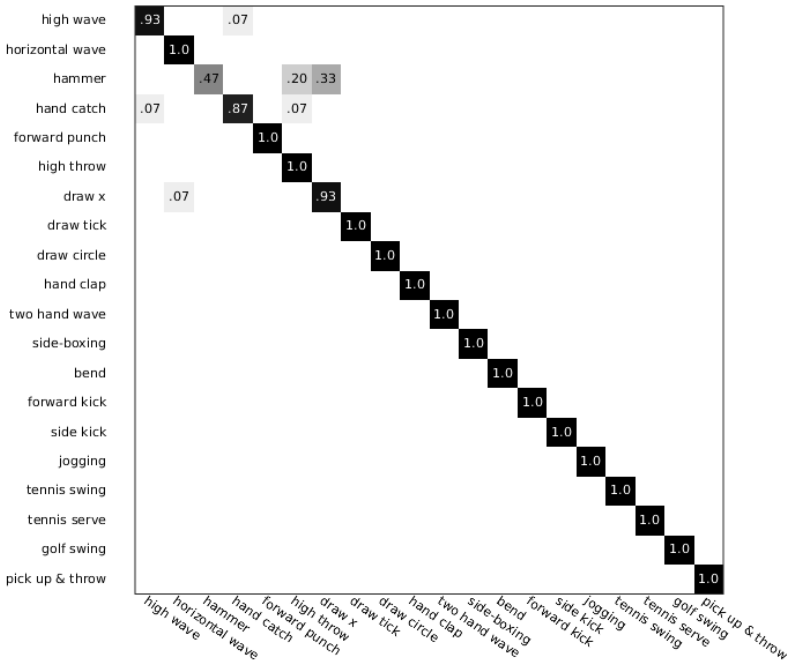

Figure 3.6: Confusion matrix of our approach on the MSR Action3D dataset: except for the *hammer* class, all other action classes are classified with more than 80% accuracy. 16 out 20 action classes are perfectly classified.

of Wang et al. [2012] using only skeleton data. Although MSR DailyActivity3D share the same data structure as the MSR Action3D, it is much more challenging because: 1) the activities are complex combinations of multiple sub-actions, 2) human-object interaction information is not available in skeleton data, 3) partial occlusion by interacting objects causes the skeleton data to be highly noisy. However, the results show that our approach still outperforms all other state-of-the-art methods. Note that although the reported result in Zanfir et al. [2013] is 73.8%, we never achieved this accuracy with their code due to environmental factors. For a fair comparison, we used the result 70.6%, which is the best performance under the same environment and setting with our approach. As shown in Figure 3.7, most of the poorly classified actions involves interaction with objects, such as *read book*, *call cellphone*, and *use laptop*. On the other hand, non-interactive action classes like *cheer up*, *walk*, and *sit down*,

Methods	Accuracy
Dynamic Temporal Warping Müller and Röder [2006]	54.0
Actionlet Ensemble (skeleton data only) Wang et al. [2012]	68.0
Moving Pose Zanfir et al. [2013]	70.6
Ours	75.0

Table 3.4: Results on the MSR DailyActivity3D dataset.

are recognized with high accuracy. This demonstrates that our method is able to capture distinctive patterns of actions in terms of "movement", but may be confused if some actions share similar "movement" patterns despite the presences of different interacting objects, because they are not described in the skeleton data.

3.4.3 MSRC-12

MSRC-12 Fothergill et al. [2012] is a much larger scale dataset compared to the first two datasets. It contains 12 action classes performed by 30 subjects instructed with different sources (text/image/video). As this dataset is originally designed for detection tasks, multiple action instances are performed continuously in a single sequence. We use the annotation of Hussein et al. [2013], which marks the start and end frames of each action instance, to accommodate MSRC-12 to action recognition tasks. This ends up with total 6,244 annotated action instances of 100. We apply the same parameter settings of MSR Action3D for N_A, M_A and K. We follow two variants of cross-subject test in Hussein et al. [2013]: leave-one-out and 50% subject split. In the leave-one-out test, instances of 29 out of the 30 subjects are used for training and remaining instances are used for testing. The final result is averaged over 30 experiments where each of the subjects is left for test for once. The 50% subject split test is similar to previous cross-subject tests, where half of the subjects are used for training and the other half are kept for testing, except that the split is randomly chosen. Following Hussein et al. [2013], we report the averaged results over 20 random splits.

As seen in results of Table 3.5, on both protocols our method outperforms the baseline method significantly. This experiment also demonstrates that our method is

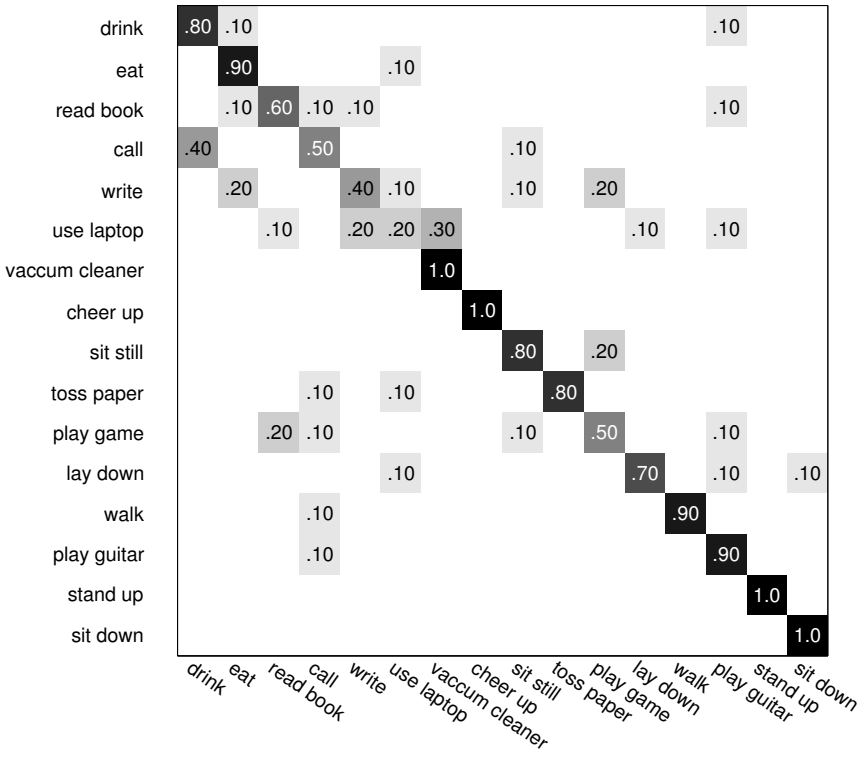

Figure 3.7: Confusion matrix of our approach on the MSR DailyActivity3D dataset: although this is a challenging dataset for skeleton-based action recognition, 11 out of 16 classes are classified with more 70% accuracy.

able to generalize to large scale action recognition tasks by tuning the parameters in relatively small datasets.

3.4.4 HDM05

To further test the generalization ability of our method, we now evaluate it on HDM05 Müller et al. [2007], a motion capture dataset, which is less noisy than the previous three Kinect datasets. The skeleton is composed of 31 joints instead of 20 joints in the Kinect datasets. We use the subset defined in Ofli et al. [2012], which contains 11 action classes performed by 5 subjects, totalling 249 action instances. For computational reasons, we downsample the frame rate from 120 fps to 30 fps and apply the same parameter settings of MSR Action for N_A, M_A and K in this experiment. We follow the protocol of Ofli et al. [2012]: 3 subjects (142 instances) are for

Methods	Accuracy
Cov3DJ Hussein et al. [2013] (Leave-one-out)	93.6
ours (Leave-one-out)	95.1
Cov3DJ Hussein et al. [2013] (50% subject split)	91.7
ours (50% subject split)	94.9

Table 3.5: Results on the MSRC-12 dataset.

training and 2 subjects (109 instances) are testing.

Methods	Accuracy
SMIJ Ofli et al. [2012]	84.4
Skeletal QuadsEvangelidis et al. [2014]	93.9
Cov3DJ Xia et al. [2012]	95.4
HOD Gowayyed et al. [2013]	97.3
ours	96.3

Table 3.6: Results on the HDM05 dataset.

Table 3.6 shows that our method achieves the state-of-the-art-results with motion capture data. Although our method does not significantly outperform all baselines on the less nosy motion capture data, the experiment clearly confirms that our method is applicable to data sources of different skeleton configurations and noise levels, and is better at handling noisy data. As we downsample the frame rate from 120 fps to 30 fps, it would cause information loss of actions. However, a frame rate as high as 120 fps is not practically necessary for conventional action recognition. In our experiment , ultra-high frame rate has little effect on the performance except for bringing extra computational cost. As for low frame rates such as 15 fps, the performance drop is obvious (90.5% at 15 fps vs 96.3% at 30 fps). We believe a frame rate consistent with normal RGBD devices would be enough for accurate action recognition. For example, humans can easily recognize actions from normal videos of 24 fps and this ability does not seem to improve when they see actions from 60 fps videos.

3.4.5 Parameter analysis

In this section we analyse how the parameter settings affect the performance. Using the same protocol of Wang et al. [2012], we provide results of MSR Action3D dataset from other parameter settings. Figure 3.8 illustrates the performances of our method as K ranges from {25, 50, 100, 200, 300, ... , 1000}, while keeping $N_A = 50$ and $M_A = 10$. When we set the size of detector set more than 500, the results tend to converge to a value above 94.5%. Table 3.7 presents results of choosing different pairs of M_A and N_A while keeping $K = 500$.

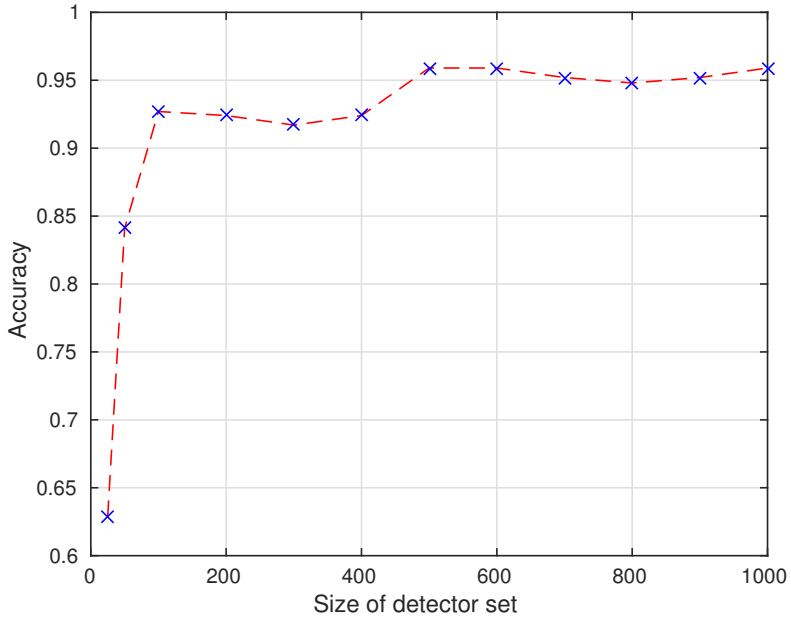

Figure 3.8: Recognition accuracies obtained from varying K on the MSR Action 3D dataset: when $K \geq 500$ the results become stable.

For the MSR DailyActivity3D dataset, Figure 3.9 illustrates the performances of our method as K ranges from {25, 50, 100, 200, 300, ... , 1000}, while keeping $N_A = 50$ and $M_A = 15$. When K is set to more than 500, the results become stable. The effect of choosing different pairs of M_A and N_A is listed in in Table 3.8. When M_A is large enough, the results variation becomes small. It can be observed that, on both datasets, there are multiple choices of parameters that are able to produce the

M_A \ N_A	5	10	15	20	30	50
5	91.7					
10	92.7	93.4				
20	93.1	93.1	94.1	94.8		
30	94.8	95.2	94.1	93.8	94.8	
50	95.5	95.9	95.9	94.8	94.8	94.2

Table 3.7: Results from different pairs of the M_A and N_A on MSR Action3D: we can obtain the best performance from multiple choices.

M_A \ N_A	5	10	15	20	30	50
5	68.7					
10	68.1	69.4				
20	68.7	71.2	70.0	69.4		
30	70.0	73.1	73.8	71.2	71.2	
50	73.1	74.3	75.0	75.0	74.3	71.9

Table 3.8: Results from different pairs of the M_A and N_A on MSR DailyActivity3D.

optimal result and this verifies the robustness of our approach.

Table 3.9 shows the results under different temporal pyramid settings for the three datasets from Kinect sensors. A typical 3-level pyramid is the best choice for MSR DailyActivity3D as low level pyramids fail to grasp the temporal information while higher level ones brings too much noise. On the other hand, when temporal pyramid is applied to MSR Action3D, the performance is worsened. For MSRC-12, we observe no significant differences among the temporal pyramid settings. It is interesting to observe that for sequences that contain a single/simple action like MSR Action3D and MSRC-12, discriminative trajectorylets are able to accurately recognize actions without long term temporal information. Only for complex activities composed of multiple and repetitive actions, a long term temporal modelling is needed.

3.4.6 Power of local trajectorylet descriptor

The moving pose descriptor proposed in Zanfir et al. [2013] captures local information at frame-level of human skeleton actions. Our trajectorylet can be seen as a

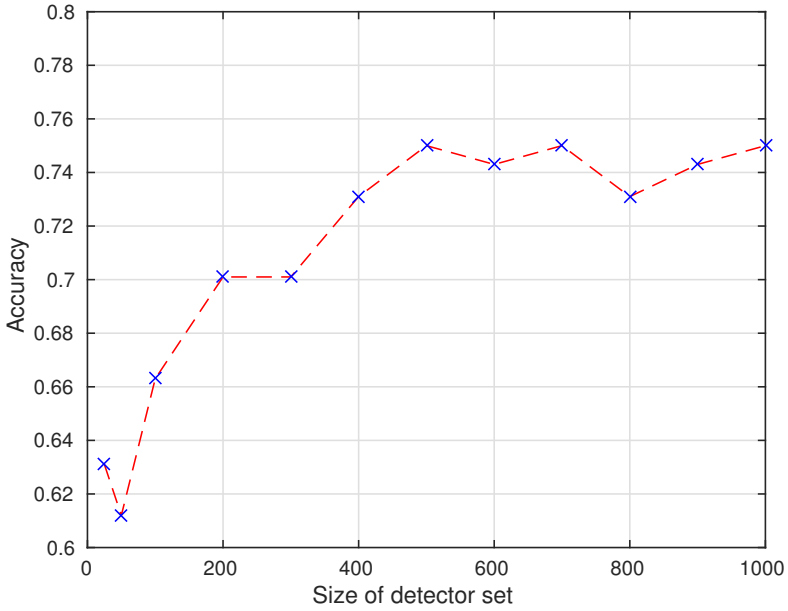

Figure 3.9: Recognition accuracy obtained from varying K on the MSR Daily Activity 3D dataset: when $K > 400$ the results become stable.

TP level	1	2	3	4
Action	95.9	92.4	89.7	N/A
DailyActivity	66.3	70.6	75.0	68.8
MSRC	94.9	95.2	95.2	N/A

Table 3.9: Results obtained from different temporal pyramid levels on MSR Action3D, MSR DailyActivity3D and MSRC-12 datasets.

natural extension of it in the sense that we extend the dynamic information from frame-level range to a longer temporal range. In order to demonstrate the power of our descriptor we now apply our template detector learning framework to moving pose descriptor and compare its performance with that of trajectorylet.

In order to evaluate the effect of varying L on performance we have varied the length our trajectorylet from (3, 5, 7). Table 3.10 shows that using the same detector learning and classification approach, trajectorylets achieve better results on both datasets for all tested values of L. As seen, this extension of moving pose descriptor is superior over the original design. It is worth noting that performance does not

necessarily improve as the length of trajectorylets increases. A moderate length of trajectorylet ($L = 5$) leads to the best performance.

We also test the effect of using different components of the trajectorylet descriptor. In our experiment, we examine the performance of single dynamic components, including static pose $\mathbf{x}_0$, relative joint displacement $\mathbf{x}_1$, velocity $\mathbf{x}_2$, and their combinations. We also further define an acceleration component analogous to (3.2) and (3.3):

$$\begin{aligned} \mathbf{x}_3^{t_0} &= [\Delta^3\mathbf{j}^{t_0+2^\top}, \cdots, \Delta^3\mathbf{j}^{t_0+L-1^\top}]^\top \in \mathbb{R}^{((L-3)\times 3J)}, \\ \Delta^3\mathbf{j}^{t_0+i} &= \Delta^2\mathbf{j}^{t_0+i} - \Delta^2\mathbf{j}^{t_0+i-1},\ i = 3, \cdots, L-1. \end{aligned} \tag{3.9}$$

The results of varying settings of a trajectorylet with $L = 5$ are listed in Table 3.11. We find that the dynamic components of $\mathbf{x}_1$ and $\mathbf{x}_2$ alone do not show promising results, especially on the MSR DailyActicity Dataset. However, when combined with static $\mathbf{x}_0$, the performance is significantly improved. Table 3.11 also shows that the additional acceleration component in (3.9) does not improve the performance. Additionally, we notice that the dynamic components $\mathbf{x}_1$ and $\mathbf{x}_2$ of MSR DailyActicity dataset perform much worse than their counterparts in MSR Action dataset. We believe that the performance discrepancy may be due to two different properties of the two datasets. (1) As the skeletons are badly captured in MSR DailyActivity dataset, the positions of joints become more unstable over time. This makes the velocity and displacement information even noisier than the static positions. (2) The differences between different categories in MSR DailyActivity is more subtle, e.g. *drink* vs. *call*, than MSR Action. Thus, less discriminative features like $\mathbf{x}_1$ and $\mathbf{x}_2$ which performed well on the simple MSR Action dataset may no longer perform as well in the more challenging MSR DailyActivity dataset.

Descriptors	MSR Action	MSR DailyActivity
Moving Pose Zanfir et al. [2013]	91.7	71.3
Ours($L = 3$)	93.1	72.5
Ours($L = 5$)	95.9	75.0
Ours($L = 7$)	95.9	73.1

Table 3.10: Comparison of using different descriptors.

Component	MSR Action	MSR DailyActivity
$\mathbf{x}_0$	92.4	72.5
$\mathbf{x}_1$	91.7	50.3
$\mathbf{x}_2$	90.3	42.5
$(\mathbf{x}_0, \mathbf{x}_1)$	93.8	73.1
$(\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2)$	95.9	75.0
$(\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$	95.9	74.3

Table 3.11: Comparison of different using different components of trajectorylet ($L = 5$).

3.4.7 Power of template detector learning

Our method generates action representation from learned detector set of discriminative trajectorylets. In this section, we compare this method with three state-of-the-art bag-of-feature techniques that learn global feature for the action instance from the same local trajectorylet features: VLAD (vector of locally aggregated descriptors)Jegou et al. [2010], LLC (locality-constrained linear coding)Wang et al. [2010], and LSC (localized soft-assignment coding) Liu et al. [2011].

We train codebook of the same size $K = 128$ with k-means for all three methods, and set the neighbourhood size of codewords as $\kappa = 10$ for LSC and LLC. The results listed in Table 3.12 show, for the task of action recognition, our proposed feature learning framework produces the most discriminative action representation, compared with the state-of-the-art methods. Figure 3.10 illustrates some trajectorylets fired on the template detector set of MSR Action3D. It is clear that they show representative patterns for the corresponding action classes.

Method	MSR Action	MSR DailyActivity
VLAD Jegou et al. [2010]	83.1	51.9
LLC Wang et al. [2010]	90.7	65.6
LSC Liu et al. [2011]	92.1	66.9
Ours	95.9	75.0

Table 3.12: Comparison of feature learning methods.

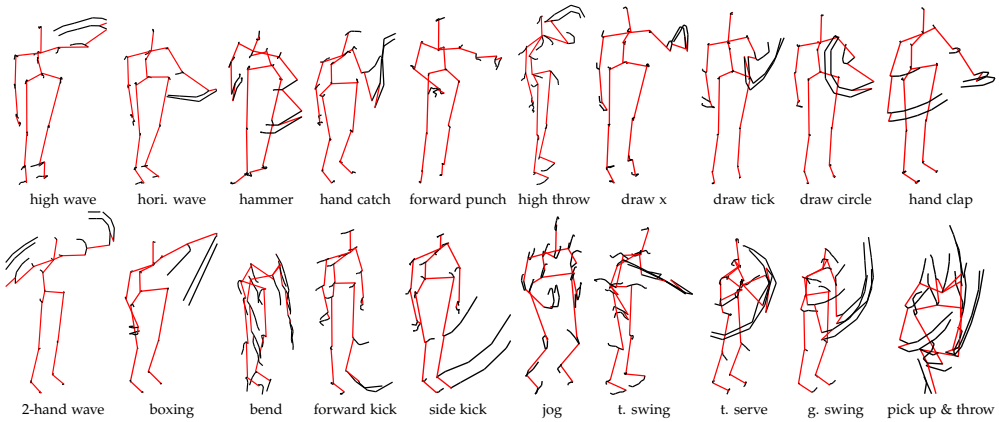

Figure 3.10: Some examples responding on the template detector set of MSR Action3D. The black curves represent the velocity components of current trajectorylets with $L = 5$. The fact that the our approach identifies discriminative patterns of movement seems clear.

3.5 Summary

This chapter describes an effective skeleton-based action approach that achieves high accuracy on the relevant benchmark datasets. The keys to this performance are two factors. We propose trajectorylet, a novel local descriptor that captures static and dynamic information in a short interval of joint trajectories. We also devise a novel framework to generate robust and discriminative representations for action instances by learning a set of distinctive trajectorylet detectors. On various benchmark datasets acquired from the Kinect sensor, our method outperforms, to our knowledge, all existing approaches by a significant margin. We also separately demonstrate the validity of our local descriptors and template detector learning method.

Chapter 4

Zero-shot Learning with Online Textual Documents

4.1 Introduction

Unlike traditional object classification tasks in which the training and test categories are identical, zero-shot learning aims to recognize objects from classes not seen at the training stage. It is recognized as an effective way for large scale visual classification since it alleviates the burden of collecting sufficient training data for every possible class. The key component ensuring the success of zero-shot learning is to find an intermediate semantic representation to bridge the gap between seen and unseen classes. In a nutshell, with this semantic representation we can first learn its connection with image features and then transfer this connection to unseen classes. So once the semantic representation of an unseen class is given, one can easily classify the image through the learned connection.

Attributes, which essentially represent the discriminative properties shared among both seen and unseen categories, have become the most popular semantic representation in zero-shot learning (Farhadi et al. [2009]; Ferrari and Zisserman [2008]; Torresani et al. [2010]; Lampert et al. [2009]; Yao et al. [2011]). Although the recent use of attributes has led to exciting advances in zero-shot learning (Fu et al. [2015]; Akata et al. [2015]; Zhang and Saligrama [2015]), the creation of attributes still relies on much human labour. This is inevitably discouraging since the motivation for zero-

shot learning is to free large-scale recognition tasks from cumbersome annotation requirements.

To remedy this drawback and move towards the goal of fully automatic zero-shot learning, several recent works (Socher et al. [2013]; Frome et al. [2013]; Norouzi et al. [2014]) have explored the possibility of using the easily accessed online information sources to create the intermediate semantic representation. One possible choice is to directly use online textual documents, e.g., those found in Wikipedia, to build such a representation (Elhoseiny et al. [2013]; Ba et al. [2015]). This is promising because online text documents can be easily obtained and contain rich information about the object. To conduct zero-shot learning with textual documents, existing works (Akata et al. [2015]; Fu et al. [2015]) develop various ways to measure the similarity between text and visual features. Our work is also based on this idea. We take a step further, however, to consider one additional important factor: the document representation is much more noisy than the human specified semantic representation and negligence of this fact would inevitably lead to inferior performance. For example, when the bag-of-words model is adopted as the document representation, the occurrence of every word in a document will trigger a signal in one dimension of the document representation. However, it is clear that most words in a document are not directly relevant for identifying the object category. Thus it is necessary to design a noise suppression mechanism to down weight the importance of those less relevant words for zero-shot learning.

This mechanism is closely related to feature selection. However, it is not exactly the same. As will be discussed in the following sections, the solution of our method does not discard the less relevant dimensions of the document representation but only suppress their impact for zero-shot learning.

To this end, we propose a zero-shot learning method which particularly caters for the need for noise suppression. More specifically, we proposed a simple yet effective $l_{2,1}$-norm based objective function which simultaneously suppresses the noisy signal

within text descriptions and learns a function to match the visual and text domains. Furthermore, we develop an efficient optimization algorithm to solve this problem. By conducting experiments on two large scale zero-shot learning evaluation benchmarks, we demonstrate the benefit of the proposed noise suppression mechanism as well as its superior performance over other zero-shot learning methods which also rely on online information sources. In addition, we also conduct an in-depth analysis of the proposed method which provides an insight as to what kinds of information within a document are useful for zero-shot learning.

4.2 Background

Most zero-shot learning approaches rely on human specified attributes. As one of the earliest attempt in zero-shot learning, Lampert et al. [2009] adopted a set of attributes obtained from a psychology study. By learning probabilistic predictors of those attributes, they developed a framework to estimate the posterior of the test class. Later, a number of works has been proposed to improve the way of learning the connection between attributes and object categories. For example, the work in Jayaraman and Grauman [2014] addresses unreliability of attributes by exploring the idea of random forest. The work in Akata et al. [2013] converted the zero-shot learning into a cross-domain matching problem and they proposed to learn a matching function to compare the attribute and the image feature. Built upon this idea, Romera-Paredes and Torr [2015] propose a simpler but more effective objective function to learn the matching function. Zhang and Saligrama Zhang and Saligrama [2015] advocate the benefits of using attribute-attribute relationships, termed semantic similarity, as the intermediate semantic representation and they learn a function to match the image features with the semantic similarity.

To go beyond the human specified attributes, recent works also explore the use of other form of semantic representations which can be easily obtained (Mensink et al. [2014]; Akata et al. [2015]; Frome et al. [2013]; Fu et al. [2015]). For example, the

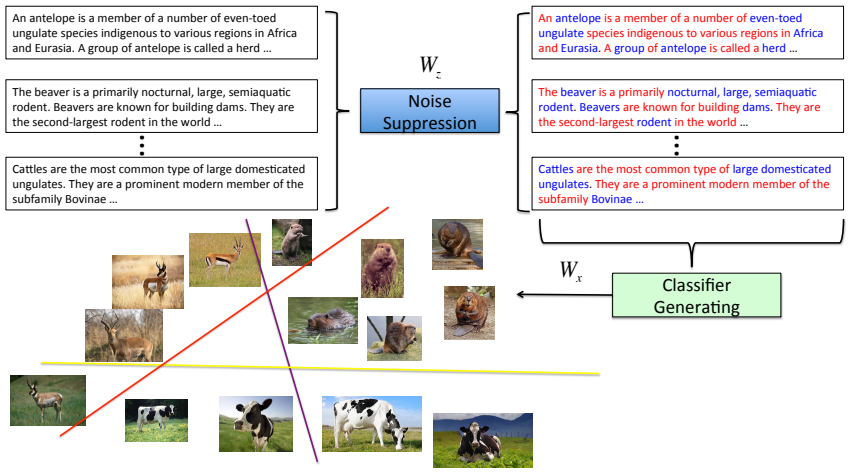

Figure 4.1: Overview of our zero-shot learning approach. The text representations are processed by the noise suppression mechanism to generate a classifier to detect relevant images and the noisy components of text representations are suppressed to gain better performance.

co-occurrence statistics of words has been explored in (Mensink et al. [2014]; Akata et al. [2015]) to capture the semantic relevance of two concepts. The distributed word representation, e.g., word2vec, has been utilized as a substitution of attributes Frome et al. [2013] and more recently the word2vec representation has been shown to be complementary to the human specified attributes Fu et al. [2015].

The other information source for creating the semantic representation is the online textual document, such as Wikipedia articles. In an earlier work, Berg et al. [2010] attempt to discover attribute representation from a noisy web source by ranking the visual-ness scores of attribute candidates. (Rohrbach et al. [2013, 2010]) mine semantic relatedness for attribute-class association from different internet sources. More recent works (Elhoseiny et al. [2013]; Ba et al. [2015]) directly learn a function to measure the compatibility between documents and visual features. However, compared with the state-of-the-art zero-shot learning methods, their performance seems to be disappointing even though some advanced technologies, such as deep learning, has been applied Ba et al. [2015].

4.3 Our approach

4.3.1 Overview

The overview of our method is depicted in Figure 4.1. It starts with a raw document representation which is simply a binarized histogram of words. This representation is fed into our zero-shot learning algorithm to generate a classifier to detect relevant images. In the process of generating this classifier, the noise suppression regularizer in our method will automatically suppress the impact of less relevant words (illustrated as the red words in Figure 4.1).

4.3.2 Text representation

We extract our text representation based on a simple bag-of-words model. We start by a preprocessing step of tokenizing the words and removing stop words and punctuations. Then a histogram of the remaining word occurrences is calculated and is subsequently binarized as the text representation. In other words, once a word appears in a document, its corresponding dimension within the text representation is set to "1". One more commonly used choice for the text representation is based on TF-IDF as in (Elhoseiny et al. [2013]; Ba et al. [2015]). However, we find that it produces worse performance than directly using the binarized representation. Using TF-IDF is about 7% and 5% inferior to binarized representations on AwA and CUB, respectively. This is probably because the weighting calculated of TF-IDF is not suitable for our zero-shot learning although it is considered to be less noisy for applications like document classification. In the binarized histogram we essentially treat each word in a document equally and this inevitably introduces a lot of noisy signals. However, thanks to our noise suppressing zero-shot learning algorithm, we can substantially down-weight the less relevant words and achieve good performance even with a noisy document representation.

4.3.3 Learning to match text and visual features

We first formally define our problem and introduce the notation used in the following sections. At the training stage, both image features and document descriptions for C seen categories are available. Let $\mathbf{X} \in \mathbb{R}^{d \times N}$ denote the image features of N training examples and $\mathbf{Z} \in \{0,1\}^{\hat{d} \times C}$ the aforementioned document representations for C seen classes, where $\hat{d}$ and d are the dimensionality of the document representation and the image features respectively. We also define $\mathbf{Y} \in \{0,1\}^{N \times C}$ as the indicator matrix for the C seen classes. Each row of $\mathbf{Y}$ has a unique "1" indicating its corresponding class label. At the test stage, the document representations of the $\hat{C}$ unseen classes are given and our task is to assign $\hat{C}$ unseen class labels to the test images.

4.3.4 Formulation

Our method is inspired by a recently proposed zero-shot learning approach Romera-Paredes and Torr [2015] which has demonstrated impressive performance despite a very simple learning process. More specifically, it learns a matrix $\mathbf{V}$ which optimizes the following objective function.

$$\min_{\mathbf{V}} \|\mathbf{X}^\top \mathbf{V}\mathbf{S} - \mathbf{Y}\|_F^2 + \lambda \|\mathbf{V}\mathbf{S}\|_F^2 + \gamma \|\mathbf{X}^\top \mathbf{V}\|_F^2 + \lambda\gamma \|\mathbf{V}\|_F^2 \tag{4.1}$$

where $\mathbf{S}$ denotes the semantic attribute matrix and it can be either a binary matrix or a real value matrix. The scalars γ and λ are weights controlling the prominence of the various terms. The underlying idea of this algorithm can be understood as follows. If the task is to classify $\mathbf{X}$ into C categories, we can simply learn a linear classifier by fitting to $\mathbf{Y}$, that is, $\min_{\mathbf{W}} \|\mathbf{X}^\top \mathbf{W} - \mathbf{Y}\|_F^2$. However, in this case $\mathbf{W}$ cannot be transferred to unseen classes. Thus we further impose that $\mathbf{W} = \mathbf{V}\mathbf{S}$. In other words, the classifier of a class is generated from its attributes. With this requirement, the classifier of an unseen class can be easily obtained and utilized to predict the category of a test image. Similarly, we can also treat $\mathbf{X}^\top \mathbf{V}$ as the classifier operated on the attributes $\mathbf{S}$.

The above understanding naturally gives rise to the regularization terms $\lambda\|\mathbf{VS}\|_F^2$ and $\gamma\|\mathbf{X}^\top\mathbf{V}\|_F^2$ which play the same role of the Frobenius norm regularizer as commonly introduced in multi-class classification or regression.

Since our document representation can also be seen as an attribute vector, the method in Romera-Paredes and Torr [2015] can be readily applied to our problem by simply setting $\mathbf{S} = \mathbf{Z}$. However, this naive solution ignores an important fact that the document representation is much more noisy than the human specified attribute vectors. To handle this issue, we introduce a noise suppression mechanism into Eq. (4.1). More specifically, we first decompose $\mathbf{V}$ into two terms:

$$\mathbf{V} = \mathbf{W_x}^\top\mathbf{W_z}, \tag{4.2}$$

where $\mathbf{W_x} \in \mathbb{R}^{m\times d}$ and $\mathbf{W_z} \in \mathbb{R}^{m\times\hat{d}}$. These two matrices will play different roles in our method. $\mathbf{W_z}$ is used to suppress the noisy components of $\mathbf{Z}$ and transform $\mathbf{Z}$ into a $m \times C$ intermediate representation. $\mathbf{W_x}$ is used to generate the image classifier from the noise-suppressed intermediate representation. Thus, two different regularization terms are imposed to suit these two different roles. The first term is the $l_{2,1}$-norm of $\mathbf{W_z}^\top$ which achieves the noise suppression effect. The second term is the Frobenius norm of $\mathbf{W_x}^\top\mathbf{W_z}\mathbf{Z}$ which is similar to the $\lambda\|\mathbf{VS}\|_F^2$ term in Eq. (4.1). The formulation of our method is expressed as follows:

$$\begin{aligned}\min_{\mathbf{W_x},\mathbf{W_z}} \; & L(\mathbf{W_x},\mathbf{W_z}) + \lambda_1\|\mathbf{W_x}^\top\mathbf{W_z}\mathbf{Z}\|_F^2 + \lambda_2\|\mathbf{W_z}^\top\|_{2,1}, \\ & L(\mathbf{W_x},\mathbf{W_z}) = \|\mathbf{X}^\top\mathbf{W_x}^\top\mathbf{W_z}\mathbf{Z} - \mathbf{Y}\|_F^2.\end{aligned} \tag{4.3}$$

The $l_{2,1}$-norm is defined as $\|\mathbf{W_z}^T\|_{2,1} = \sum_{i=1}^{\hat{d}}\|\mathbf{w_z}^i\|_2$, where $\mathbf{w_z}^i$ denotes the i-th column of $\mathbf{W_z}$. It is known that the $l_{2,1}$-norm will encourage the column vectors of $\mathbf{W_z}$ to have few large values, which means that the impact of noisy dimensions of $\mathbf{Z}$ will be substantially suppressed or even completely eliminated. In fact, if λ_2 becomes sufficient large, it achieves the effect of feature selection on the document

representation. However, by cross-validating λ_1 and λ_2, our method does not lead to an exactly sparse solution as it seems that the algorithm prefers to keep the majority of the dimensions in $\mathbf{Z}$ for zero-shot learning. This is probably due to the joint regularization effect of $\|\mathbf{W}_\mathbf{x}^\top \mathbf{W}_\mathbf{z} \mathbf{Z}\|_F^2$ or the fact that dimensions corresponding to lower values of $\|\mathbf{w}_\mathbf{z}^i\|_2$ are still useful for zero-shot learning. Therefore we consider the use of the $l_{2,1}$-norm here as a noise suppression mechanism rather than a feature selection mechanism. We drop out the other regularization terms in Eq. (4.1) since we find them have little impact on performance.

Similar to Romera-Paredes and Torr [2015], once $\mathbf{V}$, in our case $\mathbf{V} = \mathbf{W}_\mathbf{x}{}^\top \mathbf{W}_\mathbf{z}$, is learned, we can infer the class label of a test image $\mathbf{x}$ using the following rule:

$$c^* = \max_c \mathbf{x}^\top \mathbf{W}_\mathbf{x}^\top \mathbf{W}_\mathbf{z} \mathbf{z}_c, \tag{4.4}$$

where $\mathbf{z}_c$ is the document representation for the c-th candidate test class.

4.3.5 Optimization

Eq. (4.3) is convex for $\mathbf{W}_\mathbf{x}$ and $\mathbf{W}_\mathbf{z}$ individually but not convex for both of them. Therefore we can solve it using an alternating method, that is, we first fix $\mathbf{W}_\mathbf{x}$ and solve for $\mathbf{W}_\mathbf{z}$; then fix $\mathbf{W}_\mathbf{z}$ and solve for $\mathbf{W}_\mathbf{x}$.

(1) Fix $\mathbf{W}_\mathbf{x}$ *and solve for* $\mathbf{W}_\mathbf{z}$:

Algorithm 2 Fix $\mathbf{W}_\mathbf{x}$ and solve $\mathbf{W}_\mathbf{z}$

Input: $\mathbf{W}_\mathbf{x}$; $\mathbf{X}$ of seen classes; $\mathbf{Z}$ of seen classes; λ_1 and λ_2; maximum number of iterations τ.
Initialize $\mathbf{D}^0$ as identity matrix $\mathbf{I} \in \mathbb{R}^{\hat{d} \times \hat{d}}$.
for $t = 1 \cdots \tau$ **do**
  · Solve Sylvester equation (4.6) for $\mathbf{W}_\mathbf{z}^t$ with $\mathbf{D}^{t-1}$.
  · Update the diagonal matrix $\mathbf{D}^t$ with its i-diagonal element as $1/(2||(\mathbf{w}_\mathbf{z}^i)^{(t)}||_2)$, where $(\mathbf{w}_\mathbf{z}^i)^{(t)}$ is the i-th column of $\mathbf{W}_\mathbf{z}^t$.
  if *Converge* **then**
    · Break.
  end
end
Output: $\mathbf{W}_\mathbf{z}$.

This sub-problem is a regression problem with $l_{2,1}$-norm regularization. Nie *et al.* Nie et al. [2010] proposes an iterative framework to efficiently solve it. It has been shown that the original problem is equivalent to sequentially solving the following problem until convergence

$$\min_{\mathbf{W_z},\mathbf{D}} L(\mathbf{W_x},\mathbf{W_z}) + \lambda_1 \|\mathbf{W_x}^\top \mathbf{W_z} \mathbf{Z}\|_F^2 + \lambda_2 Tr(\mathbf{W_z} \mathbf{D}^t \mathbf{W_z}^\top), \tag{4.5}$$

where $\mathbf{D}^t$ is a diagonal matrix whose i-th diagonal element is $1/(2\|(\mathbf{w}_z^i)^{(t-1)}\|_2)$ at the t-th iteration, where $(\mathbf{w}_z^i)^{(t-1)}$ is the i-th column of the optimal $\mathbf{W_z}$ solved at the $(t-1)$-th iteration. In practice, we relax $1/(2||\mathbf{w}_\mathbf{z}^i||_2)$ to $1/(2\sqrt{\mathbf{w}_\mathbf{z}^{i\top}\mathbf{w}_\mathbf{z}^i + \sigma})$, $\sigma \to 0$, as the i-th diagonal element to avoid the case of zero columns, and the $l_{2,1}$ norm is therefore approximated by $\sum_{i=1}^{\hat{d}} \sqrt{\mathbf{w}_\mathbf{z}^{i\top}\mathbf{w}_\mathbf{z}^i + \sigma}$. It has been proved in Nie et al. [2010] that this approximation guarantees the convergence and the result approaches to that of $l_{2,1}$-norm as $\sigma \to 0$. The problem in Eq. (4.5) further reduces to a Sylvester equation of $\mathbf{W_z}$

$$\begin{aligned}
&\mathbf{A}\mathbf{W_z} + \mathbf{W_z}\mathbf{B} = \mathbf{C}, \\
&\mathbf{A} = \lambda_2(\mathbf{W_x}\mathbf{X}\mathbf{X}^\top\mathbf{W_x}^\top + \lambda_1\mathbf{W_x}\mathbf{W_x}^\top)^{-1}, \\
&\mathbf{B} = \mathbf{Z}\mathbf{Z}^\top(\mathbf{D})^{-1}, \\
&\mathbf{C} = \frac{1}{\lambda_2}\mathbf{A}\mathbf{W_x}\mathbf{X}\mathbf{Y}\mathbf{Z}^\top(\mathbf{D})^{-1}.
\end{aligned} \tag{4.6}$$

The Sylvester equation has a unique solution if and only if $\mathbf{A}$ and $-\mathbf{B}$ do not share any eigenvalues. Many state-of-the-art toolboxes are able to solve it efficiently. In our setting, since both $\mathbf{A}$ and $\mathbf{B}$ are positive definite, $\mathbf{A}$ has only positive eigenvalues and $-\mathbf{B}$ has only negative eigenvalues. Therefore Eq. (4.6) has a unique solution. In summary, the sub-problem of fixing $\mathbf{W_x}$ to solve $\mathbf{W_z}$ can be solved via the algorithm listed in Algorithm 2.

(2) Fix $\mathbf{W_z}$ *and solve for* $\mathbf{W_x}$:

Algorithm 3 Alternating algorithm for solving Eq. (4.3)

Input: $\mathbf{X}$ of seen classes; $\mathbf{Z}$ of seen classes; λ_1 and λ_2; maximum number of iterations τ.
Initialize $\mathbf{W}_\mathbf{x}^0$ with Gaussian distribution.
for $t = 1 \cdots \tau$ **do**
  · Solve (4.5) iteratively for $\mathbf{W}_\mathbf{z}^t$ with $\mathbf{W}_\mathbf{x}^{t-1}$ according to Algorithm 2.
  · Solve (4.7) for $\mathbf{W}_\mathbf{x}^t$ with $\mathbf{W}_\mathbf{z}^t$.
  if *Converge* **then**
    · Break.
  end
end
Output: $\mathbf{W}_\mathbf{x}$, $\mathbf{W}_\mathbf{z}$.

This sub-problem is a conventional least squares minimization problem which has the following closed-form solution

$$\mathbf{W}_\mathbf{x}^\top = (\mathbf{X}\mathbf{X}^\top + \lambda_1 \mathbf{I})^{-1}\mathbf{X}\mathbf{Y}\mathbf{Z}^\top \mathbf{W}_\mathbf{z}^\top (\mathbf{W}_\mathbf{z}\mathbf{Z}\mathbf{Z}^\top \mathbf{W}_\mathbf{z}^\top)^{-1}. \tag{4.7}$$

By alternating between the above two matrices, the overall alternating optimization algorithm for Eq. (4.3) is listed in Algorithm 3.

4.4 Experiments

We divide our experiments into two parts. In the first part we evaluate the proposed method and compare it against both of the methods utilizing online textual sources and human-specified semantic attributes. In the second part we analyse in-depth the noise suppression effect of the proposed method and provide insight into what kind of information in a document is useful for zero-shot learning.

4.4.1 Experimental setting

Datasets: We test our approach on two widely used benchmarks for attribute learning and zero-shot learning: Animals with Attributes Lampert et al. [2009] (AwA) and Caltech-UCSD birds-200-2011 Wah et al. [2011] (CUB-200-2011). AwA consists of

30,475 images of 50 mammals classes with 85 attributes including color, skin texture, body size, body part, affordance, food source, habitat, and behaviour. CUB-200-2011 contains 11,788 images of 200 categories of bird subspecies with 312 fine-grained attributes such as color/shape/texture of body parts. We follow the train/test split according to Lampert et al. [2009] and Wah et al. [2011], where 10 and 50 testing classes are treated as unseen for AwA and CUB-200-2011, respectively.

Textual document sources: We extract the text representation according to scheme introduced in Section 4.3.2. The raw textual sources are collected from Wikipedia articles describing each of the categories. When constructing the vocabulary, we use the articles of seen classes only. The dimensionality of the text representation is 3506 for AwA and 6815 for CUB-200-2011, respectively.

Image features: To make fair a comparison, two types of image features, the low-level features in Rohrbach et al. [2010] and the fully connected layer activations from the "imagenet-vgg-verydeep-19" Simonyan and Zisserman [2014] CNN are used in our experiments.

Implementation details: The Sylvester equation in Eq. (4.6) is solved by a MATLAB built-in function, which takes only around 5 seconds on an Intel Core i7 CPU at 3.40GHz. The number of rows of matrices $\mathbf{W_x}$ and $\mathbf{W_z}$ is equal to the number of seen classes. We choose the hyper-parameters with a five-fold cross-validation on the seen classes, where 20% (5 for AwA and 30 for CUB-200-2011) of the seen classes are held out for validation and the remaining seen classes are used for training. The hyper-parameters are tuned within the range of all cases of 10^b, where $b = \{-2, -1, \cdots, 5, 6\}$. Once the hyper-parameters are selected, we use all seen classes to train the final model. All of our reported results are averaged over 10 trials.

Method	Top-1 Acc	Top-5 Acc
Ba et al. [2015] (BCE)	1	17.6
Ba et al. [2015] (Hinge)	0.6	18.2
Ba et al. [2015] (Euclidean)	12	42.8
ESZSL Romera-Paredes and Torr [2015]	23.80	59.90
Ours	29.00 ± 0.28	61.76 ± 0.22

Table 4.1: Zero-shot learning classification results on CUB-200-2011, measured by top 1 and top 5 accuracy. 3 different loss functions are used in Ba et al. [2015] for their CNN structure: binary cross entropy (BCE), hinge loss (Hinge), and Euclidean distance (Euclidean). All methods in this table use the same text sources from Wikipedia.

Method	Mean Accuracy
Rohrbach et al. [2010] (Wikipedia)	19.7
Rohrbach et al. [2010] (WordNet)	17.8
Rohrbach et al. [2010] (Yahoo Web)	19.5
Rohrbach et al. [2010] (Yahoo Img)	23.6
Rohrbach et al. [2010] (Flickr Img)	22.9
ESZSL Romera-Paredes and Torr [2015] (Wikipedia)	24.82
Ours (Wikipedia)	29.12 ± 0.07

Table 4.2: Zero-shot learning classification results of AwA, measured by mean accuracy. In Rohrbach et al. [2010], the approach mines attributes names from WordNet and additionally mines class-attribute from online sources of Wikipedia, WordNet, Yahoo, and Flickr. All methods in this table use the same low-level features in Rohrbach et al. [2010].

4.4.2 Performance evaluation

We first compare our method against Ba et al. [2015] and Rohrbach et al. [2010]. The former is most relevant to our work in the sense that it learns a mapping to match images and textual documents. The work in Rohrbach et al. [2010] is a comprehensive comparison study of various information sources for zero-shot learning. Besides these two method, we also treat $\mathbf{S} = \mathbf{Z}$ in Eq. (4.1), and apply the ESZSL method in Romera-Paredes and Torr [2015] to our zero-shot learning problem. To make a fair comparison, we use the same low-level features in Rohrbach et al. [2010] when comparing with it and then use the "imagenet-vgg-verydeep-19" to compare with Ba et al. [2015]. The comparison results are given in Table 4.1 and Table 4.2. As can be

Method/Dataset	AwA	CUB
Rohrbach et al. [2013]	42.7	
Jayaraman and Grauman [2014]	43.01	
Mensink et al. [2014]		14.4
Akata et al. [2013]	43.5	18.0
Lampert et al. [2014] (attr. real)	57.5	
Deng et al. [2014] (hierarchy)	44.2	
ESZSL Romera-Paredes and Torr [2015] (attr. bin)	62.85	
Akata et al. [2015] (Word2Vec)	51.2	28.4
Akata et al. [2015] (GloVe)	58.8	24.2
Akata et al. [2015] (WordNet)	51.2	20.6
Akata et al. [2015] (attr. bin)	52.0	37.8
Akata et al. [2015] (attr. real)	66.7	50.1
Fu et al. [2015] (attr. & words)	66.0	
Zhang and Saligrama [2015] (attr. real)	76.33	30.41
ESZSL Romera-Paredes and Torr [2015] (Wikipedia)	58.53	23.80
Ours (Wikipedia)	66.46 ± 0.42	29.00 ± 0.28

Table 4.3: Zero-shot learning classification results on AwA and CUB-200-2011. Blank spaces indicate these methods are not tested on the corresponding datasets. Contents in braces indicate the semantic sources which these methods use for zero-shot learning. Methods in the upper part of the table use low-level features and the remaining methods in the lower part use deep CNN features.

seen in Table 4.1, the proposed method significantly outperforms the methods in Ba et al. [2015], although they have used a more complicated deep learning framework. Also, we find that our baseline ESZSL achieves good performance. However, it is still 5% inferior to our approach, which clearly demonstrates the advantage of the noise suppression mechanism introduced in this chapter. The results in Table 4.2 further show that our method is superior over other approaches which rely on automatically mined information from the web. Again, our method achieves a significant improvement (more than 4%) over ESZSL.

We now compare our work with a few other state-of-the-art approaches on zero-shot learning, even though some of them are not based on online information sources. The results are summarized in Table 5.2. Results (Rohrbach et al. [2013]; Jayaraman and Grauman [2014]; Mensink et al. [2014]; Akata et al. [2013]) listed in the upper part of the table utilize hand-crafted features and not surprisingly their performance is much inferior to that of the proposed method. The lower part of Table 5.2 are

methods with visual features extracted from a pre-trained CNN and thus are more comparable to our method. In this setting, we find that our method is comparable to most of the state-of-the-art results on AwA and results better than ours are all obtained from the methods using cleaner human defined attributes. The work in Akata et al. [2015] evaluates various semantic representations such as Word2Vec embedding, GloVe word co-occurrence from Wikipedia sources, taxonomy embedding inferred from WordNet Hierarchy, and pre-defined binary and real-valued attributes. Our approach outperforms all methods that use online text sources. This shows that although online text sources provide transferable semantic representations, their discriminative ability is affected by the inherent noise and our method is better at handling the noisy information source for zero-shot learning.

Similar results are observed on the CUB-200-2011 dataset. Our approach again outperforms the methods using online sources and those methods that beat ours are all based on human specified fine-grained attributes. Note that many of the bird categories in CUB-200-2011 have very subtle differences which may not be well captured in Wikipedia articles. However, better performance may be expected by using a higher quality text corpus, such as bird watching articles.

4.4.3 In-depth analysis of the proposed method

In this section we provide an in-depth analysis of the proposed method by examining its noise suppression mechanism and the words that are most discriminative in the view of our method.

4.4.3.1 Effectiveness of the noise suppression method

In our method, the $l_{2,1}$-norm is expected to allow only a few dimensions of the document representation to have large values. The importance of each individual dimension of the document representation can therefore be measured by the l_2-norm of each column of learned $\mathbf{W_Z}$ (we call it the importance weight in the following).

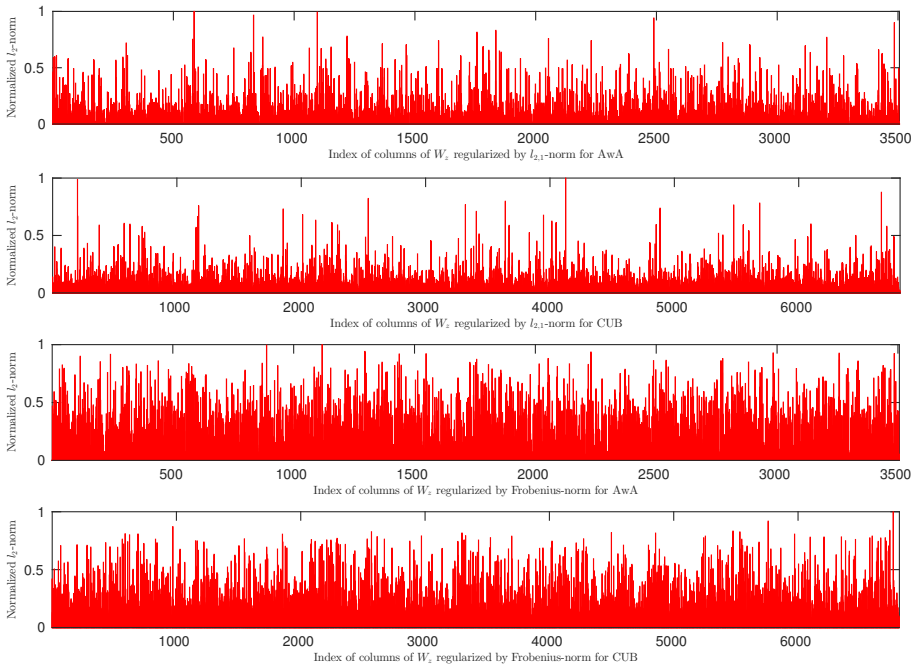

Figure 4.2: The two subfigures at the top show column-wise l_2-norms of $\mathbf{W_z}$ learned with $l_{2,1}$-norm regularization. The two subfigures at the bottom show column-wise l_2-norms of $\mathbf{W_z}$ learned with Frobenius-norm regularization.

We visualize this measurement for each dimension of the document representation in the top two subfigures in Figure 4.2. As can be seen, most of the importance weights are not exactly zero as one might expect given that the $l_{2,1}$-norm is applied. In fact, there are only 702 zero columns (out of 3506) for AwA and 949 (out of 6815) for CUB-200-2011. As also mentioned in Section 4.3, this is probably because of the joint regularization effect of $||\mathbf{W_x^\top W_z Z}||_F^2$ in Eq. (4.3) or because by cross-validation most dimensions are still identified as being useful although their weighting should be very low. The second postulate might be supported by the observation that poorer performance will be obtained if we manually remove the dimensions which have low importance weights.

Although our formulation does not achieve the feature selection effect, it does only assign large importance weights to a small number of dimensions. To visually compare its effect, we replace the $l_{2,1}$-norm and with the Frobenius norm and carry

out our learning algorithm again. The resulting importance weights are shown in the two subfigures at the bottom of Figure 4.2. As can be seen, large importance weights appear in more dimensions in this case. This observation verifies the noise suppression effect of the regularizer introduced in Eq. (4.3) and explains the superior performance of our method over other text-based zero-shot learning approaches.

Seen Class	Top Ranked Words/Dimensions
Antelope	antler, woodland, fight, stomach, spike, antelope, escape, mate, night, variety, ruminant, ridge, broad, scent, herd
Beaver	river, protect, semiaquatic, web, branch, eurasian, american, land, insular, hunt, fur, extant, adult, stream, pond
Blue Whale	ton, whale, flipper, kilometre, marine, ocean, belong, mph, shape, dive, earth, worldwide, indian, travel, pacific
Buffalo	climate, extant, herd, indian, cattle, dairy, animate, bc, trade, behaviour, human, milk, northern, southeast, field
Cow	draft, milk, cattle, widespread, product, meat, domestic, strong, cart, plow, oxen, bullock, cow, animate, india
Deer	antler, fight, mate, elk, palmate, moose, wolf, season, bear, woodland, herd, ruminant, deer, stomach, spike
Moose	herd, elk, palmate, moose, wolf, fight, deer, compete, alces, temperate, climate, aggressive, sedentary, season
Mouse	rodent, house, eat, avoid, burrow, general, genetic, popular, breed, wild, small, tail, vermin, nocturnal, prey
Dolphin	flipper, whale, ton, kilometre, indian, dive, mph, earth, shape, blubber, belong, marine, ocean, capture, prevent
Horse	draft, strong, milk, meat, ungulate, equip, widespread, loose, past, history, compete, endure, technique, style, flee
Hamster	mix, underground, fragile, house, bear, seed, worn, silky, rapid, classify, general, tail, flexible, dwarf, pouch
Killer Whale	ton, whale, dolphin, click, dive, killer, pollution, belong, capture, vocal, calf, tail, threat, fish, fin
Otter	semiaquatic, branch, eurasian, lake, engage, bed, play, trap, river, deplete, giant, cetacean, mink, weasel, web
Rabbit	fragile, house, classify, general, introduce, underground, pad, vegetarian, companionship, defensive, shelf, detect
S. Monkey	agile, arm, walk, tropic, rainforest, primate, source, primary, bark, passage, balance, thumb, moist, threaten
Uneen Class	**Top Ranked Words/Dimensions**
Chimpanzee	agile, finger, primate, arm, walk, human, forest, similar, occasion, blood, move, ape, lowland, hair, ft
Giant Panda	tall, occupy, area, food, white, rare, ft, black, brown, claw, gather, protect, kg, fur, day
Leopard	group, great, world, call, ft, common, typic, predator, individual, show, year, increase, member, red, form
Persian Cat	fur, active, head, kitten, breed, cat, color, carry, england, state, short, north, popular, nose, extreme
Pig	eat, form, plant, species, type, mean, popular, meat, increase, year, ungulate, estimate, remain, large, human
Hippo.	eat, large, hunt, remain, people, kill, water, family, species, skin, size, consider, indian, fat, animate
H. Whale	ton, whale, ocean, shape, kilometre, dive, hour, pacific, feed, indian, baleen, fin, water, dorsal, hunt
Raccoon	tail, bushy, species, long, omnivorous, claw, family, cunning, mammal, skull, solitary, consist, repute, point, part
Rat	rodent, genetic, popular, general, predator, characteristic, species, small, typic, pouch, laboratory, wild, breed
Seal	marine, ocean, sea, spend, water, size, hunt, time, family, fat, large, blubber, seal, flipper, aquatic

Table 4.4: Category-wisely top ranked words, sorted by average importance weights within each class. The blue words are generally considered as meaningful attributes of this class. The green words are concepts somewhat related to this class, but are less informative to define it. The red words are concepts that are not semantically related to the corresponding class.

4.4.3.2 Understanding the important dimensions of the document representation

Since each individual dimension of the textual document representation corresponds to an unique word, we can visualize the dimensions/words with large importance weights for better understanding our zero-shot learning algorithm. Table 4.4 lists at most 15 top scored words for 15 out of 40 seen classes and all unseen classes in AwA and we could make several observations from it: (1) even though the document representations are extremely noisy, most of the top-ranked words are semantically

meaningful to describe discriminative properties of a category (an animal in this case), such as body parts, habitat, behaviour, affordance, taxonomy, and environment. In fact, we find many top weighted words are consistent with some of the human specified attributes in AwA. (2) Many top-ranked words are not explicitly "visualizable" but they imply visual information of a category. For example, the abstract concept "ruminant" implicitly tells that the creature with this property is "deer-like" or "cattle-like" and builds a visual connection between antelope and deer in Table 4.4. This observation has also been made in the literature (Lampert et al. [2009, 2014]; Osherson et al. [1991]; Berg et al. [2010]; Shao et al. [2015]). (3) Interestingly, we also notice that although some concepts are not commonly considered as attributes, they exhibit large importance weight as inferred by our algorithm. By taking a close examination, we categorize these words into two types. The first (labelled green in Table 4.4) are some concepts that are more likely to co-occur with meaningful attributes. For example, the word "stomach" is only shared by antelope and deer in Table 4.4, despite its existence in all mammals. This is probably because "stomach" is more likely to be co-occurred with "ruminant", a discriminative property of ruminant animals. Another type of words (labelled red in Table 4.4) are not sufficiently meaningful for human interpreter. For example, "belong" and "general" are assigned with high importance weight for all cetaceans (blue whale, dolphin, killer whale etc.) and rodents (mouse, rabbit, hamster etc.), respectively. We suspect the reason is due to the dataset bias of documents. For example, documents of similar categories may be edited by authors from the same background who prefer a certain word choice. In sum, we find most of the top ranked words carry weak information by their own, but it seems that using them collaboratively produces impressive discriminative power for zero-shot learning.

4.5 Summary

In this chapter, we have introduced a noise suppression mechanism to text-based zero-shot learning. The proposed $l_{2,1}$-norm based objective function generates classifiers that are robust against textual noise and achieve state-of-the-art zero-shot learning performance. We have made several findings in the experiments. (1) The inherent noise within text sources has a significant impact on zero-shot learning performance. As all the text-methods without noise suppression are inferior to our approach, we speculate that noise in a component of the mid-level representation decreases its discriminative power. (2) Most noisy components are suppressed rather than completely eliminated by our mechanism. Some words, although unimportant individually, can produce meaningful discriminative power when put together. (3) We find three kinds of words in the de-noised representation that can provide useful information for zero-shot learning. The first kind are the attribute-like words that explicitly describe the category. The second are words that are weakly related to the category. They usually occur with definitive words. The last kind of words is non-informative to humans, but shows certain distribution patterns among related categories.

Overall, this chapter points out an important factor in text-based zero-shot learning that has been previously ignored. By dealing directly with the inevitable variations in human expression, and suppressing words that contain little or no value, the performance of text-based automatic zero-shot learning can be significantly improved.

Chapter 5

Zero-shot Learning with Word Vectors

5.1 Introduction

Zero-shot learning (ZSL) aims at recognizing objects of categories that are not available at the training stage. Its basic idea is to transfer visual knowledge learned from seen categories to the unseen categories through the connection made by the semantic embeddings of classes. Attribute Farhadi et al. [2009] is the first kind of semantic embedding utilized for ZSL and remains the best choice for achieving the state-of-the-art performance of ZSL (Akata et al. [2015]; Zhang and Saligrama [2015]). Its good performance, however, is obtained at the cost of extensive human labour to label these attributes.

Recently, several works have explored to use distributed word embeddings (DWE) (Mikolov et al. [2013]; Pennington et al. [2014]) as the alternative to attributes in zero-shot learning (Frome et al. [2013]; Norouzi et al. [2014]). In contrast to human annotated attributes, DWEs are learned from a large-scale text corpus in an unsupervised fashion, which requires little or no human labour to collect. However, the training process of DWEs does not involve visual information and thus they only capture the semantic relationship between different classes. In practice, the semantic similarity does not necessarily correspond to the visual similarity and this visual-semantic discrepancy may lead to the inferior performance of ZSL. In fact, it has been shown that

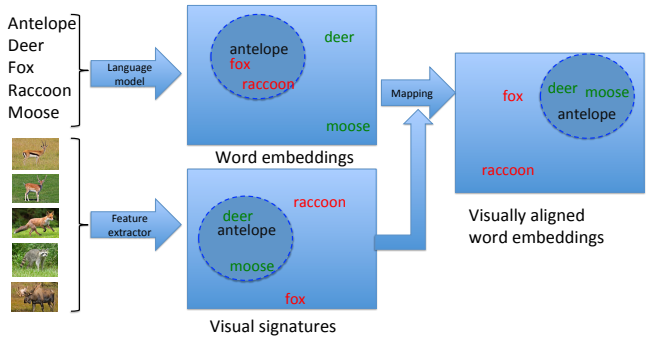

Figure 5.1: The key idea of our approach. Given class names and visual features of the seen classes, we extract the word embeddings from a pre-trained language model and obtain the visual signatures that summarize the appearances of the seen classes. The word embeddings are mapped to a new space where the neighbourhood structure of the mapped embeddings are enforced to be consistent with their visual domain counterparts. During the inference stage, the VAWEs and visual features of seen classes are used to train the ZSL model. Then VAWEs of unseen classes are fed to the trained ZSL model for zero-shot prediction.

when applied to the same ZSL approach, DWE is always outperformed by attribute (Akata et al. [2015]; Changpinyo et al. [2016]; Xian et al. [2016]). To reduce the visual-semantic discrepancy, a popular way in ZSL is to map the semantic embeddings and visual features into a shared space (Frome et al. [2013]; Romera-Paredes and Torr [2015]; Xian et al. [2016]; Zhang and Saligrama [2016]; Long et al. [2016]) to make these two domains comparable. However, when a large visual-semantic discrepancy exists, finding such a mapping can be difficult.

Different to existing work, the method proposed in this chapter directly learns a neural network to map the semantic embedding to a space in which the mapped semantic embeddings preserves a similar neighbourhood structure as their visual counterparts. In other words, we do not require the mapped semantic embeddings to be comparable to visual features but only impose constraints on their structure. This gives more freedom in learning the mapping function, and this could potentially enhance its generalizability. Moreover, since our approach is not tied to a particular zero-shot learning method, the learned mapping can be applied to any zero-shot

learning algorithm.

Three contributions are made in this work. First, we experimentally demonstrate that the inferior ZSL performance of DWE is caused by the discrepancy of visual features and semantic embeddings. Second, to overcome this issue, we propose the visually aligned word embeddings (VAWE) which preserve similar neighbourhood structure with that in the visual domain. Third, we show that VAWE has improved the word embedding based ZSL methods to state-of-the-art performance and is potentially generalizable to any type of ZSL method.

5.2 Background

Zero-shot learning and semantic embedding: Zero-shot learning was firstly made possible by attributes (Lampert et al. [2009]; Farhadi et al. [2009]), which describe the visual appearance of the concept or instance by assigning labelled visual properties to it, and they are easily transferable from seen to unseen classes. Distributed word embeddings, most notably word2vec Mikolov et al. [2013] and GloVe Pennington et al. [2014], are recently explored (Socher et al. [2013]; Frome et al. [2013]; Norouzi et al. [2014]) as a promising alternative semantic embedding towards fully automatic zero-shot learning since their unsupervised training process does not involve any human intervention. ZSL approaches learn a connection between visual and semantic domains either by directly mapping visual features to semantic space (Socher et al. [2013]; Norouzi et al. [2014]; Fu and Sigal [2016]) or projecting both visual and semantic embeddings into a common space (Akata et al. [2013]; Frome et al. [2013]; Romera-Paredes and Torr [2015]; Akata et al. [2015]; Xian et al. [2016]; Zhang and Saligrama [2016]; Long et al. [2016]; Jetley et al. [2015]; Li et al. [2015]). It should be noted that specically in (Long et al. [2016, 2017]), similar issues like visual-semantic ambiguity or visual-semantic structure preservation are proposed and attribute based ZSL methods are designed to deal with them. Although our work shares a common goal with Long et al. [2016] and Long et al. [2017], VAWE

is learned in the semantic domain only which serves as a general tool for any word embedding based ZSL methods. In other words, we are **NOT** proposing a particular ZSL method, and VAWE can be regarded as a meta-method for improving existing ZSL methods.

Word embedding with visual information: As distributed word embedding is limited to pure textual representation, a few works have proposed to improve it with visual cues. Visual word2vec Kottur et al. [2016] is trained by adding abstract scenes to context. In Lazaridou et al. [2015], the language model learns to predict visual representations jointly with the linguistic features. Our work is different from those two works in two aspects: 1) our target is to learn a mapping function which can be generalized to words in unseen classes while the above works try to learn an embedding for the words in the training set. 2) the objective of our method is to encourage a certain neighbourhood structure of the mapped word embedding rather than applying the context prediction objective across visual and semantic domains as in (Kottur et al. [2016]; Lazaridou et al. [2015]).

5.3 Motivation

Assume a set of class labels $\mathcal{W}_s$ and $\mathcal{W}_u$ for images from seen and unseen classes, where $\mathcal{W}_s \cap \mathcal{W}_u = \varnothing$. Most zero-shot learning approaches can be summarised by the general form

$$s(x) = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \; F(x, \phi(y)), \tag{5.1}$$

where $F(x, \phi(y))$ measures compatibility score of the visual feature x and a semantic embedding $\phi(\cdot)$ of class y. During the training phase, where $\mathcal{Y} = \mathcal{W}_s$, $F(\cdot, \cdot)$ is learned to measure the compatibility between x and $\phi(y)$. During the testing phase, the learned $F(x, \phi(y))$ is applied to measure the compatibility between novel classes

$y \in \mathcal{W}_u$ and testing visual samples $x \in \mathcal{X}_{unseen}$ for zero-shot prediction.

This formulation indicates that $\phi(y)$ is an important factor of ZSL. It is desirable that the relationship among $\phi(y)$ retains consistency with the relationship among their visual features, that is, $\phi(y)$ of the visually similar classes remains to be similar and vice versa. Human defined attributes are empirically proven to be qualified $\phi(y)$ because the annotators implicitly infuses the attribute vectors with visual information based on their knowledge and experience of the concepts. In this chapter, we use "concept" and "word" interchangeably to denote category names. However, this is not always the case for semantic embeddings learned from pure text sources, which are trained to maintain the semantic relation of concepts from large text corpora. For example, the concepts "violin" and "piano" are strongly related in the semantic sense even though their appearances are completely different.

To investigate how visual-semantic consistency affects ZSL performance, we conduct a preliminary experiment on AwA dataset. We use the state-of-the-art ESZSL Romera-Paredes and Torr [2015] method to measure the ZSL performance and the average neighbourhood overlap to measure visual-semantic consistency. To calculate the latter, we measure the visual distance between two classes as the average distance between all pairs of visual features within those two classes and this is also equivalent to calculating the distance between their mean feature vectors.That is to say,the visual distance between two classes i and j are

$$D_{i,j} = \|\mathbf{f}_i - \mathbf{f}_j\|_2 \tag{5.2}$$

where $\mathbf{f}_i$ is the mean feature vectors for each class and $\|\cdot\|_2$ is the L-2 norm. Likewise, the semantic distance between two classes can be calculated in the same manner by replacing $\mathbf{f}_i$ and $\mathbf{f}_j$ with the semantic embeddings of classes i and j.

We define $N_v(i, K)$ and $N_s(i, K)$ as the sets that includes the K most similar classes to class i in visual and semantic domains respectively. Then for each class i, we calculate its top-K nearest classes in visual domain using (5.2) and put them in $N_v(i, K)$.

Similarly, we calculate the top-K nearest classes of i in the semantic domain and put them in $N_s(i, K)$.

Four types of semantic embeddings: word2vec, GloVe, binary attribute (presence/absence of the attribute for a class) and continuous attribute (strength of association to a class) are tested. The average neighbourhood overlap is defined in (5.3) as the average number of shared neighbours (out of K=10 nearest neighbours in this case) for all C classes in semantic and visual domains. A value closer to 10 indicates that the embedding is more consistent with the visual domain.

$$Consistancy = \sum_{i=1,\cdots,C} |N_v(i, K) \cap N_s(i, K)|/C \tag{5.3}$$

Method	Embedding	Consistency	Accuracy
ESZSL	word2vec	2.88	58.12
ESZSL	GloVe	2.84	59.72
ESZSL	binary attribute	4.80	62.85
ESZSL	continuous attribute	5.66	75.12
ESZSL	visual feature mean	10.00	86.34

Table 5.1: Preliminary experiment: ZSL accuracies of ESZSL on AwA dataset with different semantic embeddings. The visual feature mean summaries the visual appearance of each seen or unseen class.

The results in Table 5.1 demonstrate that semantic embeddings with more consistent visual-semantic neighbourhood structure clearly produce better ZSL performance. Motivated by that, in this chapter we propose to map the semantic word embedding into a new space in which the neighbourhood structure of the mapped embeddings becomes consistent with their visual domain counterparts. Hereafter, we call the mapped word embedding visually aligned word embedding (VAWE) since the mapped word embedding is re-aligned with visual information in comparison with its unmapped counterpart.

5.4 Approach

Notations: Before elaborating our approach, we formally define the notations as follows. For each visual category i, we denote its semantic word embedding as $\mathbf{s}_i \in \mathbb{R}^{d^s}$ and its visual signature as $\mathbf{v}_i \in \mathbb{R}^{d^v}$, where d^v and d^s are the dimensionality of the visual and semantic space, respectively. The visual signature will be used to define the neighbourhood structure in the visual domain. In the main body of this chapter, we use the mean vector of the visual features in the ith category as its visual signature. Certainly, this is merely one way to define the visual neighbourhood structure, and our method also applies to other alternative definitions.

The to-be-learned mapping function (neural network) is represented by $f_\Theta(\cdot)$, where Θ is the model parameters. For simplicity, we omit the parameter Θ in later notations. This function will be learned on the seen classes and is expected to generalize to unseen classes. In this way, we can apply the VAWE to any zero-shot learning methods. We use the notation $i^* \in \mathcal{W}_u$ and $\mathbf{s}_i^*$ to denote an unseen class and its semantic embedding respectively.

5.4.1 Visually aligned word embedding

To learn $f(\cdot)$, we design an objective function to encourage that $f(\mathbf{s}_i)$ and $\mathbf{v}_i$ share similar neighbours. Specifically, we consider a triplet of classes (a, p, n), where a is more visually similar to p than n. We assume that by examining the consistency of the neighbourhood of class a in the view of its visual signature, the VAWE of the class a and p should be pulled closer while the VAWE of the class a and n should be pushed far part. Hereafter, we call class a, p and n anchor class, positive class and negative class respectively. The training objective is to ensure the distance between $f(\mathbf{s}_a)$ and $f(\mathbf{s}_p)$ is smaller than the distance between $f(\mathbf{s}_a)$ and $f(\mathbf{s}_n)$. Therefore we

employ a triplet hinge loss function:

$$\sum_{\forall(\mathbf{s}_a,\mathbf{s}_p,\mathbf{s}_n)\in\mathcal{T}} \left[\|f(\mathbf{s}_a) - f(\mathbf{s}_p)\|_2^2 - \|f(\mathbf{s}_a) - f(\mathbf{s}_n)\|_2^2 + \alpha\right]_+, \tag{5.4}$$

where $[]_+$ denotes the hinge loss and α is an enforced margin between the distances from anchor class to positive and negative classes. Note that our method does **not** map the semantic word embedding into a shared space with visual feature as in many ZSL methods such as DeViSE Frome et al. [2013]. The mapping function only applies at the semantic domain. We set $\alpha = 1.0$ in all experiments.

5.4.2 Triplet selection

The choice of the triplet (a, p, n) plays a crucial role in our method. Our method encourages the output embedding to share neighbourhood with visual signatures. Therefore if two classes are close in the visual domain, but distant in the semantic domain, their semantic embeddings should be pulled closer, and vice versa. Specifically, for an anchor class $a \in \mathcal{W}_s$, if another class $p \in \mathcal{W}_s$ is within the top-K_1 neighbours $N_v(a, K_1)$ in the view of visual domain but not within the top-K_2 ($K_2 > K_1$) neighbours $N_s(a, K_2)$ in the view of semantic domain, then p should be pulled closer to a and we include p as a positive class. On the other hand, if another class $n \in \mathcal{W}_s$ is within the top-K_1 neighbours of a in semantic view but not within the top-K_2 neighbours in visual view, n should be pushed far away from a and we include n as the negative class. Note that using $K_2 > K_1$ avoids over-sensitive decision on the neighbourhood boundary. In other words, if j is within the top-K_1 neighbourhood of i, it is deemed "close" to i and only if j is not within the top-K_2 neighbourhood of i, it is considered as "distant" from i.

As noted by Dinu and Baroni [2015], nearest neighbour approaches may suffer from the hubness problem in high dimension: some items are similar to all other items and thus become hubs. In our experiment if a positive concept appears in the neighbourhood of many words during training, the VAWE $f(\mathbf{s})$ would concentrate

around this hub and this could be harmful for learning a proper $f(\cdot)$. We design a simple-but-effective hubness correction mechanism as a necessary regularizor for training by removing such hub vectors from the positive class candidates as the training progresses. We calculate the "hubness level" for each concept before each epoch. Concretely, we accumulate the each concept's times of appearances in the neighbourhood of other concepts in the mapped semantic domain $f(\mathbf{s})$. We mark the concepts that appear too often in the neighbourhood of other concepts as hubs and remove them from positive classes in the next training epoch. In our experiment the hubness correction usually brings 2-3% of improvement over the ordinary triplet selection. We summarize the triplet selection process and hub vectors generation in Algorithm 4 and Algorithm 5, respectively.

Algorithm 4 Dynamic triplet selection at epoch $t+1$

Input: Nearest neighbourhood structure sets $N_v(i)$ and $N_s(i)$ in visual and semantic domains for each seen class computed from semantic and visual signatures $(\mathbf{s}_i, \mathbf{v}_i), i \in \mathcal{W}_s$ and K_1 and K_2;

Initialize triplet set $\mathcal{T}_{t+1} = \varnothing$ and hub vector set $H_t = \varnothing$ at epoch $t+1$.

for $i = 1 \cdots |\mathcal{W}_s|$ **do**
- $N_v(i) = N_v(i) - H_t$.
- $a = i$.
- **for** $s \in N_s(i)$ **do**
  - **if** $s \notin N_v(i)$ **then**
    - $n = s$
  - **end**
  - **for** $v \in N_s(i)$ **do**
    - $p = v$.
    - $\mathcal{T}_{t+1} = \mathcal{T}_{t+1} \cup (a, p, n)$.
  - **end**
- **end**

end

- Randomly shuffle the order of triplets in $\mathcal{T}_{t+1}$.

Output: $\mathcal{T}_{t+1}$.

Algorithm 5 Generating hub vector set before epoch $t+1$

Input: output embeddings at epoch t $f_t(s_i), i \in \mathcal{W}_s$; number of neighbours in visual and semantic domains M and K;
Initialize $Hubs \in \mathbb{N}^{|\mathcal{W}_s|}$ as a zero-valued vector which each of its element counting the hubness level of each vector; hub vector set at epoch t $H_t = \varnothing$.
for $i = 1 \cdots |\mathcal{W}_s|$ **do**
  · Get $N(f_t(s_i))[1:K]$.
  for $j = 1 \cdots |\mathcal{W}_s|$ **do**
    if $j \in N(f_t(s_i))[1:K]$ **then**
      · $Hubs_j += 1$.
    end
  end
end
for $i = 1 \cdots |\mathcal{W}_s|$ **do**
  if $Hubs_i > K$ **then**
    · $H_t = H_t \cup i$
  end
end
Output: H_t.

5.4.3 Learning the neural network

We formulate $f(\cdot)$ as a neural network that takes inputs from the pre-trained word embeddings and outputs new visually aligned word embeddings. During the training stage, the training triplets are selected from the seen classes according to Algorithm 4, and parameters of $f(\cdot)$ are adjusted by SGD to minimize the triplet loss (5.5). Note that although the number of training classes is limited, $f(\cdot)$ is trained with the triplets of classes, which amount up to $\mathcal{O}(|\mathcal{W}_s|^3)$. The inference structure of $f(\cdot)$ contains two fully-connected hidden layers with ReLU non-linearity, and the output embedding is L-2 normalized to d'-D unit hypersphere before being propagated to the triplet loss layer.

$$\min_{\Theta} \sum_{\forall(\mathbf{s}_a,\mathbf{s}_p,\mathbf{s}_n)\in\mathcal{T}_t} \left[\|f(\mathbf{s}_a) - f(\mathbf{s}_p)\|_2^2 - \|f(\mathbf{s}_a) - f(\mathbf{s}_n)\|_2^2 + \alpha\right]_+ + \lambda\|\Theta\|_2^2. \tag{5.5}$$

where $\mathcal{T}_t$ is the set of all selected triplets at epoch t.

During the inference stage, $f(\cdot)$ is applied to word embeddings of both seen and

unseen classes. The output VAWEs are off-the-shelf for any zero-shot learning tasks.

5.5 Experiments

In order to conduct a comprehensive evaluation, we train the VAWE from two kinds popular distributed word embeddings: word2vec Mikolov et al. [2013] and GloVe Pennington et al. [2014]. We apply the the trained VAWE to four state-of-the-art methods. We compare the performance against the original word embeddings and other ZSL methods using various semantic embeddings, including attributes, category hierarchy, text documents, and distributed word embeddings.

Datasets: We test the methods on four widely used benchmark dataset for zero-shot learning: aPascal/aYahoo object dataset Farhadi et al. [2009] (aPY), Animals with Attributes Lampert et al. [2009] (AwA), Caltech-UCSD birds-200-2011 Wah et al. [2011] (CUB), and the SUN scene attribute dataset Xiao et al. [2014] (SUN). AwA and aPY are coarse-grained datasets that consist of distinctive animal species or daily life objects. The CUB dataset is a fine-grained dataset that contains 200 categories of bird subspecies. The SUN dataset also contains fine-grained categories of scenes. The train/test split protocols for each dataset are set in accordance with the aforementioned literature.

Distributed word embeddings: We train the VAWE from two pre-trained distributed word emedding models: word2vec and GloVe. We pre-train the word2vec model from scratch on a large combination of text corpus: UMBC Web Base[1], the latest Wikipedia dump [2], and a corpus of English news articles[3]. The resulted model generates 1000-D real valued vectors for each concept. As for GloVe, we use the pre-trained 300-D word embeddings provided by Pennington et al. [2014][4] We only test GloVe on aPY and AwA datasets because the pre-trained GloVe model does not con-

[1]http://ebiquity.umbc.edu/redirect/to/resource/id/351/UMBC-webbase-corpus
[2]http://dumps.wikimedia.org/enwiki/latest/enwiki-latest-pages-articles.xml.bz2
[3]http://www.statmt.org/wmt14/training-monolingual-news-crawl/news.2012.en.shuffled.gz; http://www.statmt.org/wmt14/training-monolingual-news-crawl/news.2013.en.shuffled.gz
[4]http://nlp.stanford.edu/projects/glove/glove.6B.zip

Method	Feature	Embedding	aPY	AwA	CUB	SUN
Lampert et al. [2014]	V	continuous attribute	38.16	57.23		72.00
Deng et al. [2014]	D	class hierarchy		44.2		
Ba et al. [2015]	V	web documents			12.0	
Akata et al. [2015]	V	word2vec		51.2	28.4	
Akata et al. [2015]	V	GloVe		58.8	24.2	
Akata et al. [2015]	V	continuous attribute		66.7	50.1	
Qiao et al. [2016]	V	web documents		66.46	29.00	
Zhang and Saligrama [2015]	V	continuous attribute	46.23	76.33	30.41	82.50
Zhang and Saligrama [2016]	V	continuous attribute	50.35	79.12	41.78	83.83
Long et al. [2016]	L	continuous attribute	39.42	51.75		
SynC	G	continuous attribute		69.7	53.4	62.8
LatEm	G	continuous attribute		72.5	45.6	
ESZSL	V	continuous attribute	24.22	75.32		82.10
ConSE	V	word2vec	21.82	46.80	23.12	43.00
ConSE + Ours	V	VAWE word2vec	35.29	61.24	27.44	63.10
ConSE	V	GloVe	35.17	51.21		
ConSE + Ours	V	VAWE GloVe	42.21	59.26		
SynC	V	word2vec	28.53	56.71	21.54	68.00
SynC + Ours	V	VAWE word2vec	33.23	66.10	21.21	70.80
SynC	V	GloVe	29.92	60.74		
SynC + Ours	V	VAWE GloVe	31.88	64.51		
LatEm	V	word2vec	19.64	50.84	16.52	52.50
LatEm + Ours	V	VAWE word2vec	35.64	61.46	19.12	61.30
LatEm	V	GloVe	27.72	46.12		
LatEm + Ours	V	VAWE GloVe	37.29	55.51		
ESZSL	V	word2vec	28.32	58.12	24.82	64.50
ESZSL + Ours	V	VAWE word2vec	43.23	76.16	24.10	71.20
ESZSL	V	GloVe	34.53	59.72		
ESZSL + Ours	V	VAWE GloVe	44.25	75.10		

Table 5.2: ZSL classification results on 4 datasets. Blank spaces indicate these methods are not tested on the corresponding datasets. Bottom part: methods using VAWE and the original word embeddings as semantic embeddings. Upper part: state-of-the-art methods using various sources of semantic embeddings. Visual features include V:VGG-19; G:GoogLeNet; D:DECAF; L:low-level features.

tain the associated word embeddings for too many fine-grained categories in CUB and SUN. It should be noted that Akata et al. [2015] and Xian et al. [2016] train the word embeddings for CUB by replacing the category names with unique scientific names. Since we assume a more realistic ZSL scenario that the names of unseen class are not known until testing, we do not follow this preprocessing step and simply use

the word embeddings of the original class names.

Image features and visual signatures: For all the four test methods in our experiments, we extract the image features from the fully connected layer activations of the deep CNN VGG-19 Simonyan and Zisserman [2014]. As aforementioned, we use the average VGG-19 features of each seen category as the visual signatures for them.

Test ZSL methods: We apply trained VAWEs on four state-of-the-art methods denoted as ConSE Norouzi et al. [2014], SynC Changpinyo et al. [2016], LatEm Xian et al. [2016] and ESZSL Romera-Paredes and Torr [2015] in the following sections. We use our own implementation of ConSE and ESZSL and the publicly available codes of SynC and LatEm. For ConSE, the hyperparameter T (top-T nearest embeddings as the combination embedding for a test image) is set to 10. For ESZSL, the hyperparameters are tuned by cross-validation as in Romera-Paredes and Torr [2015]. We use the same default parameter settings in their codes for SynC and LatEm.

Implementation details: We stop the training when the triplet loss stops decreasing. This usually takes 150 epochs for aPY, 250 epochs for AwA, 50 epochs for CUB and 20 epochs for SUN. Number of nearest neighbours in visual space is $K_1 = 10$ for all datasets. Number of nearest neighbours in semantic space K_2 is set to half the number of seen classes for each dataset (except for SUN, which has many seen classes), that is 10, 20, 75 and 200 for aPY, AwA, CUB and SUN, respectively. The output dimension is set to d'=128. We report the multi-class accuracies averaged over 10 trials of embedding trainings. We follow the same parameter selection or cross-validation settings as in the literatures of the to-be-tested methods. Note that for some methods we report different results to those in their original papers by using the same type of word embeddings. This is because we train the word embeddings with different text corpus and use different visual features, or the original papers report average per-class accuracies while we report the multi-class accuracies. For all to-be-tested methods, we use the same word embeddings and visual features for a fair comparison. Our unoptimized code based on Tensorflow takes less than 5

minutes to train.

5.5.1 Performance improvement and discussion

In this section, we test the effect of using VAWE trained from word2vec and GloVe in various ZSL methods. The main results of VAWE compared against the original word embeddings are listed in the bottom part of Table 5.2. Except for the fine-grained dataset CUB, the VAWEs trained from both word embeddings gain overall performance improvement on all test methods. Most notably on the coarse-grained datasets, i.e., aPY and AwA, the VAWEs outperform their original counterparts by a very large margin.

For ZSL methods, we find that the performance improvement is most significant for ConSE and ESZSL, partly because these two methods directly learns a linear mapping between visual and semantic embeddings. A set of semantic embeddings that is inconsistent with the visual domain would hurt their performance the most. By using the VAWEs, those methods learn a much better aligned visual-semantic mapping and earns a great performance improvement. In comparison, SynC benefits less from the VAWE than other methods on some datasets. because it learns a new semantic space that can correct some deficiency of original space. Nevertheless, VAWE still helps it to achieve better accuracies.

The performance improvement is limited on fine-grained datasets CUB and SUN. Compared to the coarse-grained class datasets, the difference in categories in CUB and SUN is subtle in both visual and semantic domains. This causes the their visual signatures and semantic embeddings more entangled and results higher hubness level. Therefore it is more challenging to re-align the word embeddings of fine-grained categories by our method.

Overall, the VAWEs exhibit consistent performance gain for various methods on various datasets (improved performance on 22 out of 24 experiments). This observation suggests that VAWE is able to serve as a general tool for improving the perfor-

Method	Dim.	aPY	AwA	CUB	SUN
ConSE + Ours	64	34.14	60.26	25.21	59.00
ConSE + Ours	128	35.29	61.24	27.44	63.10
ConSE + Ours	256	35.46	62.27	26.18	64.70
ConSE + Ours	512	33.13	61.31	23.94	59.40
SynC + ours	64	34.09	64.21	22.31	70.10
SynC + ours	128	33.23	66.10	21.21	70.80
SynC + ours	256	31.92	61.71	20.11	70.40
SynC + ours	512	30.41	59.43	18.54	69.40
LatEm+ Ours	64	35.11	62.17	20.25	61.00
LatEm + Ours	128	35.64	61.46	19.12	61.30
LatEm + Ours	256	34.41	61.52	20.08	60.50
LatEm + Ours	512	34.01	60.01	20.11	57.80
ESZSL + Ours	64	43.14	75.90	23.88	69.50
ESZSL + Ours	128	43.23	76.16	24.10	71.20
ESZSL + Ours	256	42.91	77.21	22.14	70.50
ESZSL + Ours	512	37.07	74.54	19.09	70.90

Table 5.3: ZSL accuracies of four test methods on four datasets, applied with VAWE from word2vec with various output dimensionalities.

mance of ZSL approaches.

5.5.2 Comparison against the state-of-the-art

We also compare the improved results of VAWE against the results of recently published state-of-the-art ZSL methods using various sources of semantic embeddings in the upper part of Table 5.2. It can be observed that methods using VAWE beat all other methods using non-attribute embeddings. Even compared against the best performing attribute-based methods, our results are still very competitive on coarse-grained class datasets: only a small margin lower than Zhang and Saligrama [2016] that uses continuous attributes. The results indicate that VAWE is a potential substitution for human-labelled attributes. The VAWE is not only human labour free but also provides comparable performance to attributes.

5.5.3 Dimensionality of output embeddings

The dimensionality of VAWE is a free parameter in our framework. We investigate its influence by evaluating the ZSL performance of all test methods on all datasets. Due to space limitation, we only report the results of VAWE obtained from word2vec in Table 5.3. It can be observed that higher dimensionality 512 may cause a slight performance drop, especially for SynC and ESZSL. Overall, the VAWE is robust on dimensionality lower than 256.

5.5.4 The effect of visual features

Visual signature source	Low-level	DeCAF	VGG-19
ConSE + Ours	55.57	60.08	61.24
SynC + Ours	59.06	67.30	66.10
LatEm + Ours	58.43	63.33	61.46
ESZSL + Ours	67.28	73.23	76.16

Table 5.4: ZSL accuracies on the AwA dataset of VAWE trained with visual signatures from different feature sources. For the ZSL methods, the VGG-19 features are still used for training and testing.

The learning process of the mapping function relies on the choice of visual features which implicitly affects the neighbourhood structure in the visual domain. In this section, we investigate the impact of the choice of visual features on the quality of the mapped VAWE. Again, the quality of the VAWE is measured by its performance on ZSL. In previous sections, we extracted the visual signature as the mean of the VGG-19 features of each class. Here we further replace it with low-level features or DeCAF features provided by Lampert et al. [2009] and use them to obtain the VAWEs of word2vec. Once the VAWEs is learned we apply the ZSL with VGG-19 features and the experiment is conducted on the AwA dataset. Note that both DeCAF features and low-level features are weaker image features than VGG-19. The experiment results are shown in Table 5.4. From the experimental results, we find that performance of all four ZSL methods do not change too much when we replace

VGG-19 with DeCAF to learn the mapping function. Using low-level features will degrade the performance but comparing to the performance of using the original word2vec the learned VAWE still shows superior performance. These observation suggests that we may use one type of visual features to train the VAWE and apply them to ZSL methods trained with another kind of visual features and still obtain good results.

5.6 Summary

In this chapter, we show that the discrepancy of visual features and semantic embeddings negatively impacts the performance of ZSL approaches. Motivated by that, we propose to learn a neural network with triplet loss to map the word embeddings into a new space in which the neighbourhood structure of the mapped word embedding becomes similar to that in the visual domain. The visually aligned word embeddings boost the ZSL performance to a level that is competitive to human defined attributes. Besides that, our approach is independent of any particular ZSL method. This gives it much more flexibility to generalize to more potential applications of vision-language tasks.

Chapter 6

Conclusion and Future Directions

In this thesis, we have studied how mid-level representations benefit the tasks of action recognition and zero-shot learning in computer vision. In this final chapter, we summarize the key highlights of our work and discuss several future directions for the two tasks.

6.1 Conclusion

In Chapter 3, we describe an effective skeleton-based action approach that achieves high accuracy on the relevant benchmark datasets. The keys to this performance are two factors. We propose trajectorylet, a novel local descriptor that captures static and dynamic information in a short interval of joint trajectories. We also devise a novel framework to generate robust and discriminative mid-level representations for action instances by learning a set of distinctive trajectorylet detectors.

In Chapter 4, we discover an important factor in text-based zero-shot learning that has been previously ignored. By dealing directly with the inevitable variations in human expression, and suppressing words that contain little or no value, the performance of text-based automatic zero-shot learning are significantly improved. The proposed $l_{2,1}$-norm based objective function generates classifiers that are robust against textual noise. We have made an in-depth analysis of important components within a document that are contributing the performance zero-shot learning.

In Chapter 5, we show that the discrepancy of visual features and semantic embeddings causes negative impacts on the performance of ZSL approaches. To over-

come this visual-semantic discrepancy, we here augment the distributed word embedding with visual information by learning a neural network to map it into a new representation called the visually aligned word embedding (VAWE). We further design an objective function to encourage the neighbourhood structure of VAWEs to mirror that in the visual domain. The visually aligned word embeddings boost the ZSL performance to a level that is competitive to human defined attributes.

Overall, this thesis focuses on learning or improving mid-level representations for boosting the performance of action recognition and zero-shot learning. To achieve this goal, these mid-level representations can be learned, de-noised, or re-aligned. As a result, we believe the approaches proposed in this thesis will bring insights on object classification tasks.

6.2 Future Directions

Even though this thesis has made considerable contribution to mid-level representations for the tasks of action recognition and zero-shot learning, we still notice that several open problems remain. We point out future directions for these problems.

6.2.1 Action Recognition

In Chapter 3, we design a discriminative mid-level representation with local trajectorylet for action recognition. One issue remaining is that our model is trained and tested on completed action sequences with global temporal pyramids. The local temporal information is expected to be utilized for real-time action detection before an action is completed. This can be implemented via low-latency sequence prediction models such as RNN.

As noted in the experiments of Chapter 3, the performance of our approach may be affected by noisy sequences caused by occlusions of interacted objects. An intuitive solution is to incorporate other modalities such as RGB videos of human-object interactions. By identifying the interacted objects which are not present in the skele-

ton data, our approach could be modified to become more robust against noisy data.

From the visualized discriminative trajectorylets in Figure 3.10, we observe that semantic information are highly correlated to these patterns. Therefore, we plan to study how semantic information such as attributes can be extracted from action sequences in the sense of attribute discovery. This may potentially benefit zero/few-shot learning tasks for actions.

Since the discriminative trajectorylets are detected from a sequence of action frames, our approach can be naturally generalized to the classification tasks of other sequential data. Notable examples include speech recognition and sentence recognition.

6.2.2 Zero-shot Learning

In Chapter 4 and Chapter 5, we explore the application of two alternative mid-level representations of attribute to the zero-shot learnig tasks. For the online documents, we have made several findings in the experiments regarding the words highest scores. It is interesting to further extract meaningful semantic information from those weighted words. These may help attribute discovery tasks.

We also notice that some higher-scored words are non-informative to human interpretations, but show certain distribution patterns among related categories. The combinations of these words also exhibit discriminative power for defining the categories. We plan to delve into this observation and exploit it for performance improvement of zero-shot learning.

For the distributed word embeddings, except for the fine-grained dataset CUB, the visually aligned word embeddings gain overall performance improvement on all test methods. We suspect the lower performance on the fine-grained dataset is due to the subtle visual and semantic differences among the fine-grained categories. It is worth incorporating some fine-grained information into our framework, for example, the local visual features and larger and more detailed text corpus on fine-grained

concepts.

Since our framework of VAWE is independent of any particular zero-shot learning method, it is able to generalize to other potential applications of vision-language tasks, such as sentence-to-image retrieval Hu et al. [2016] or visual question answering (Wu et al. [2016]; Shih et al. [2016]).

Bibliography

Akata, Z.; Perronnin, F.; Harchaoui, Z.; and Schmid, C., 2013. Label-embedding for attribute-based classification. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 22, 59, 69, and 77)

Akata, Z.; Reed, S.; Walter, D.; Lee, H.; and Schiele, B., 2015. Evaluation of output embeddings for fine-grained image classification. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* IEEE, IEEE. (cited on pages 4, 23, 24, 57, 58, 59, 60, 69, 70, 75, 76, 77, and 86)

Ba, J. L.; Swersky, K.; Fidler, S.; and Salakhutdinov, R., 2015. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *Proc. IEEE Int. Conf. Comp. Vis.* IEEE. (cited on pages xxi, 4, 25, 58, 60, 61, 68, 69, and 86)

Beauchemin, S. S. and Barron, J. L., 1995. The computation of optical flow. *ACM Comput. Surv.*, 27, 3 (Sep. 1995), 433–466. doi:10.1145/212094.212141. http://doi.acm.org/10.1145/212094.212141. (cited on page 10)

Berg, T. L.; Berg, A. C.; and Shih, J., 2010. Automatic attribute discovery and characterization from noisy web data. In *Proc. Eur. Conf. Comp. Vis.* (Heraklion, Crete, Greece, 2010), 663–676. Springer-Verlag, Berlin, Heidelberg. http://dl.acm.org/citation.cfm?id=1886063.1886114. (cited on pages 16, 24, 60, and 73)

Bobick, A. F. and Davis, J. W., 2001. The recognition of human movement using temporal templates. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23, 3 (Mar 2001), 257–267. doi:10.1109/34.910878. (cited on page 10)

Bruderlin, A. and Williams, L., 1995. Motion signal processing. In *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH

'95, 97–104. ACM, New York, NY, USA. doi:10.1145/218380.218421. http://doi.acm.org/10.1145/218380.218421. (cited on page 11)

Chaaraoui, A. A.; Padilla-López, J. R.; Climent-Pérez, P.; and Flórez-Revuelta, F., 2014. Evolutionary joint selection to improve human action recognition with RGB-D devices. *Expert Syst. Appl.*, 41, 3 (February 2014), 786–794. (cited on pages 33 and 43)

Changpinyo, S.; Chao, W.; Gong, B.; and Sha, F., 2016. Synthesized classifiers for zero-shot learning. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 23, 76, and 87)

Chaudhry, R.; Ofli, F.; Kurillo, G.; Bajcsy, R.; and Vidal, R., 2013. Bio-inspired dynamic 3D discriminative skeletal features for human action recognition. In *Proc. Workshops of IEEE Conf. Comp. Vis. Patt. Recogn.*, 471–478. (cited on pages 12, 33, and 45)

Chen, C.-Y. and Grauman, K., 2014. Inferring analogous attributes. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 17)

Chen, L.; Zhang, Q.; and Li, B., 2014. Predicting multiple attributes via relative multi-task learning. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 19)

Deng, J.; Ding, N.; Jia, Y.; Frome, A.; Murphy, K.; Bengio, S.; Li, Y.; Neven, H.; and Adam, H., 2014. Large-scale object classification using label relation graphs. In *Proc. Eur. Conf. Comp. Vis.*, 48–64. Springer International Publishing. (cited on pages 69 and 86)

Dinu, G. and Baroni, M., 2015. Improving zero-shot learning by mitigating the hubness problem. In *Proc. Int. Conf. Learning Representations*. http://arxiv.org/abs/1412.6568. (cited on page 82)

Du, Y.; Wang, W.; and Wang, L., 2015. Hierarchical recurrent neural network for

skeleton based action recognition. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 13, 33, and 43)

Elgammal, A.; Harwood, D.; and Davis, L., 2000. Non-parametric model for background subtraction. In *Proc. Eur. Conf. Comp. Vis.*, vol. 2. (cited on page 10)

Elhoseiny, M.; Saleh, B.; and Elgammal, A., 2013. Write a classifier: Zero-shot learning using purely textual descriptions. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2584–2591. IEEE, Washington, DC, USA. (cited on pages 4, 25, 58, 60, and 61)

Ellis, C.; Masood, S. Z.; Tappen, M. F.; Laviola, J. J., Jr.; and Sukthankar, R., 2013. Exploring the trade-off between accuracy and observational latency in action recognition. *Int. J. Computer Vision*, 101, 3 (2013), 420–436. (cited on pages 13, 31, 33, and 45)

Escorcia, V.; Niebles, J. C.; and Ghanem, B., 2015. On the relationship between visual attributes and convolutional networks. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 1256–1264. (cited on page 2)

Evangelidis, G.; Singh, G.; and Horaud, R., 2014. Skeletal quads: Human action recognition using joint quadruples. In *Int. Conf. Patt. Recogn.* (cited on pages 43 and 49)

Fan, R.-E.; Chang, K.-W.; Hsieh, C.-J.; Wang, X.-R.; and Lin, C.-J., 2008. Liblinear: A library for large linear classification. *J. Machine Learning Research*, 9 (2008), 1871–1874. (cited on page 42)

Farhadi, A.; Endres, I.; Hoiem, D.; and Forsyth, D., 2009. Describing objects by their attributes. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages xvii, 1, 2, 14, 57, 75, 77, and 85)

Feng, J.; Jegelka, S.; Yan, S.; and Darrell, T., 2014. Learning scalable discriminative dictionary with sample relatedness. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 1645–1652. (cited on page 16)

Ferrari, V. and Zisserman, A., 2008. Learning visual attributes. In *Proc. Adv. Neural Inf. Process. Syst.* (cited on pages 1, 14, and 57)

Fothergill, S.; Mentis, H. M.; Kohli, P.; and Nowozin, S., 2012. Instructing people for training gestural interactive systems. In *Proc. SIGCHI Conf. Human Factors in Computing Systems*, 1737–1746. ACM. (cited on page 47)

Frome, A.; Corrado, G.; Shlens, J.; Bengio, S.; Dean, J.; Ranzato, M.; and Mikolov, T., 2013. Devise: A deep visual-semantic embedding model. In *Proc. Adv. Neural Inf. Process. Syst.* (cited on pages 5, 27, 58, 59, 60, 75, 76, 77, and 82)

Fu, Y. and Sigal, L., 2016. Semi-supervised vocabulary-informed learning. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 28 and 77)

Fu, Z.; Xiang, T.; Kodirov, E.; and Gong, S., 2015. Zero-shot object recognition by semantic manifold distance. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 4, 57, 58, 59, 60, and 69)

Gowayyed, M. A.; Torki, M.; Hussein, M. E.; and El-Saban, M., 2013. Histogram of oriented displacements (hod): describing trajectories of human joints for action recognition. In *Proc. Int. Joint Conf. Artificial Intelligence*. (cited on pages 3, 12, 29, 32, 43, and 49)

Han, J.; Shao, L.; Xu, D.; and Shotton, J., 2013. Enhanced computer vision with microsoft kinect sensor: A review. *IEEE T. Cybernetics*, 43, 5 (2013), 1318–1334. (cited on pages 3 and 29)

Hu, R.; Xu, H.; Rohrbach, M.; Feng, J.; Saenko, K.; and Darrell, T., 2016. Natural language object retrieval. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 96)

Hussein, M.; Torki, M.; Gowayyed, M.; and El-Saban, M., 2013. Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations. In *Proc. Int. Joint Conf. Artificial Intelligence*. (cited on pages 43, 47, and 49)

Jayaraman, D. and Grauman, K., 2014. Zero-shot recognition with unreliable attributes. In *Proc. Adv. Neural Inf. Process. Syst.* (Eds. Z. Ghahramani; M. Welling; C. Cortes; N. Lawrence; and K. Weinberger), 3464–3472. Curran Associates, Inc. (cited on pages 22, 59, and 69)

Jegou, H.; Douze, M.; Schmid, C.; and Perez, P., 2010. Aggregating local descriptors into a compact image representation. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 3304–3311. (cited on pages 42, 54, and 55)

Jetley, S.; Romera-Paredes, B.; Jayasumana, S.; and Torr, P. H. S., 2015. Prototypical priors: From improving classification to zero-shot learning. In *Proc. British Mach. Vis. Conf.* (cited on page 77)

Kottur, S.; Vedantam, R.; Moura, J. M. F.; and Parikh, D., 2016. Visual word2vec (vis-w2v): Learning visually grounded word embeddings using abstract scenes. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 78)

Lampert, C. H.; Nickisch, H.; and Harmeling, S., 2009. Learning to detect unseen object classes by betweenclass attribute transfer. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 2, 15, 21, 57, 59, 66, 67, 73, 77, 85, and 90)

Lampert, C. H.; Nickisch, H.; and Harmeling, S., 2014. Attribute-based classification for zero-shot visual object categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36, 3 (2014), 453–465. doi:http://doi.ieeecomputersociety.org/10.1109/TPAMI.2013.140. (cited on pages 69, 73, and 86)

Lazaridou, A.; Pham, N. T.; and Baroni, M., 2015. Combining language and vision with a multimodal skip-gram model. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 153–163. Association for Computational Linguistics, Denver, Colorado. http://www.aclweb.org/anthology/N15-1016. (cited on page 78)

Li, W.; Zhang, Z.; and Liu, Z., 2010. Action recognition based on a bag of 3D points.

In *Proc. Workshops of IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 11, 43, and 44)

Li, X.; Guo, Y.; and Schuurmans, D., 2015. Semi-supervised zero-shot classification with label representation learning. In *Proc. IEEE Int. Conf. Comp. Vis.*, 4211–4219. (cited on page 77)

Liang, L. and Grauman, K., 2014. Beyond comparing image pairs: Setwise active learning for relative attributes. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 208–215. doi:10.1109/CVPR.2014.34. http://dx.doi.org/10.1109/CVPR.2014.34. (cited on page 19)

Liu, L.; Wang, L.; and Liu, X., 2011. In defense of soft-assignment coding. In *Proc. IEEE Int. Conf. Comp. Vis.*, 2486–2493. (cited on pages 42, 54, and 55)

Long, Y.; Liu, L.; and Shao, L., 2016. Attribute embedding with visual-semantic ambiguity removal for zero-shot learning. In *Proc. British Mach. Vis. Conf.* (cited on pages 76, 77, and 86)

Long, Y.; Liu, L.; Shao, L.; Shen, F.; Ding, G.; and Han, J., 2017. From zero-shot learning to conventional supervised classification: Unseen visual data synthesis. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 77)

Malisiewicz, T.; Gupta, A.; and Efros, A. A., 2011. Ensemble of exemplar SVMs for object detection and beyond. In *Proc. IEEE Int. Conf. Comp. Vis.* (cited on pages 31 and 37)

Martens, J. and Sutskever, I., 2011. Learning recurrent neural networks with hessian-free optimization. In *Proc. Int. Conf. Mach. Learn.*, 1033–1040. ACM, New York, NY, USA. (cited on page 45)

Mensink, T.; Gavves, E.; and Snoek, C. G., 2014. Costa: Co-occurrence statistics for zero-shot classification. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 59, 60, and 69)

Messing, R.; Pal, C.; and Kautz, H., 2009. Activity recognition using the velocity histories of tracked keypoints. In *Proc. IEEE Int. Conf. Comp. Vis.*, 104–111. (cited on page 32)

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J., 2013. Distributed representations of words and phrases and their compositionality. In *Proc. Adv. Neural Inf. Process. Syst.*, 3111–3119. http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality. (cited on pages 5, 26, 75, 77, and 85)

Moeslund, T. B.; Hilton, A.; and Krüger, V., 2006. A survey of advances in vision-based human motion capture and analysis. *Comput. Vis. Image Underst.*, 104, 2 (2006), 90–126. (cited on page 3)

Moussa, M.; Hemayed, E.; Nemr, H.; and Fayek, M., 2017. Human action recognition utilizing variations in skeleton dimensions. In *Arabian Journal for Science and Engineering*. (cited on page 13)

Müller, M. and Röder, T., 2006. Motion templates for automatic classification and retrieval of motion capture data. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 137–146. (cited on pages 45 and 47)

Müller, M.; Röder, T.; Clausen, M.; Eberhardt, B.; Krüger, B.; and Weber, A., 2007. Documentation mocap database hdm05. Technical report, Universität Bonn. (cited on page 48)

Nie, F.; Huang, H.; Cai, X.; and Ding, C. H. Q., 2010. Efficient and robust feature selection via joint $l_{2,1}$-norms minimization. In *Proc. Adv. Neural Inf. Process. Syst.*, 1813–1821. http://papers.nips.cc/paper/3988-efficient-and-robust-feature-selection-via-joint-l21-norms-minimization. (cited on page 65)

Norouzi, M.; Mikolov, T.; Bengio, S.; Singer, Y.; Shlens, J.; Frome, A.; Corrado,

G.; and Dean, J., 2014. Zero-shot learning by convex combination of semantic embeddings. In *Proc. Int. Conf. Learning Representations*. (cited on pages 5, 27, 58, 75, 77, and 87)

Ofli, F.; Chaudhry, R.; Kurillo, G.; Vidal, R.; and Bajcsy, R., 2012. Sequence of the most informative joints (SMIJ): A new representation for human skeletal action recognition. In *Proc. Workshops of IEEE Conf. Comp. Vis. Patt. Recogn.*, 8–13. (cited on pages 12, 33, 48, and 49)

Ohn-bar, E. and Trivedi, M. M., 2013. Joint angles similiarities and HOG 2 for action recognition. In *Proc. Workshops of IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 12, 32, and 45)

Oreifej, O. and Liu, Z., 2013. HON4D: Histogram of oriented 4d normals for activity recognition from depth sequences. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 45)

Osherson, D. N.; Stern, J.; Wilkie, O.; Stob, M.; and Smith, E. E., 1991. Default probability. *Cognitive Science*, 15, 2 (1991), 251–269. doi:10.1207/s15516709cog1502_3. http://dx.doi.org/10.1207/s15516709cog1502_3. (cited on page 73)

Parikh, D. and Grauman, K., 2011. Relative attributes. In *Proc. IEEE Int. Conf. Comp. Vis.*, 503–510. IEEE. (cited on pages 18 and 19)

Pennington, J.; Socher, R.; and Manning, C. D., 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. http://www.aclweb.org/anthology/D14-1162. (cited on pages 5, 26, 75, 77, and 85)

Qiao, R.; Liu, L.; Shen, C.; and van den Hengel, A., 2016. Less is more: Zero-shot learning from online textual documents with noise suppression. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 86)

Raptis, M. and Sigal, L., 2013. Poselet key-framing: A model for human activity recognition. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 2650–2657. IEEE Computer Society, Washington, DC, USA. (cited on pages 13 and 33)

Rastegari, M.; Diba, A.; Parikh, D.; and Farhadi, A., 2013. Multi-attribute queries: To merge or not to merge? In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 3310–3317. (cited on page 15)

Rastegari, M.; Farhadi, A.; and Forsyth, D., 2012. Attribute discovery via predictable discriminative binary codes. In *Proc. Eur. Conf. Comp. Vis.* (Florence, Italy, 2012), 876–889. Springer-Verlag, Berlin, Heidelberg. doi:10.1007/978-3-642-33783-3_63. http://dx.doi.org/10.1007/978-3-642-33783-3_63. (cited on pages 1, 2, and 16)

Reed, S.; Akata, Z.; Schiele, B.; and Lee, H., 2016. Learning deep representations of fine-grained visual descriptions. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 25)

Rohrbach, M.; Ebert, S.; and Schiele, B., 2013. Transfer learning in a transductive setting. In *Proc. Adv. Neural Inf. Process. Syst.* (cited on pages 60 and 69)

Rohrbach, M.; Stark, M.; Szarvas, G.; Gurevych, I.; and Schiele, B., 2010. What helps where - and why? semantic relatedness for knowledge transfer. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 910–917. doi:10.1109/CVPR.2010.5540121. http://dx.doi.org/10.1109/CVPR.2010.5540121. (cited on pages xxi, 22, 60, 67, and 68)

Romera-Paredes, B. and Torr, P. H., 2015. An embarrassingly simple approach to zero-shot learning. *Proc. Int. Conf. Mach. Learn.*, (2015). (cited on pages 23, 59, 62, 63, 64, 68, 69, 76, 77, 79, and 87)

Sethi, I. K. and Jain, R., 1987. Finding trajectories of feature points in a monocular image sequence. *IEEE Trans. Pattern Anal. Mach. Intell.*, PAMI-9, 1 (Jan 1987), 56–73. doi:10.1109/TPAMI.1987.4767872. (cited on page 10)

Shankar, S.; Garg, V. K.; and Cipolla, R., 2015. Deep-carving: Discovering visual attributes by carving deep neural nets. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 17)

Shao, J.; Kang, K.; Loy, C. C.; and Wang, X., 2015. Deeply learned attributes for crowded scene understanding. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 73)

Shao, Z. and Li, Y., 2013. A new descriptor for multiple 3D motion trajectories recognition. In *Proc. IEEE Int. Conf. Robotics and Automation*, 4749–4754. (cited on page 29)

Shih, K. J.; Singh, S.; and Hoiem, D., 2016. Where to look: Focus regions for visual question answering. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 96)

Shotton, J.; Fitzgibbon, A.; Cook, M.; Sharp, T.; Finocchio, M.; Moore, R.; Kipman, A.; and Blake, A., 2011. Real-time human pose recognition in parts from single depth images. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 1297–1304. IEEE Computer Society, Washington, DC, USA. (cited on pages 3, 11, and 29)

Shrivastava, A.; Singh, S.; and Gupta, A., 2012. Constrained semi-supervised learning using attributes and comparative attributes. In *Proc. Eur. Conf. Comp. Vis.* (cited on page 18)

Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. In *Proc. Int. Conf. Learning Representations*. (cited on pages 67 and 87)

Socher, R.; Ganjoo, M.; Manning, C. D.; and Ng, A., 2013. Zero-shot learning through cross-modal transfer. In *Proc. Adv. Neural Inf. Process. Syst.* (Eds. C. Burges; L. Bottou; M. Welling; Z. Ghahramani; and K. Weinberger), 935–943. Curran Associates, Inc. http://papers.nips.cc/paper/

5027-zero-shot-learning-through-cross-modal-transfer.pdf. (cited on pages 27, 28, 58, and 77)

Torresani, L.; Szummer, M.; and Fitzgibbon, A., 2010. Efficient object category recognition using classemes. In *Proc. Eur. Conf. Comp. Vis.*, 776–789. (cited on pages 2, 14, and 57)

Usunier, N.; Buffoni, D.; and Gallinari, P., 2009. Ranking with ordered weighted pairwise classification. In *Proc. Int. Conf. Mach. Learn.* (Montreal, Quebec, Canada, 2009), 1057–1064. ACM, New York, NY, USA. doi:10.1145/1553374.1553509. http://doi.acm.org/10.1145/1553374.1553509. (cited on page 23)

Vedaldi, A.; Mahendran, S.; Tsogkas, S.; Maji, S.; Girshick, B.; Kannala, J.; Rahtu, E.; Kokkinos, I.; Blaschko, M. B.; Weiss, D.; Taskar, B.; Simonyan, K.; Saphra, N.; and Mohamed, S., 2014. Understanding objects in detail with fine-grained attributes. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 15)

Vemulapalli, R.; Arrate, F.; and Chellappa, R., 2014. Human action recognition by representing 3D skeletons as points in a Lie group. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 588–595. (cited on pages 3, 12, 29, 32, 43, and 45)

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S., 2011. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology. (cited on pages 66, 67, and 85)

Wang, C.; Wang, Y.; and Yuille, A., 2013. An approach to pose-based action recognition. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 915–922. (cited on pages 33 and 45)

Wang, H.; Kläser, A.; Schmid, C.; and Liu, C.-L., 2011. Action recognition by dense trajectories. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 3169–3176. (cited on page 32)

Wang, J.; Liu, Z.; Wu, Y.; and Yuan, J., 2012. Mining actionlet ensemble for action recognition with depth cameras. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 1290–1297. (cited on pages 12, 33, 44, 45, 46, 47, and 50)

Wang, J.; Yang, J.; Yu, K.; Lv, F.; Huang, T.; and Gong, Y., 2010. Locality-constrained linear coding for image classification. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 3360–3367. (cited on pages 42, 54, and 55)

Wu, D. and Shao, L., 2014a. Deep dynamic neural networks for gesture segmentation and recognition. In *Proc. Workshops of Eur. Conf. Comp. Vis.*, 552–571. Springer. (cited on pages 12 and 32)

Wu, D. and Shao, L., 2014b. Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.*, 724–731. (cited on pages 3, 12, 29, 32, and 45)

Wu, Q.; Wang, P.; Shen, C.; Dick, A.; and van den Hengel, A., 2016. Ask me anything: Free-form visual question answering based on knowledge from external sources. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 96)

Wu, S.; Li, Y.; and Zhang, J., 2008. A hierarchical motion trajectory signature descriptor. In *Proc. IEEE Int. Conf. Robotics and Automation*, 3070–3075. (cited on page 29)

Xia, L.; Chen, C.; and Aggarwal, J., 2012. View invariant human action recognition using histograms of 3D joints. In *Proc. Workshops of IEEE Conf. Comp. Vis. Patt. Recogn.*, 20–27. IEEE. (cited on pages 12, 32, 43, and 49)

Xian, Y.; Akata, Z.; Sharma, G.; Nguyen, Q.; Hein, M.; and Schiele, B., 2016. Latent embeddings for zero-shot classification. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on pages 23, 76, 77, 86, and 87)

Xiao, J.; Ehinger, K. A.; Hays, J.; Torralba, A.; and Oliva, A., 2014. Sun database:

Exploring a large collection of scene categories. *Int. J. Comp. Vis.*, (2014). (cited on page 85)

Yang, X. and Tian, Y., 2012. Eigenjoints-based action recognition using naÃŕve-bayes-nearest-neighbor. In *Proc. Workshops of IEEE Conf. Comp. Vis. Patt. Recogn.*, 14–19. (cited on pages 13, 30, 34, and 43)

Yao, B.; Jiang, X.; Khosla, A.; Lin, A. L.; Guibas, L. J.; and Fei-Fei, L., 2011. Action recognition by learning bases of action attributes and parts. In *Proc. IEEE Int. Conf. Comp. Vis.* Barcelona, Spain. (cited on page 57)

Yu, F.; Cao, L.; Feris, R.; Smith, J.; and Chang, S.-F., 2013. Designing category-level attributes for discriminative visual recognition. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* Portland, OR. (cited on pages 1, 2, and 17)

Zanfir, M.; Leordeanu, M.; and Sminchisescu, C., 2013. The "moving pose": An efficient 3D kinematics descriptor for low-latency action recognition and detection. In *Proc. IEEE Int. Conf. Comp. Vis.* (cited on pages 13, 30, 31, 33, 34, 42, 43, 44, 45, 46, 47, 51, and 54)

Zhang, Z. and Saligrama, V., 2015. Zero-shot learning via semantic similarity embedding. In *Proc. IEEE Int. Conf. Comp. Vis.* IEEE. (cited on pages 2, 23, 57, 59, 69, 75, and 86)

Zhang, Z. and Saligrama, V., 2016. Zero-shot learning via joint latent similarity embedding. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* IEEE. (cited on pages 24, 76, 77, 86, and 89)

Zhao, Z. and Elgammal, A., 2008. Information theoretic key frame selection for action recognition. In *Proc. British Mach. Vis. Conf.* (cited on pages 13 and 33)

Zheng, S.; Cheng, M.-M.; Warrell, J.; Sturgess, P.; Vineet, V.; Rother, C.; and Torr, P., 2014. Dense semantic image segmentation with objects and attributes. In *Proc. IEEE Conf. Comp. Vis. Patt. Recogn.* (cited on page 15)