Visual Understanding by Learning from Multiple Data Aspects

XIONG, Yuanjun

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Doctor of Philosophy
in
Information Engineering

The Chinese University of Hong Kong
July 2016

Abstract of thesis entitled:

Visual Understanding by Learning from Multiple Data Aspects

Submitted by XIONG, Yuanjun

for the degree of Doctor of Philosophy

at The Chinese University of Hong Kong in July 2016

Visual understanding lies in the heart of visual perception. To understand visual signals, a human vision system must learn to process perceived information from multiple aspect. For example, a complex real life event may involve several objects and human object interactions. Seeing the image from one aspect and make prediction is obvious suboptimal. This imposes challenge to current single view based methods. In this thesis work, we emphasize the idea that computer vision system should combine multiple aspects of data in the learning and prediction processes. The resultant approaches have led to superior performances in several high-level visual understanding tasks driven by curated data.

In the first part of this thesis work, we propose a multi-channel deep neural networks architecture to tackle the problem of event recognition from still images. The model is devised to

unify both appearance and spatial configuration information of event images in an end-to-end manner of learning. The learned model performs well in capturing both the visual appearances and human-object interaction and combining them to predict the underlying event categories. This part of work has been published in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015.*

In the second part of this thesis work, a new image tagging framework featuring two complementary techniques: Scaled View Integration and Contrastive Bundled Loss, is proposed. These two techniques effectively combines the information scattered in different locations and scales of the input images and use them to help the classifier learning the underlying visual concepts contrastively. Improvement of performance on several dataset is observed to corroborate the effectiveness of out method.

In the third part of this thesis work, we propose a framework called temporal segment networks (TSN) to deal with the problem of recognizing human activities from videos. The framework aims to combine appearances, shot-term motions, and long-term temporal structures. A unified deep neural networks model is designed under the framework to learn the activity representation from these multiple aspect of video data. The framework combines a sparse temporal sampling strategy and video-level supervision to enable efficient and effective learning using the whole action video. It is demonstrated that this strategy leads

to superior recognition performance and better activity representations. This part of work has been accepted to *European Conference on Computer Vision (ECCV) 2016*

摘要

視覺理解一直是計算機視覺研究中的核心內容。為了理解視覺信息，人類的視覺系統通常從多個方面去處理感受到的視覺信號。以事件識別為例：真實生活中的複雜事件，通常包含了數個物體以及人與物體之間的互動關係。此時，只從一個方面去觀察圖象顯然是不夠的，這就為目前計算機視覺研究中常用的基於單個視角的方法帶來了挑戰。在本論文中，對於計算機視覺系統，我們强調在其學習和推斷的過程中，需要對輸入數據不同方面的信息進行綜合。基於此思想我們在數個基於數據驅動的高層次視覺理解任務中，提出了新的模型框架比得到了較好的性能表現。

本論文的第一部分中，我們提出了一個多通道的深度神經網絡結構來解決從靜態圖象中進行事件識別的問題。該模型可以同時聯合形狀和物體空間關係的信息以實現端到端的事件識別系統的學習。學習完成後的系統可以很好地同時捕捉形狀信息以及物體與物體，物體與人之間的相互作用的信息，以用於事件識別。這部分工作已於IEEE計算機視覺和模式識別會議（CVPR）2015發表。

本論文的第二部分中，我們提出了一個新的圖象自動標籤框架。該框架包含兩項新的技術：多尺度視角綜合，以及對比

式集束學習。通過這兩個技術，我們可以有效地綜合散落在圖片上各個區域以及不同尺度的信息，並通過不同標籤類別之前的對比，提高學習的效率和準確性。在多個數據集的測試中，本框架都得到了優秀的表現，證明了我們的方法的優越性。

在本論文的第三部分中，我們提出了分段式網絡模型來解決在視頻中進行人類動作識別的問題。該框架致力於聯合的信息包括形狀、短期的運動以及長期的時序關係。一個統一的卷積神經網絡被設計用來從這些信息中學習動作的表示。這個框架結合了新的稀疏時間采樣策略和全視頻監督方法來快速有效地從整個動作視頻中進行學習。在實驗中，該模型被證實可以帶來較大的性能提升。這部分工作于歐洲計算機視覺會議（ECCV）2016發表。

Acknowledgement

The four years PhD life in the Chinese University of Hong Kong, and the Multimedia Laboratory, has been an fruitful experience for me, in both academic and personal aspects. Here I would like to give my greatest gratitudes and thanks to those who had helped me along the journey, and made this thesis work possible.

First of all, I would like to thank my supervisor Prof. Xiaoou Tang. He is a great supervisor. He supports me with the resources for my research works without conservation and is always there to share my concerns and findings. More importantly, his insightful suggestions and innovative mind have always guided me in conducting cutting edge research for computer vision. His emphasizes on an atmosphere of sharing and discussion, together with a healthy life style, are also the key factors to the success of the Multimedia Laboratory and my own PhD career.

Secondly I would like to thank my co-supervisor, Prof. Dahua Lin. We have worked together through many great projects and had countless inspiring discussions. I have learned from him how to think out of the box and then consolidate the ideas with serious analysis. He is and will always be my role model as an

creative and disciplined young scholar. I also want to give my thanks to Wei Liu and Deli Zhao, who had given me great help during my early years of research life. Their enormous patience and step-by-step guidances have led me all the way from an inexperienced beginner to a qualified PhD candidate.

Furthermore, I would like to thank all my current and former colleagues in the Multimedia Laboratory. Many works in this thesis would not be possible without the discussions and insightful advices from them. My thanks also go to my caring friends in and outside this great university, Chao Dong, Jiyi Cheng, Linjie Yang, Shuo Yang, Willow Zhang, Yonglong Tian, and Zhenyao Zhu, to name a few. I am always grateful for their undoubted support and really enjoy the time we spent together.

Finally, I would like to thank my father, Fangming Xiong, and mother, Jiaming Yuan. They have always backed me up in my hardest time and supported me unconditionally. I feel that words are not enough to express my gratefulness. May health and happiness be with them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Visual recognition describes a set of tasks for visual system to recognize visual patterns, objects, and concepts. Due to its unlimited potential applications, it has always been in the central position of computer vision research, drawing plenty of efforts. With years of hard works, people have amasses plenty of previous works on this task. Most classical approaches to deal with visual recognition can be viewed as the combination of two components: a hand-crafted feature extraction module and one machine learning based classifier module. It has been shown that, simple hand-crafted features are not enough to recognize complex visual objects. The major obstacle here, is a giant semantic gap from low-level feature descriptors, which are often based on pixel statistics or gradients, to complex visual concepts in our lives. Due this difficulty, recognition system has paced slowly for decades.

1.1 Deep Learning

A series of breakthroughs took place in the past several year with the introducing of deep learning based methods. The core idea of deep learning features two important concepts: 1) a unified architecture that performs feature extraction and classification in one pass, and can jointly optimize all its parameters, which may sometimes be referred to as *end-to-end learning*; 2) a deep architecture that are composed a lot of intermediate neural networks modules (sometimes called "layers"), which process information consecutively and thus ease the abstraction burden for each module. With these two distinct properties, deep learning has achieved exciting results on many of visual recognition tasks, including face recognition, single object recognition and detection.

In spite of these great achievements, most current deep learning methods are still focusing on one single perspective of its input data. For example an object recognition system usually only considers the pixel appearances as they are sufficient for recognizing the object in one input image. As we are moving toward higher-level visual understanding tasks, this simple approach will gradually fall insufficient to fulfill the requirement for precise recognition. The major problem is that high-level visual concepts, such as events, activities, or other general visual concepts, usually refer to multiple simple objects, motions, changes, and sometimes their interactions. These important information

will not all present in one perspective of the input images or videos, requiring techniques to identify and utilize information from multiple aspects of data. However, how to achieve this is still an open question, which motivates us to explore the solutions.

1.2 Combining Multiple Aspect of Data

It has been described above the importances of combining multiple aspects of data. The question remains how to efficiently realize this philosophy. In general, there are three major challenges for related approaches. The first one, before any learning can be performed, is to identify potential useful perspectives of data. For example, in event recognition, the spatial configuration of object is a rarely explored aspects but serves an important role in forming an event. The second challenge is how to represent and comprehend the additional aspects of data. As an example, the optical flow data in action is known to provide information on shot-term motions, but there lack an efficient model to capture the long-term information. The last but not the least one, is how to combine the information surfaced from these aspects. In this thesis work, we aim to address these challenges in context of specific high-level visual understanding tasks.

1.3 Our Approaches

In this thesis work, we deals with three visual understanding tasks, namely *complex event recognition*, *multi-label image tagging*, and *human activity classification from video*. In dealing with these problems, we bear in mind the philosophy of combining multiple aspects of data. For each specific task, we get the solution by addressing the challenges described in Sec. 1.2. By evaluating the performances gain of the proposed models against the baselines, we can examine the effectiveness of this strategy.

Complex Event Recognition The essential task of a event recognition system is to decide for each image the underlying event happening. The target events are mostly daily events in normal people's life, which makes the source of image abundant on the Internet. One major difficulty of event recognition problem lies in that there is a great semantic gap from appearances on the image to complex meaning of a daily event. Some events are the composition of multiple objects, people, and their interactions. This makes the baseline model with single convolution neural network, which is designed mainly for dominant object recognition task, insufficient to extract enough information to consolidate the classification.

In the first part of this thesis work, a multi-layer framework is proposed to tackle this problem, which takes into account both visual appearance and the interactions among humans

and objects, and combines them via semantic fusion. To address the problem of how to represent their interactions and incorporate them into a deep model, a novel strategy is devised which projects detected instances onto multi-scale spatial maps. On a large dataset with $60,000$ images, models based on the proposed framework achieved substantial improvement over the state-of-the-art, raising the accuracy of event recognition by over 10%. This part of work has been published in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2015.*

Multi-label Image Tagging Multi-label image tagging is to assign an uncertain number of related visual concepts to one input image. The candidate concept pool can contain thousands of concepts, where most of them are irrelevant. As another form of high-level visual understanding, the difficulty here is how to precisely identify the highly versatile and numerous visual concepts with as least misses as possible. In this task, seeing the image from single scale, and predicting labels independently, as the baseline methods does, will not work optimally in solving the problem. This reveals the necessities of seeing the images from multiple scales and views, and learning the visual concepts by comparison and contrasting.

In the second part of this thesis work, a new image tagging framework suited for large-scale real-world applications is proposed. The frameworks comprises two complementary tech-

niques: Scaled View Integration, a new modeling strategy that takes into account the association between tags and local regions, and Contrastive Bundled Loss, a new loss formulation with strong scalability to large tag space. Experimental results on large datasets show that the combination of both techniques results in superior tagging performance and higher learning efficiency as compared to the state-of-the-art.

Human Activity Video Recognition Human activity recognition is about classifying the human performing actions in a video clip into the corresponding categories. Moving from still images to videos, the challenges not only arises from the increased amount of image frames, but also from motion and temporal structures that are non-existent in single images. Activities, by their definitions, depict changes of appearance in a moderate period of time. Although short-term motion information is usually characterized by dense optical flow. It is considered very hard to be modeled with deep neural networks. Meanwhile, the long-term structure, which could serve more vital roles in deciding the category of an action, rarely received any attention in the research community. High-level visual understanding, in this case, calls for an system that can combine the above aspects of the video data to perform accurate predictions.

In the third part of this thesis work, a novel framework, called temporal segment networks, is proposed to tackle the activity classification problem. This framework features a very deep

segmental architecture which enhance the modeling capacity of deep networks by fusing information from different temporal segments of an action video. It also utilizes more diverse representation of both motion and appearance than previous approaches. In experiments, the proposed approach improves state-the-of-art performances on several datasets by a large margin. With the help of a novel visualization tool, it is qualitatively demonstrated that performance gain of the learned model comes from the propose approach as expected. This part of work has been accepted to *European Conference on Computer Vision (ECCV) 2016*

Chapter 2

Recognize Complex Events from Images

The explosive growth of web images, driven primarily by the thriving of online photo sharing services such as Flickr and Instagram, has been gradually and profoundly transforming our lives and the way we communicate. Many of these images are event photos, namely the ones that capture human activities in either private or social contexts. Such images not only provide valuable records of our lives and our world, but also convey useful information that one can exploit to analyze consumer preferences or study socioeconomic trends. The primary goal of this paper is to develop an effective method for recognizing events from images. Inspired by the recent success of deep learning, we formulate a multi-layer framework to tackle this problem, which takes into account both visual appearance and the interactions among humans and objects, and combines them via semantic fusion. On a large dataset with $60,000$ images, the proposed

method achieved substantial improvement over the state-of-the-art, raising the accuracy of event recognition by over 10%.

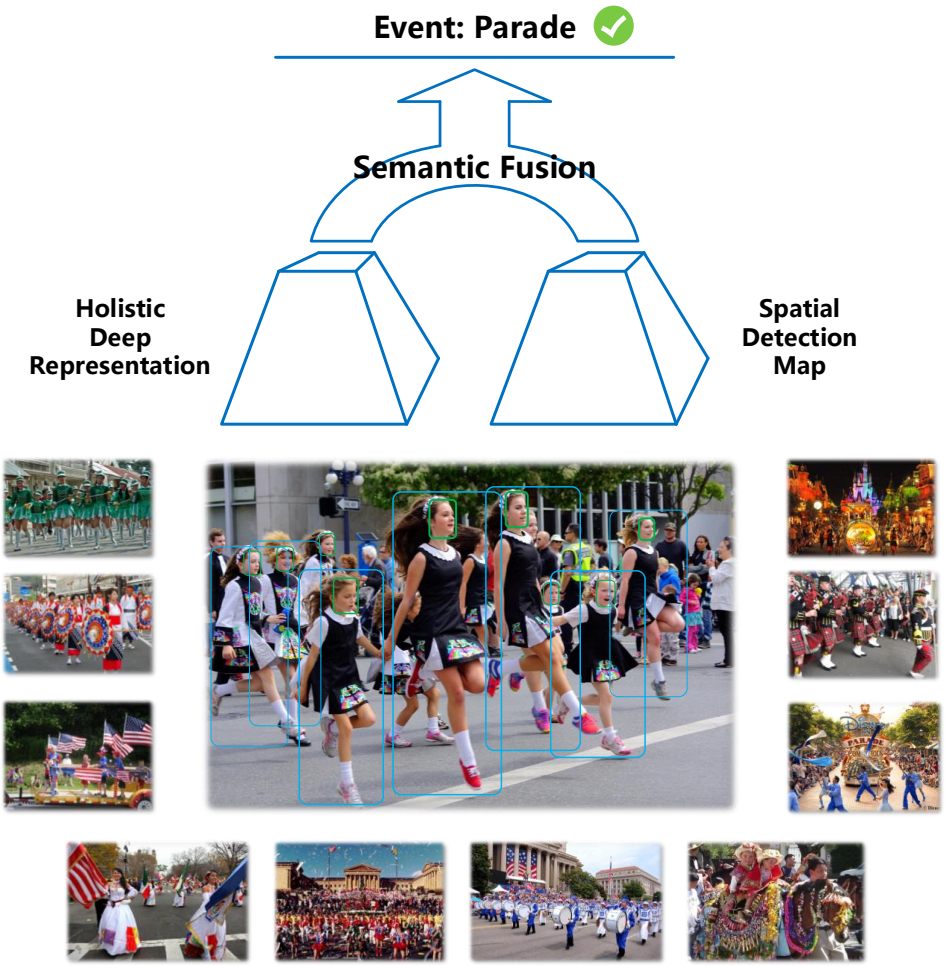


Figure 2.1: Event recognition is highly challenging due to the large semantic gap. Even in the same event class, *Parade*, the images can look very different. This calls for methods that are capable of reasoning about high-level semantics by fusing evidences of multiple aspects.

2.1 Introduction of Event Recognition

Event recognition is not a new story in computer vision. However, most existing efforts [84, 16, 4] are devoted to recognizing events from *videos*. Our daily experience seems to suggest people can effortlessly identify events from photos most of the time. This motivates us to explore a new approach, one that is able to recognize events from static images.

This is a challenging problem. A major obstacle standing in our way is the large gap between high-level event semantics and low-level visual features. Event images are complex as compared to object images. They usually involve multiple objects interacting with each other. As we can see in Figure 2.1, two images capturing the same kind of events can be vastly different in their visual structures. Traditional methods that rely mainly on shallow analysis of visual appearance would be faced with substantial difficulties when applied to this task.

Recently, the use of convolutional neural networks (CNN) has led to remarkable progress in several important vision tasks, including image classification [41], object detection [24], and face verification [81]. This line of work clearly demonstrates the superior capability of deep models in capturing complex variations and the critical role of intermediate layers in bridging the semantic gap. Following the lead of these efforts, we explore the use of deep learning in this work, with an aim to bring its success to the next level – from recognizing individual objects to

understanding complex images as a whole.

Events, by nature, are defined by the interactions among key *entities*, including *humans* and *objects*. Therefore, identifying such entities in an image is a key step towards event understanding. While a convolutional network formulated upon entire images is very powerful in modelling visual appearance, we found empirically that it is not as effective as an dedicated detector, especially in detecting humans. Our idea to tackle this problem is very simple – use dedicated detectors to locate relevant entities and incorporate them with the convolutional network to predict the event class.

However, bounding boxes of detected objects and visual appearance features are very different by nature, and can not be combined using conventional feature combination methods. In this paper, we propose a novel way to address this. Instead of directly using the bounding boxes, we project them onto multi-scale spatial maps, bring the resultant maps together, and thereon construct a convolutional network to derive a higher-level representation. This construction not only provides a way to express detecting results that is suited for higher-level analysis, but also makes it possible to exploit the spatial co-occurrences of different objects, which are important cues of their interactions. With two convolutional networks, one upon the image and the other upon the detection maps, we integrate them via *semantic fusion* and obtain a fused representation that captures key semantic elements of the event image.

The major contributions of this work are summarized here: (1) We explore a new approach to event recognition, which, unlike most previous methods, rely solely on static images. (2) Recognizing that interactions among people and objects are essential for event understanding, we propose using dedicated detectors to locate key entities, and develop a novel strategy, namely multi-scale spatial maps, to uniformly represent the detected results. (3) We propose a new framework that combines evidences from multiple channels via semantic fusion. (4) To facilitate this study and to promote future efforts towards image-based event recognition, we construct a large dataset comprised of nearly $60,000$ images annotated with event classes. The dataset can be found in the project website listed in the supplementary materials.

The rest of the paper is organized as follows. Section 2 provides a brief review of related work. Section 3 introduces a new dataset for image-based event recognition, called WIDER. Section 4 discusses the proposed framework in detail. Section 5 presents the experimental results. Finally, we conclude this paper in Section 6.

2.2 Related Works

Event recognition is a very active area in computer vision [105]. Most existing methods rely on videos to recognize events [14], with emphasis placed on the use of dynamics and temporal

relations [84, 4]. These methods generally fall in three categories: *feature-based* [14, 91, 71], *concept-based* [100], and *model-based* [31, 103]. Recently, Duan *et al.*proposed a new method [16] that utilizes web images to help video-based event recognition. Despite the technical differences among these methodologies, they all rely heavily on using the dynamics extracted from videos and therefore can not be directly applied to static images.

Understanding of still images is an active field of research. Efforts on holistic scene understanding [67] are also related to this work, as they both target high-level interpretations of given images. Yet, essential differences exist. Prior work on scene understanding mainly considers visual patterns, with relatively less attention to human activities, which, however, are a key factor in event analysis. In this paper, we take into account this factor through a dedicated channel and derive a novel way, namely multi-scale maps, to incorporate it.

Analyzing human actions [33] with the help of human poses [104, 58] and human-object interactions [12, 106] also provides significant cues in recognizing certain categories of events. However it is worth to note that the events we investigate here are usually characterized by combinations of multiple aspects, including background appearance, spatial patterns of people, and their interactions. Hence this work is related but different from recognizing actions of individuals.

Among previous work on image understanding, a CRF-based method proposed by Li and Fei-fei [50] that jointly infers the

classes of event, scene, and objects is perhaps the most related. This method couples two LDA models formulated directly upon low-level features, and therefore lacks the capability of capturing complex variations and bridging the semantic gap. It is also worth noting that many previous methods [51] require training sets with detailed annotations (*e.g.* object bounding boxes), which are often very costly to obtain. On the contrary, our method only needs training images labeled with event classes, making it particularly appealing to large-scale applications.

A key strategy adopted in this work is to combine information from multiple channels. This strategy has been widely used in previous work. In conventional frameworks, the fusion of channels is usually accomplished by combining features [53] or optimization objectives [54]. A limitation of these approaches is that they are not able to exploit the relations among the constituent elements of different channels. Following the recent success of deep models [41, 57, 17], attempts [79, 62] have been made to connect multiple modalities through deep networks. In recent work, auxiliary channels, such as depth [29] and optical flow [75], are captured using additional networks. It is worth emphasizing that depth maps or optical flows are both spatial maps by nature and thus it is relatively easy to construct C-NNs thereon. However, incorporating external detectors that produce bounding boxes is not as straightforward. In this work, we develop a novel method, namely the multi-scale maps, which provides a principled solution to this problem. This method en-

ables us to directly draw on state-of-the-art detectors [24, 13, 49] for improving the overall recognition performance.

2.3 WIDER: A New Dataset

Datasets are an important force in driving the advancement in a research area. Whereas there have been plenty of datasets for object recognition [73], scene understanding [101], and video-based event recognition [66]. A dataset to support the research on image-based recognition remains needed. Along with this work, we constructed a large dataset from web images, called *Web Image Dataset for Event Recognition (WIDER)*. This dataset contains $60,000$ images of 60 event classes, where the numbers of images in different classes are balanced. All images have been carefully annotated with event labels, which can be used for model training and performance evaluation. Figure 3.1 show some examples of the data. We can see that the dataset comprises a diverse set of event categories and there exist substantial variations in visual patterns among the images within each category. We will make the dataset available to the public following the publication of this paper in order to foster future research on this topic.

Construction of this dataset took a lot of efforts. This course is comprised of three stages:

Selecting event categories. A majority of the event categories are from the *Large Scale Ontology for Multimedia (LSCOM)* [60],

which provides a list of around 1,000 concepts relevant to video event analysis. Many of these concepts are the names of objects or low-level actions. Hence, we manually go through the list, picking those representing event classes while filtering out the others. We also noticed that the concepts in *LSCOM* are primarily from TV news, and consequently events in personal lives were not thoroughly covered. To enrich the dataset, we invited a group of students to propose activities related to their daily lives and find a number of new categories therefrom, *e.g. car-driving*. Altogether, we obtained 60 event classes.

Collecting images. We resorted to search engines like *Google* and *Bing* to collect images. Specifically, we retrieve 1000 to 3000 images for each category using the class name as the input query. We found that many images resulted from this process are simply irrelevant. To obtain more qualified images, we adopt the *query expansion* strategy. In particular, we acquire additional queries for each event class by finding highly frequent phrases from a variety of sources, such as WordNet, Wikipedia, and the text snippets that come with the retrieved images. Using these phrases as queries to expand the search substantially enrich the pool of candidate images for building the dataset.

Screening data. The collection process above results in hundreds of thousands of candidate images. In this pool, lots of samples are cartoons or cliparts while many others are irrelevant to the events of interest. To clean the data, we first filter

Figure 2.2: Examples of several categories in the WIDER dataset, which exhibit diverse visual patterns.

out cartoons, cliparts, and blank images using bilateral filtering[1]. Then we asked human annotators to identify irrelevant images in the remaining set. To expedite this process, we developed a GUI tool, where the images are grouped into pages and hence the annotator can inspect 80 images at the same time. In this way, we can process a large quantity of images very quickly and reliably. The screening retained about 60,000 images in the dataset.

2.4 Fusing Multiple Information as Channels

Generally, an event can be considered as an activity taking place in a certain environment. Hence, it can be reasoned from two aspects: (1) Environment: *e.g. is it by the seashore or in a forest? is there a large crowd of people?* (2) Activity: *e.g. is the man running? are the people in the scene sitting together?* Event recognition, in essense, is a process to answer such questions

[1]The overall response of a bilateral filter can be used to test whether an image has enough textures to be qualified as a real-world photo.

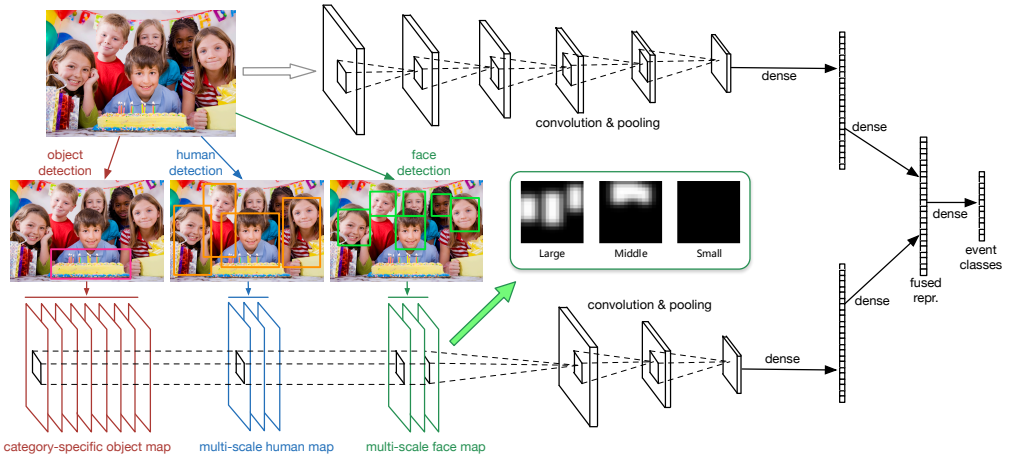


Figure 2.3: Overall, this framework integrates two channels. The upper channel, devised to capture the visual appearance, is formulated directly upon the input images; while the lower channel, devised to capture the interactions among humans and objects, takes as input the results of three detectors, respectively for faces, humans, and objects. In this channel, the bounding boxes obtained by the detectors are projected onto multi-scale spatial maps, which are then modeled by another CNN. On top of both CNNs, a fused representation is introduced, which is linked to the top representations of both networks, respectively via a fully-connected layer.

and arrive at a prediction by bringing the answers together.

Following this consideration, we develop a multi-layer framework as shown in Figure 2.3. This framework is comprised of two major channels, one is to model the observed visual patterns, which are important for reasoning about the environment; while the other is to capture the interactions among humans and objects, which are significant cues of the activity taking place. Particularly, to ensure the reliability of detection, the latter channel employs state-of-the-art detectors to locate the

entities of interest (*i.e.* humans and objects), and subsequently uses spatial maps to express the distribution of the detected results. This enables the use of deep models to capture the variations in their spatial configurations. These two channels are combined through a semantic fusing layer, resulting in a fused representation that captures the key semantic elements of the image. In what follows, we will introduce these components in detail.

2.4.1 Model Visual Appearance with CNN

We use a deep convolutional neural network (CNN) to model the visual appearance of event images. In previous work [41], CNNs have demonstrated excellent capability of capturing complex variations in visual patterns. Here, we are interested in studying how well they perform in higher-level tasks, *e.g.* event recognition. Particularly, we adopt the architecture of AlexNet presented in [41].

This network comprises eight layers, five convolutional and three fully-connected, and takes as input a 3-channel color image of size 224×224. The 1st, 2nd, and 5th convolutional layers are each followed by a max-pooling layer to compress the inputs. Each fully connected layer has 4096 neurons. The last layer is linked to a multi-way softmax classifier with dense connections. The settings of these layers follow [41]. The detailed model specification will be provided in the supplemental materials.

2.4.2 Find Humans with Complementary Detectors

We found empirically that humans appear in a majority of images in our dataset. This is not surprising. The interactions among humans are often a key factor in defining an event. However, locating humans from event images is very challenging. In such images, people are often occluded by one another, and their facial appearance can be seriously blurred when they are far away from the camera. There are also cases where faces of some people are completely invisible, as they are facing towards the opposite side. To tackle this problem, we combine two complementary techniques: *face detection* and *human detection*. As Figure 2.4 illustrates, this strategy can substantially increase the chance of successful detection even under adverse circumstances – when one technique fails, the other can come to rescue.

Specifically, we use the SURF cascade presented in [49] for face detection. This method uses multi-dimensional SURF features for describing local patches together with an improved weak classifier for boosting, thus significantly increasing the runtime efficiency without compromising the accuracy. For human detection, we employ the ACF detector developed in [13], which uses a feature pyramid for multi-scale detection with an approximation to speed up the computation. Both detectors are highly efficient and thus are suited for large-scale applications.

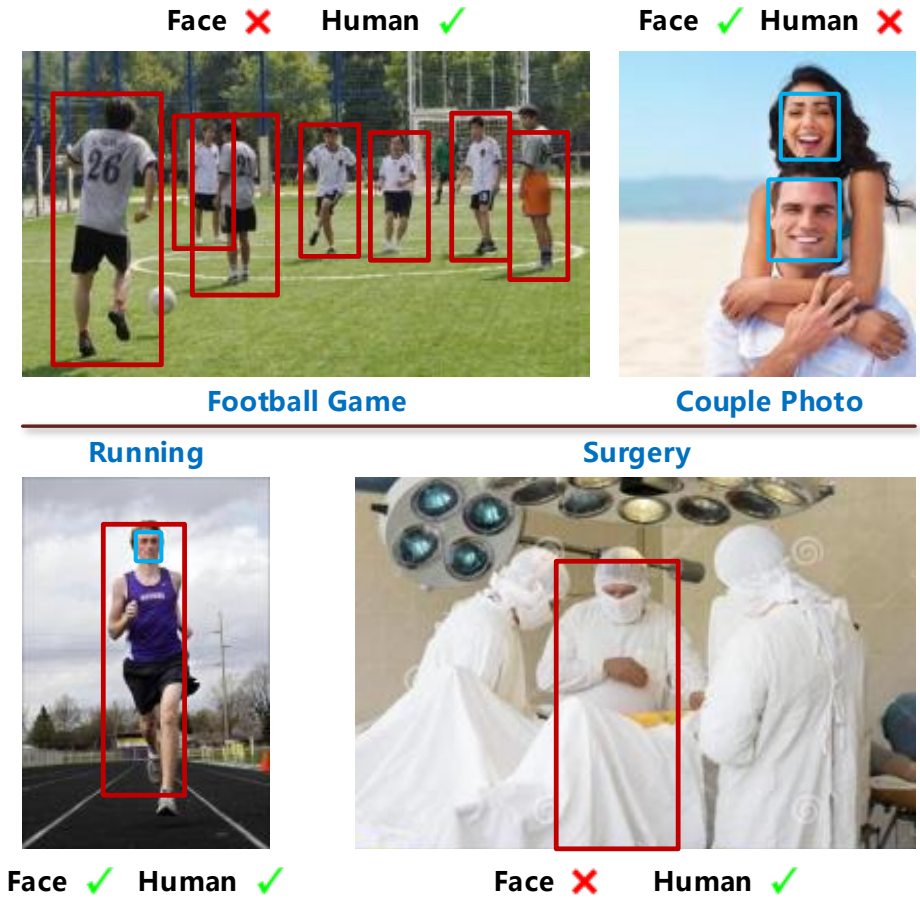


Figure 2.4: The face and human detectors are complementary. In case one detector fails, the other tends to find out the missed humans in image.

2.4.3 Multi-scale Spatial Maps

Detectors output bounding boxes. Each bounding box is represented by a 4-tuple comprised of the corner coordinates. These boxes contain rich information about the event, which, for example, include the spatial distribution of entities and their geometric relations *i.e.* relative location and size. However, a question

arises here: *how can CNNs understand the bounding boxes?* This problem is not as trivial as it seems to be. Simply concatenating the coordinates of all bounding boxes does not yield a sensible representation.

Our idea to tackle this problem is simple. Since the primary message conveyed by these bounding boxes is the spatial configuration of the entities, to get this message, we can project the boxes onto a spatial map. Here, a *spatial map* is a binary image with the elements covered by detected objects set to one. However, there is an issue with this approach. Consider the two images in Figure 2.5, one containing a group of people, while the other containing two larger faces that cover a similar region. While these images represent very different events, one cannot distinguish them by inspecting their spatial maps.

Here, we propose a solution – *multi-scale spatial maps*. Instead of using a single channel to capture all detected entities, we expand the map into multiple channels, each for a scale level, so that entities of different scales will be reflected by different channels. In particular, we use a multi-scale spatial map comprised of three scale channels to represent the detected faces, where each channel is a binary map of size 18×18. We use two scale thresholds τ_1 and τ_2 with $\tau_1 < \tau_2$ to determine the choice of channels. Given a bounding box, we normalize its coordinates *w.r.t.* the 18×18 frame and compute its area a. If $a < \tau_1$, we project it to the 1st channel, setting all the covered elements of this channel to one. Otherwise, we project it to the 2nd or 3rd

Figure 2.5: Here is an illustration of multi-scale spatial maps. Over these two images, the face detector produces bounding boxes of different sizes. Spatial maps resulted from the projection of these boxes are difficult to be distinguished from each other. However, when boxes of different sizes are projected onto different channels (L, M, and S), the distinction between these maps becomes much more obvious.

channel, depending on whether $a < \tau_2$ holds.

Likewise, we can apply this multi-scale representation to express the results obtained from the human detector. Altogether,

we have a spatial map with 6 channels, 3 for faces and the other 3 for human bodies. This method not only provides a uniform representation that can be readily handled by higher level models, *e.g.* CNN, but also makes it possible to differentiate the spatial configurations at different scales, *e.g.* crowded gathering *vs.* private conversation.

2.4.4 Detect and Characterize Objects

Besides humans, the presence of objects of certain categories is often a strong indicator of some event classes. Figure 2.6 uses several examples to illustrate this relation. In this paper, we use R-CNN [24], a state-of-the-art technique in object detection, to locate objects of interest.

The R-CNN method consists of three steps. First, it requires object candidates to be generated. For this purpose, we use a latest technique, called *Edge boxes* [111], which is much more efficient than the standard selective search algorithm [87]. On average, *Edge boxes* takes 0.25 second to process an image, while selective search takes about 10 seconds. Subsequently, a 4096-dimension CNN feature is derived for each candidate, which is then fed to SVMs to predict whether it belongs to specific object classes or not. Finally, a greedy non-maximum suppression procedure is applied to filter out redundant candidates.

We observed objects of thousands of different classes in our dataset. Many of them, however, are irrelevant to event under-

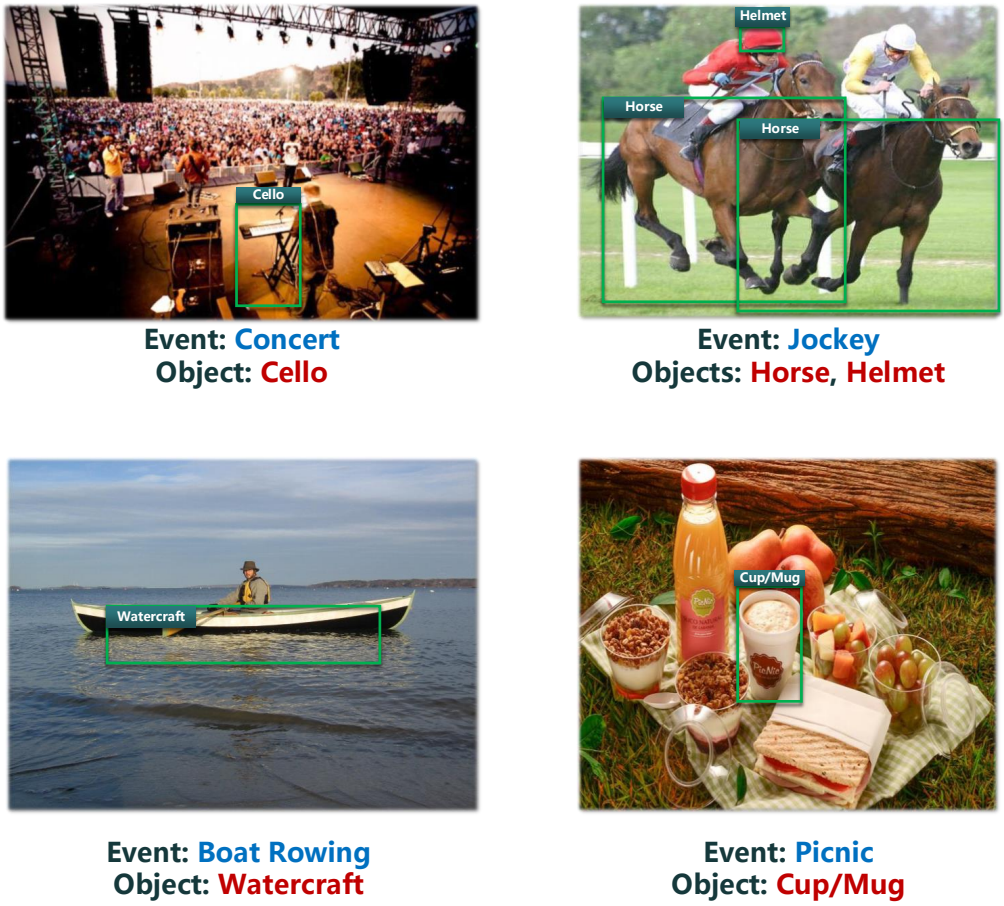


Figure 2.6: Existence of significant objects indicates the event categories. For example, the presence of horses and helmets is a strong indicator to the class *Jockey*.

standing. To choose the ones that are truly pertinent to our task, we run a large collection of object detectors over a subset of event images, and select the 30 most frequently occurring classes[2].

Again, we use spatial maps to express detected objects. Unlike humans, we have a number of object classes but the presence

[2]The number of object classes was determined using cross validation. We found the risk of overfitting to be higher as we use more object classes.

of a specific object class is generally quite sparse. Hence, we use *class-specific maps* instead of *multi-scale maps* for general objects (except humans). In particular, we construct a spatial map with 30 channels, each for an object class. When an object is detected, the bounding box will be projected onto the corresponding channel. This representation enables one to exploit the interactions among objects, *e.g.* co-occurrences of objects of different categories.

2.4.5 Channel Fusion

Stacking the spatial maps for faces, humans, and objects, we obtain an integrated spatial map with 36 channels, each of size 18×18. We construct a convolutional network thereon to derive a higher-level representation. Through a series of empirical experiments, we obtain an architecture suitable for modelling such spatial maps. This architecture comprises two convolutional layers. The first layer filters the inputs with 64 kernels, each of size $3 \times 3 \times 36$, producing an output of size $18 \times 18 \times 64$. This is followed by a max-pooling layer that compresses the result into an array of size $6 \times 6 \times 64$. The second convolutional layer, with 128 kernels of size $1 \times 1 \times 64$ is then applied, yielding an output of size $6 \times 6 \times 128$. Here, the first convolutional layer is to exploit the spatial interactions among neighboring parts and the co-occurrence patterns of different entities, while the second layer is mainly to adjust the relative contribution of different

channels. The output of the second layer is then linked to a representation layer via a fully-connected network, resulting in a 4096-dimensional vector to capture the information derived from the detectors. Note that the detection channel needs less layers compared to the network for visual appearance. This is partly due to the reason that the detectors perform a series of visual analysis internally, which already narrows the semantic gap to some extent.

The visual appearance channel and the detection channel respectively yield a 4096-dimensional representation at the top. Through the computation across multiple layers, these representations are abstracted away from the low-level variations and thus are more consistent in expressing the semantics. To integrate both aspects, we further introduce a *semantic fusion layer*, which is linked to the top layers of both channels via dense connections, and thereon derive a 4096-dimensional *fused representation*. Like in other discriminative networks, this fused representation will be linked directly to the event classes via a softmax layer.

2.4.6 Training Algorithms

At the training stage, the CNN of the first channel was pre-trained on ImageNet [41]. This relives the over-fitting problem of deep models in the sense that natural images share similar low-level features. For the CNN of the second channel, the weights were randomly initialized from a zero-mean normal dis-

Table 2.1: Class averaged recognition accuracy.

Method	Top-1 Accuracy	Top-5 Accuracy
Gist [65]	13.8%	34.6%
SPM [45]	26.8%	47.2%
RCNNBank	37.7%	62.5%
CNN [41]	38.5%	65.5%
FCNN+H	42.1%	67.3%
FCNN+H+O	**42.4**%	**67.5**%

tribution. After initialization, the entire framework is jointly trained using stochastic gradient descent. Training strategy like data augmentation, weight-decay, and dropout are also used to alleviate over-fitting. The learning rate is initialized at 0.001 for the pre-trained CNN, while the learning rates of the two convolutional layers for the second channel are set to 5 and 2 times the base rate.

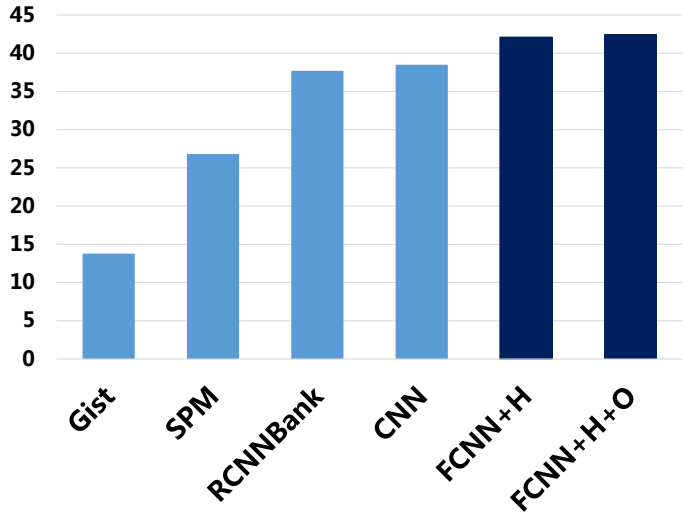
2.5 Experimental Results

We conducted experiments on the WIDER dataset (described in Section 2.3) to evaluate the proposed method and compare it with representative methods on image classification. The entire dataset, which contains $60,000$ images in 60 classes, is randomly divided into two disjoint halves, one for training and the other for testing.

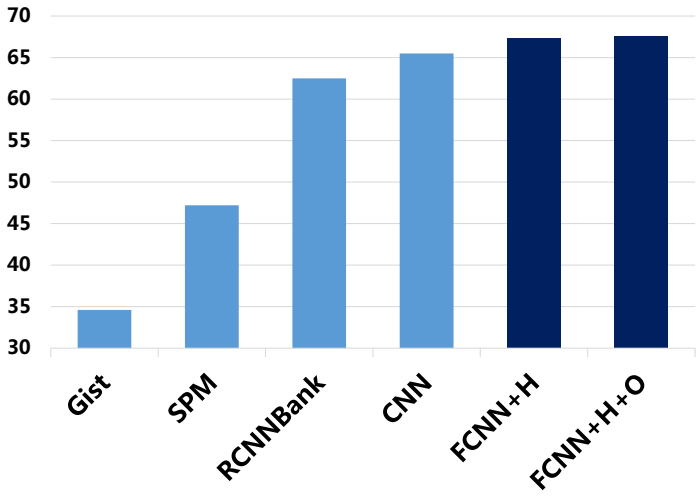
We tested our method under two settings: *"FCNN+H"* and *"FCNN+H+O"*. The former is a simplified version where the de-

Table 2.2: Comparison of per class recognition accuracy. To save space, we only show abbreviations of category names here. We compare the accuracy of FCNN with the original fine-tuned CNN on these categories. With the help of spatial detection maps, accuracies on 40 out of 61 categories have been improved.

	Para.	Hand.	Demo.	Riot	Danc.	CarA.	Fune.	Chee.	Elec.	Pres.
CNN	.412	**.322**	.572	.141	**.221**	.630	.295	**.249**	.171	.502
FCNN+H+O	**.500**	.308	**.631**	**.219**	.207	**.679**	**.373**	.232	**.171**	**.525**
	Peop.	Meet.	Group	Inte.	Traf.	Stoc.	Awar.	Cere.	Conc.	Coup.
CNN	.160	.534	.266	.239	**.541**	.393	.314	**.306**	.276	.337
FCNN+H+O	**.162**	**.555**	**.269**	**.273**	.524	**.393**	**.400**	.301	**.359**	**.366**
	Fami.	Fest.	Picn.	Shop.	Sold.	Sold.	Sold.	Spa	Spor.	Stud.
CNN	.271	.262	.617	.297	.295	.457	.158	.666	.167	.122
FCNN+H+O	**.366**	**.309**	**.637**	**.304**	**.313**	**.512**	**.203**	**.672**	**.200**	**.146**
	Surg.	Wait.	Work.	Runn.	Base.	Bask.	Foot.	Socc.	Tenn.	IceS.
CNN	.567	.585	.229	.498	.590	.645	**.587**	.523	.643	.580
FCNN+H+O	**.612**	**.591**	**.326**	**.529**	**.596**	**.647**	.580	**.573**	**.663**	**.630**
	Gymn.	Swim.	CarR.	RowB.	Aero.	Ball.	Jock.	Mata.	Para.	Gree.
CNN	.631	**.652**	.774	.670	.389	**.348**	.268	.725	.440	.141
FCNN+H+O	**.637**	.642	**.807**	**.718**	**.400**	.345	**.328**	**.791**	**.494**	**.157**
	Cele.	Dres.	Phot.	Raid	Resc.	Spor.	Voter	Angl.	Hock.	Peop.
CNN	.364	**.617**	.267	.254	.231	.104	.091	**.444**	.608	**.429**
FCNN+H+O	**.405**	.615	**.310**	**.308**	**.259**	**.125**	**.125**	.415	**.608**	.410
	Stre.									
CNN	.115									
FCNN+H+O	**.124**									



(a) Top-1 Accuracy



(b) Top-5 Accuracy

Figure 2.7: Average recognition accuracy by percentages.

tection channel only uses the results of face & human detection, while the latter is the full version with both humans and objects taken into account. We also compared it with *"Gist"* [65], *"Spatial Pyramid Matching (SPM)"* [45], *"ObjectBank"* [52], and *"CNN"* [41]. These methods have been widely adopted in prac-

tical systems. Note that when we implemented *ObjectBank*, we made an important improvement, using the responses of R-CNN instead of the original SVM detectors. This change, which we call *"RCNNBank"*, leads to much better performance.

For all these methods, we learned the model on the training set and assessed them on the testing set. The performance was evaluated in terms of *top-1* and *top-5* accuracies. Specifically, each method was used to predict a ranked list of class labels for each testing image based on classification scores, which is then compared with the ground-truth. If the ground-truth is within top k positions of the list, we call the prediction *top-k accurate*. Then, *top-k accuracy* is defined to be the fraction of top-k accurate predictions.

Comparison of results. The performance is compared in Table 2.1 and Figure 2.7. The results show that methods using deep learning techniques outperform all others, *i.e.* Gist and SPM, by a large margin. This, again, demonstrates the superior capability of deep models in capturing complex visual variations as compared to traditional techniques. More importantly, our framework, with the detection channel incorporated, takes this capability to a next level, significantly improving the classification accuracies. Compared to CNN, the top-1 accuracy increases from 0.385 to 0.424 – the gain is over 10%. This result corroborates with our intuition that the detection channel conveys complementary information and that the multi-scale maps

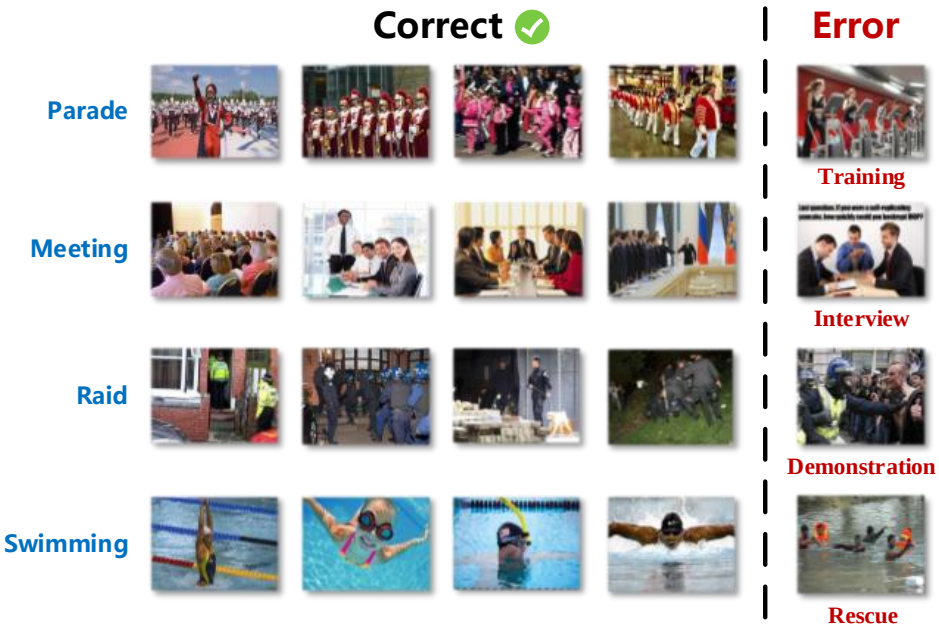


Figure 2.8: Successful and failed prediction examples on the testing set. Misclassified samples are shown with their ground-truth categories.

provide an effective means to utilize such information. Also, the use of face and human detectors makes up for the weakness of appearance-based CNN in object localization.

Table 2.2 offers class-specific comparisons. For 40 out of 60 classes, our method outperforms CNN [41]. For those classes where humans play a crucial role, the gain is remarkable. For example, the top-1 accuracies are nearly doubled for classes like *"Marching (Marc.)"* and *"Couple Photo(Coup.)"*. Figure 2.8 presents some successful and failed predictions of our model. Taking a closer look here, we can see that this model is able to identify images relevant to the same event in spite of the large variations in their visual appearance. On the other hand, many of the examples that are incorrectly classified tend to be easily confused, as the "true" classes and the predicted classes of these examples often look very similarly.

Contribution of object detection. Compared to the significant improvement due to the use of face and human detection, the performance gain brought by the object channels doesn't seem to be as notable. When investigating this issue, we found that non-human objects are only detected in about one-fourth of the images. Particularly, out of all the testing images, about 7100 contain detected non-human objects. We specifically evaluated the performance on this subset, and observed greater performance gain due to the object channels, as shown in Table 2.3. We note that the effectiveness of the object channels hinges largely

Table 2.3: Performance comparison on the "with-object" set.

Method	CNN	FCNN+H	FCNN+H+O
Top-1 Accuracy	45.56%	48.9%	49.6%
Top-5 Accuracy	71.4%	73.6%	75.3%

on the performance of the object detectors. While the R-CNN detectors [24] already represent the state-of-the-art, the overall performance remains quite limited (with AP at 31.4%). However, the computer vision community is making steady progress in object detection [73]. It is reasonable to believe that with better detectors, we can see even greater improvement with the use of object channels.

Run-time performance. We implemented the framework based on Caffe [36], a popular programming platform for deep learning. The training phase involves preprocessing (detecting humans and objects) and parameter learning. A majority of the computation is performed on GPU. With a GTX Titan, it takes about 3 seconds on average to preprocess an image, and 3 hours to train the deep networks over the entire training set with about $30,000$ images. Given a new image, it also takes about 3 seconds to preprocess. Compared to preprocessing, the time needed to make the prediction is negligible (about 2.4 milliseconds per image).

2.6 Discussion and Summary

In this part of thesis work, a new framework is proposed for recognizing complex events from static images. This framework integrates evidences from a visual appearance channel and a detection channel, both via deep convolutional networks, to predict the event class for a given image. It is particularly worth noting that we use multi-scale spatial maps in expressing the results obtained from dedicated detectors, thus enabling the use of higher-level models, *e.g.* CNN, to capture the spatial configurations of objects and their variations.

The experiments over a large dataset clearly demonstrated the effectiveness of the proposed method. In particular, our method achieves notable improvements over state-of-the-art visual recognition techniques, increasing the accuracy by over 10%. This clearly demonstrate the power of our approach by adopting the idea of combining multiple aspect of data. It opens encourage future efforts which ventures through this direction.

Chapter 3

Recognize Multiple Concepts from Images

In the event recognition tasks, each image is only assigned one class label. However, our visual world is full of information and changes. This necessitate a system to produce multiple labels for one input image. Multi-label image tagging is the exact task that deals with this problem. It also serves as a good sample of how combining multiple aspect of data can improve a specific high-level visual understanding system. We develop a new image tagging framework suited for large-scale real-world applications, particularly aiming at two key challenges that are often ignored in prior works: (1) a tag may be relevant to a local region instead of the entire image; and (2) the large number of tags would lead to difficulties in scaling. By enforcing the strategy of combining multiple aspect of data, the proposed techniques results in superior tagging performance and higher learning efficiency as compared to the state-of-the-art.

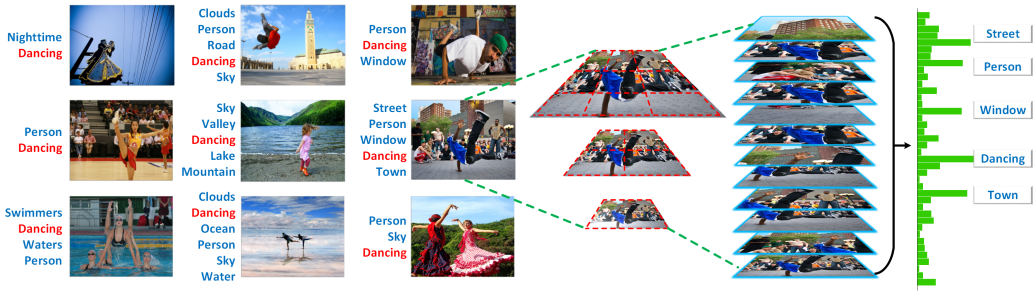


Figure 3.1: Illustration of our approaches. Note the rich information can be obtained by contrasting between the shown images, even though they share the same tag, *dancing*. For each image, we devise the scaled-view integration technique to tackle the problem of *local association*, where tags refer to local regions of of different sizes/scales.

3.1 Image Tagging in the Wild

Years of practice in image organization and sharing suggests that *tagging* is an efficient way to manage images. Many online photo sharing services, such as Facebook, Instagram, and Flickr, allow users to provide tags to photos. However, a substantial portion of the images remain untagged, as many see this as an extra burden. Even for those images with user-provided tags, the tags are usually incomplete, noisy, and unreliable. Driven by such real-world demands, automatic image tagging has attracted increasing attention from the computer vision community over the past decades, resulting in numerous techniques [56]. Recent advances in deep learning [42] also inspires a wave of new efforts that attempt to bring the power of deep neural networks to this area, leading to significant improvement in performance [26].

However, it is worth noting that existing methods mostly

rely on a common but implicit assumptions: a moderate list of tags is prescribed in advance and these tags generally describe an entire image or a dominant part thereof. Such techniques often face serious difficulties when used in real-world settings, where this assumption is far from the reality. Our primary goal is to develop a new image tagging framework that can meet the demand of real-world applications by moving beyond this limitation.

Towards this goal, a significant challenge that we face is the issue of *spatially local association*. In real-world practice, an image is often attached with multiple tags – some describing the overall scene while others referring to local regions, as shown in Figure 3.1. To tackle this problem, we propose a new strategy called *Scaled View Integration (SVL)*, which explicitly models the associations between tags and local regions, while allowing feature representations across locations and scales to be jointly learned.

Another real-world challenge consists in the *large tag space*. Many discriminative models [26] require each sample to be compared with *all* the tags in every update, leading to the complexity that increases linearly as the tag space grows and thus limited scalability. However, this is generally not necessary – our study shows that comparing each sample over a subset of tags usually provides sufficient contrastive information for learning an effective representation, especially when the tag space is large. We thus formulate a novel learning objective called *Con-*

trastive Bundle Loss (CBL), where each loss term is defined on a group of samples combined with a subset of tags, instead of individual samples over the entire tag space. This formulation improves the discriminative performance by contrasting across both tags and samples, while maintaining strong scalability – each iteration has a constant complexity regardless of the tag space.

Overall, the major contribution of this work is an image tagging framework that tackles two key challenges in real-world applications, namely *spatially local association* and *large tag space*, through the combination of *Scaled View Integration* and *Contrastive Bundle Loss*. Experiments on both NUS-Wide [10] and YFCC [85] showed that this framework achieves improved tagging accuracy as compared to the state-of-the-art, while providing superior scalability.

3.2 Related Works

Automatic image tagging has been an active research topic in computer vision [56]. It's been generally treated as a multi-label classification problem [2, 6, 28, 48] since machine learning was introduced to this task. Simple strategies such as neighbor voting [55] has also been widely used in practice to solve the problem The recent success of Convolutional Neural Network (CNN) [42] motivated the use of deep models for image tagging [26], which yields considerable performance improvemen-

t. Besides the evolution of underlying visual models, another stream of efforts towards improved tagging is the study on utilizing metadata [39, 32, 8, 98, 59].

As mentioned, the proposed framework comprises two complementary components: *Scaled-View Integration* and *Contrastive Bundle loss*. Below, we briefly review the relations between these components and existing work.

The *Scaled-View Integration* scheme is partly inspired by Spatial Pyramid Matching (SPM) [46], a classical technique in visual recognition, and aims to combine the local sensitivity of SPM with the power of CNN to tackle the issue of spatial local association. Note that there have been existing attempts [30, 25, 27] to integrate SPM and CNN, so as to exploit information across scales and locations. For example, SPPNet [30] proposes an computational efficient approach, called spatial pyramid pooling, to pool the feature maps extracted from one image over local windows of different scales. It is important to note that the pyramid pooling was considered *after* CNN features have already extracted. However, with SVI, we take into account the effect of scaling from the ground up, and jointly learn the convolutional coefficients across scales via an end-to-end learning scheme.

The *Contrastive Bundle loss* provides a flexible formulation on which losses can be defined over multiple samples in runtime. This relates to several other formulations for discriminative learning, including the pairwise and triplet contrastive losses

that have been widely used in face verification [83, 74] and image retrieval [92, 7, 68], as well as the siamese network [9]. These formulations can be considered as special cases of the proposed formulation, where the numbers of samples within each bundle and the choices of loss functions are subject to certain settings. In this work, improved versions of the softmax loss [3] and the weighted approximate ranking loss (WARP) [99] are implemented on this formulation, which promote contrasting across both classes and samples.

While the proposed components are partly inspired by previous work, they are motivated differently and use new strategies to break the limitations of existing work. More importantly, the two techniques, focusing respectively on visual analysis and learning, together constitute a unified framework to tackle the challenges in real-world image tagging that none of the existing work address at the same time, consequently bringing the state-of-the-art to a new level.

3.3 Combining Scale, Locations, and Categories in Learning Process

Generally, image tagging can be considered as a task of assigning a set of semantically relevant tags to a given image. Unlike image classification where each image belongs *exclusively* to one class, image tagging allows each image to be associated with multiple tags.

In spite of the substantial progress over the past decades, automatic tagging in real-world applications remain a challenging task. Particularly, several issues in real-world practice have not been sufficiently explored in previous study, such as the association between tags and local regions and the scalability with a large tag space. This work introduces two complementary techniques, *Scaled View Integration* and *Contrastive Bundle Loss*, to tackle these problems.

3.3.1 Scaled-View Integration

Visual recognition is crucial to image tagging. In recent years, the use of Convolutional Neural Networks (CNN) has led to remarkable improvement in recognition performance [42]. Whereas state-of-the-art CNNs, *e.g.* those pre-trained on ImageNet [42], can effectively recognize dominant objects at the center, it remains very challenging to handle variations in location and scale.

In real-world image tagging, an image can have multiple tags, some describing the whole scene while others referring to smaller local regions. This issue, which we call *local association*, posts a significant challenge. In particular, it requires a tagging system to robustly recognize visual patterns at different locations and of different scales.

A natural way to tackle this issue is to use a spatial pyramid [46]. For example, one can adapt the SPPNet [30] to image tagging by pooling the outputs from convolution layers. It is

important to note that SPPNet derives the feature representation at the input scale, and performs the pyramid-based pooling over the feature responses afterwards. In other words, the effect of scaling is not taken into account when the CNN features are extracted. This may not be an important issue if the CNN features are stable against scale changes, however, it is not the case as observed in previous work [27, 25].

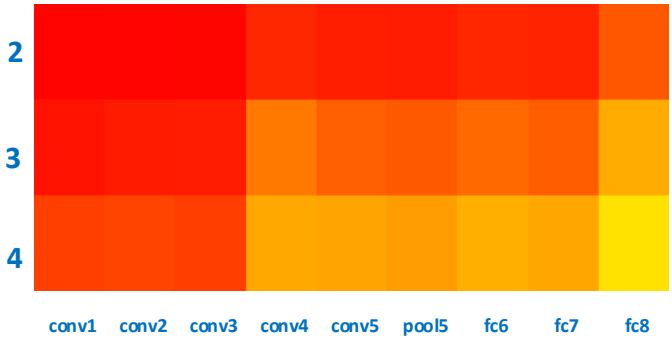


Figure 3.2: A heat map demonstrating the impact of the input scale on the output CNN features. Each cell in this map indicates the cosine similarity between the outputs (of a specific layer) corresponding to certain input scale ($2x \sim 4x$) and that corresponding to the original scale. Darker colors reflect stronger similarities. It can be observed that the similarity drops significantly as the scales diversify.

Effect of scales. To study the effect of input scales, we conducted a simple quantitative experiment. In this experiment, we feed a CNN with inputs of different scales, ranging from one to four times of the original size, and investigate the stability of the output. We compare the outputs of different layers obtained with rescaled inputs with those obtained from the original in-

puts and measure the distances between them. Figure 3.2 shows that scale changes result in remarkable differences in the output. The differences also increase as we move from bottom layers to top layers. The findings from this study clearly suggests that *CNN features are sensitive to input scales*, and therefore pooling the feature responses as in SPPNet [30] is not an effective strategy to handle scale changes.

Integrate multi-scale views. We propose a new model architecture that can jointly handle multiple local regions of different scales, while maintaining high run-time efficiency. The pipeline consists of two stages. In the first stage, each input image is resized to multiple scales. The size of an image at the k-th scale is $kw \times kh$. The resized image can then be divided into $k \times k$ local regions, each providing a *"zoom-in" view* of a local region. These zoom-in views at different scales can then be stacked into a tensor that serves as the input to the CNN, as shown in Figure 3.3, thus allowing local regions of different scales to be jointly analyzed at the same time.

Specifically, let I be an input image resized to a standard size $h \times w$, and $V_{k,i}$ be the i-th local view at the k-th scale. Then $V_{k,i}$ covers a local region of size $(h/k) \times (w/k)$ in I that is resized to $h \times w$. With these views stacked into a tensor, a CNN network with weights $\mathbf{W}$ can be applied *jointly* to derive features from all these views, denoted by $f(V_{k,i}; \mathbf{W})$. We then combine the features of each scale by *average pooling*, concatenate the

pooled features of all scales, and finally obtain the *integrated* representation $f(I)$ as

$$f(I) = \left[f(V_{1,1}; \mathbf{W}), \ldots, \frac{1}{K^2} \sum_{i=1}^{K^2} f(V_{K,i}, \mathbf{W}) \right]. \tag{3.1}$$

Here, K is the number of scales used in the model.

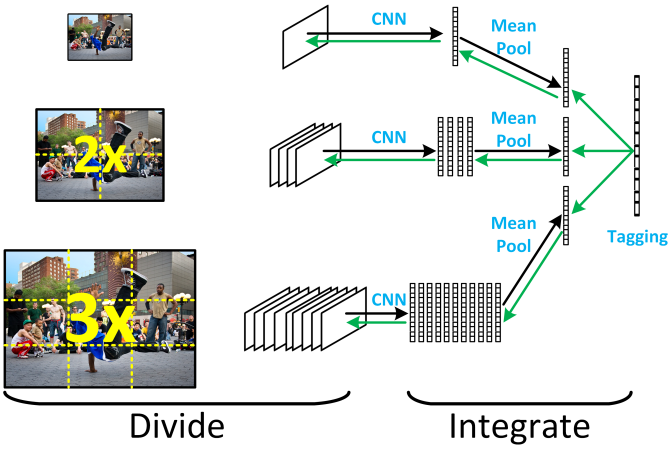
Discussion. The model architecture introduced above addresses the issue of *local association* by integrating the local views at different locations and of different scales. Unlike in SPPNet, this model explicitly handles the effect of scaling in the feature extraction stage, while allowing the CNN weights to be jointly optimized over all local parts in a single batch. We will see in the experiments that that this approach produces more effective representation for image tagging.

3.3.2 Contrastive Bundle Loss

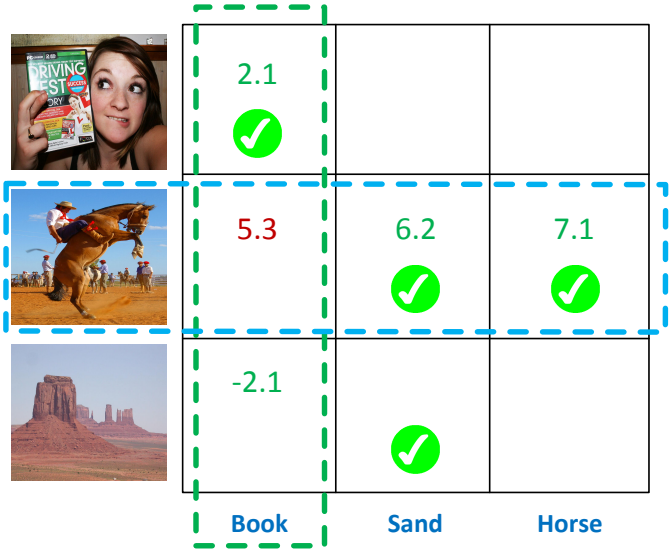
In supervised learning, a loss function is usually used to evaluate the feature representation $\mathbf{x}$ of a sample against the corresponding class label c. Particularly, the *Softmax loss* defined below is among the most widely used.

$$\mathcal{L}_{\text{softmax}}(\mathbf{x}, c) = \log \left(\sum_{k=1}^{M} \exp(\mathbf{w}_k^T \mathbf{x}) \right) - \sum_{k=1}^{M} \mathbf{y}^{(k)} \mathbf{w}_k^T \mathbf{x}. \tag{3.2}$$

Here, M is the number of classes, the $\mathbf{w}_k$ the weight vector for the class k, and $\mathbf{y}$ the class indicator with $\mathbf{y}^{(k)} = \mathbb{I}(k = c)$. The effectiveness of the softmax loss has been repeatedly proven in



(a) Scale-View Integration



(b) Bundle Loss

Figure 3.3: The framework of our proposed approach. In the training, the images are inputed in minibatches. One image will be going through the *scale-view integration* process illustrated in (a) to be transformed into a feature vector. Then our contrastive bundles are sampled on the minibatch and produce loss values and supervision signals, as shown in (b). The system can be learned ene-to-end with minibatch SGD. Flow of gradients during back-propagation is marked with green arrows in (a).

previous work. A key rationale underlying this loss function is *contrasting*, that is, to encourage high responses to the correct predictions, while suppressing the responses to others.

However, it is worth noting that such loss functions generally require every training sample to be evaluated against all classes. Hence, the computational complexity of the loss would increase linearly as the number of classes M increases, which would lead to serious difficulties in large-scale applications, *e.g.* web-scale services, where M can be extraordinary.

Bundles. In response to this challenge, we propose a novel formulation called K-*bundle*. Formally, a K-*bundle*, denoted by $B(\mathcal{D}, \mathcal{C})$, is a small group of samples combined with a small subset of distinct tags. Here, $\mathcal{D}$ is a small group of K samples and $\mathcal{C}$ is a subset of K tag classes. Each sample in a bundle comprises a feature vector $\mathbf{x}_i$ and a *restricted tag indicator* $\mathbf{y}_i \in \{0,1\}^K$. Particularly, $\mathbf{y}_i^{(j)}$ indicates whether $\mathbf{x}_i$ is relevant to the j-th tag in $\mathcal{C}$.

Compared to the conventional *each-sample-vs-all-classes* paradigm, this bundle formulation has two key advantages: (1) *Strong scalability.* The design parameter K is independent of the number of samples in the training set or the number of classes in the tag space. Our experiments show that, a relatively small K, say 32 to 128, is enough for a large tag space. Consequently, algorithms that use bundles as units for learning can maintain a stable complexity even when the number of tag classes increases

substantially. (2) *High flexibility.* This formulation allows various objective functions to be formulated thereon. Particularly, by grouping multiple samples, it allows new loss functions to be formulated over groups of samples, and promote contrast among them. Below, we present two loss functions defined on a bundle.

Restricted Dual-Softmax Loss. As mentioned, the *softmax loss* has been proven to be effective in discriminative learning. On top of a bundle B, we extend the softmax loss to encourage contrast among both samples and tag classes:

$$\mathcal{L}_{RDS}(B) \triangleq -\left(\frac{1}{K}\sum_{i=1}^{K}\frac{1}{t_i^+}\sum_{j=1}^{K}\mathbf{y}_i^{(j)}\log p_i^{(j)} + \frac{1}{K}\sum_{j=1}^{K}\frac{1}{s_j^+}\sum_{i=1}^{K}\mathbf{y}_i^{(j)}\log q_j^{(i)}\right). \quad (3.3)$$

Here, i and j are respectively the indexes of samples and tags *within the bundle* B, t_i^+ the number of tags in B that are associated with the i-th sample, and s_j^+ the number of samples in B that are relevant to the j-th tag. We explicitly enforce that each sample must be associated with at least one tag and each tag must be associated with at least one sample when constructing a bundle. Hence, t_i^+ and s_j^+ are always positive. In addition, $p_i^{(j)}$ and $q_j^{(i)}$ are defined as:

$$p_i^{(j)} = \frac{e^{\mathbf{w}_j^T\mathbf{x}_i}}{\sum_{j'=1}^{K} e^{\mathbf{w}_{j'}^T\mathbf{x}_i}}, \quad \text{and} \quad q_j^{(i)} = \frac{e^{\mathbf{w}_j^T\mathbf{x}_i}}{\sum_{i'=1}^{K} e^{\mathbf{w}_j^T\mathbf{x}_{i'}}}. \quad (3.4)$$

With this definition, the loss in Eq.(3.3) can be written as

$$\sum_{i=1}^{K}\log\left(\sum_{j=1}^{K}e^{\mathbf{w}_j^T\mathbf{x}_i}\right) + \sum_{j=1}^{K}\log\left(\sum_{i=1}^{K}e^{\mathbf{w}_j^T\mathbf{x}_i}\right) - \sum_{i=1}^{K}\sum_{j=1}^{K}\mathbf{y}_i^{(j)}\left(\frac{1}{t_i^+}+\frac{1}{s_j^+}\right)\mathbf{w}_j^T\mathbf{x}_i. \quad (3.5)$$

This function operates across samples and classes, while restricts the focus to only a subset of K classes. Hence, we call it *Restricted Dual-Softmax Loss.*

Restricted Dual Ranking Loss. The *weighted average ranking loss (WARP)* [99] is another important loss function for discriminative learning, defined as

$$\mathcal{L}_{WARP}(\mathbf{x}) = \frac{1}{|\mathcal{T}^+|} \sum_{j \in \mathcal{T}^+} \sum_{j' \in \mathcal{T}^-} \omega_j \max\left(0, 1 - \mathbf{w}_j^T \mathbf{x} + \mathbf{w}_{j'}^T \mathbf{x}\right). \tag{3.6}$$

Here, $\mathcal{T}^+$ and $\mathcal{T}^-$ are the sets of all positive and negative tags *w.r.t.*to the sample $\mathbf{x}$; ω_j is a weight assigned to each positive tag. The weight ω_j, usually in the form of

$$\omega_j = \frac{1}{r_j} \sum_{i=1}^{r_j} \frac{1}{i}, \tag{3.7}$$

adjusts the contributions of different estimated rank r to the overall loss. Note that in [26] the normalizing term $1/r_j$ is omitted, which we found empirically causes instability in learning. Similar to SVM, this loss encourages large margin between classes, and has been shown to be very effective in certain tasks.

Again, this function would face scalability issues when used with a large tag space, as it requires the comparison between all pairs of positive-negative tags for each sample. To tackle this, we formulate a restricted version over a bundle B as below, called *Restricted Dual Ranking Loss*.

$$\begin{aligned}\mathcal{L}_{RDR}(B) = &\frac{1}{K} \sum_{i=1}^{K} \frac{1}{t_i^+} \sum_{j \in \mathcal{T}_i^+} \sum_{j' \in \mathcal{T}_i^-} \omega_j \max\left(0, 1 - \mathbf{w}_j^T \mathbf{x}_i + \mathbf{w}_{j'}^T \mathbf{x}_i\right) \\ &+ \frac{1}{K} \sum_{j=1}^{K} \frac{1}{s_j^+} \sum_{i \in \mathcal{S}_j^+} \sum_{i' \in \mathcal{S}_j^-} \omega_i' \max\left(0, 1 - \mathbf{w}_j^T \mathbf{x}_i + \mathbf{w}_j^T \mathbf{x}_{i'}\right).\end{aligned} \tag{3.8}$$

Here, $\mathcal{T}_i^+$ and $\mathcal{T}_i^-$ are the sets of relevant and irrelevant tags for sample i within the bundle, $\mathcal{S}_j^+$ and $\mathcal{S}_j^-$ are the sets of positive and negative samples for tag j within the bundle. The sample-wise weights ω_i' can be estimated in the same way as estimating ω_j, except that it is along the direction of samples.

Summary. Both new loss formulations differ from their original counterparts in three aspects: (1) The complexity depends only on the design parameter K instead of the tag space size. Our experiments show that a moderate value of K is enough to handle a large tag space. (2) With multiple samples bundled together, they can encourage contrast among samples in addition to tag classes, thus further enhancing the discriminative power. (3) Both loss functions allow multiple tags to be associated to a sample.

3.4 Experiments

We tested our approaches on large datasets, and compared them with other approaches on a number of performance metrics. We investigated not only the tagging performance, but also the algorithm's scalability.

3.4.1 Experiment Settings

Data sets. We conducted tests on two public datasets, *NUS-WIDE* [10] and *YFCC100M* [85]. *NUS-WIDE* [10] is widely

used for evaluating image tagging methods. This dataset provides a list of URLs to images from Flickr. As of the time we started this project, 110,389 images remain downloadable. Each image is attached with zero to multiple ground-truth tags, chosen from a curated list of 81 distinct tags. With all those without ground-truth tags excluded, there remain 86,374 images, which we used in our experiments. Totally, there are 213,219 tags for these images, 2.47 for each on average. We divide these images into two disjoint halves, 80% for training and validation, and the other 20% for testing. The tag distribution in NUS-WIDE is highly imbalanced, the most frequent tags can be present hundreds of times more than the less frequent ones.

YFCC100M [85] was recently released by Yahoo! to promote large-scale vision research. It contains around 99.2 millions of images from Flickr, some of which come with user tags and other metadata. We construct a benchmark from YFCC100M for evaluating tagging performance as follows. First, from all associated user tags, we identify a list of 2000 most frequent ones, and then remove those irrelevant to semantic understanding, such as device brands (*e.g. "Canon"*) or places (*e.g. "New York"*). This results in the final list of 997 distinct tags. Based on these tags, we sample a subset of 397,435 relevant images, each associated with at least one of the tags in the chosen list. Over this subset, there are over 3 million tags, about 7.78 tags for each image on average. From these images, we selected 367,435 for training and validation and the other 30,000 for testing. Due to its

broader range of tags and closer relation with real-world photo sharing practice, we argue that it is more suitable for assessing the practical performance of a tagging method.

Methods for comparison. The baseline methods that we compared with are CNN models combined with three different loss functions: (1) *Sigmoid*: per-tag sigmoid cross entropy loss, which is equivalent to *per-tag logistic regression*. (2) *Softmax*: the loss usually used in image classification [42], (3) *WARP*: weighted approximate ranking loss, as presented in [26]. We also compare with *Neighbor* voting based on visual features, which is often used in image tagging practice. In this experiment, we tested neighbor voting with features learned with SVI equipped CNN for fair comparison.

For the proposed framework, we tested different configurations. Particularly, we studied the settings with and without *Scaled View Integration (SVI)*, as well as different choices of *Bundle Losses*, including *Restricted Dual Softmax Loss (RDSL)* and *Restricted Dual Ranking Loss (RDRL)*, in order to investigate their respective contributions to the tagging performance.

As mentioned, SVI is different from *Spatial Pyramid Pooling (SPP)* in an important aspect: SVI takes into account the scaling effect in feature extraction by extracting features jointly over multiple local views while SPP extracts features once over the input image. To demonstrate the impact of this difference to tagging performance, we also implemented SPPNet [30] as

well as an architecture that integrates SPPNet with SVI on top.

Note that all methods, including both the baselines and our proposed methods, share the same basic CNN architecture [42], which itself already delivers good baseline performance [72]. In our experiments, we focus on comparing how different techniques can further improve the performance.

Evaluation metrics. We use three widely used metrics to study the performance in different aspects. (1) *Top*-3 *Recall/Precision (Recall@3, Precision@3)*, (2) *Top*-5 *Recall/Precision (Recall@5, Precision@5)*, and (3) *Mean Average Precision (MAP)* [56]. Recall rates and precision scores measure the performance of a tagging system in recommending a fixed number of tags for each input image. The MAP score, on the other side, evaluates the framework's capacity in retrieving all relevant tags and rank them as high as possible. Considering the serious imbalance of tag distributions on both datasets, we follow the standard protocol and adopt a two-fold metrics setup: *per-image* average performance and *per-tag (class)* average performance. The former emphasizes more frequent tags, while the latter considers each tag class as equally important. A well-rounded model should perform well in most of the metrics, in both forms.

3.4.2 Analysis of Results

The testing performances obtained on NUS-WIDE and YFC-C100M are respectively shown in Table 3.2 and 3.3. Both *per-*

image and *per-tag* performances are presented. Overall, the proposed methods consistently outperform the baselines on different metrics, clearly demonstrating its effectiveness. It is particularly worth noting that the settings that incorporate both SVI and bundle losses, *i.e. RDSL+SVI* and *RDRL+SVI* win 17 out of 20 metrics (on both datasets) by a reasonable margin. This shows that these two techniques are complementary and therefore the combination of them results in the best performance. Below, we will analyze the results from different aspects.

Scaled-view integration. SVI aims to enhance the model's capability of handling local association and scale variances. While the baseline methods with Softmax or WARP losses are already quite strong, we still observe improvements, especially on recall rates, when SVI is incorporated. This is because the scaled local views help to recover those tags referring to local regions or objects.

Table 3.1: Comparison of models without SPP, with SPP and with SPP+SVI. Performances are measured by MAP in per-image and per-tag bases.

	NUS-WIDE		YFCC	
	MAP_I	MAP_T	MAP_I	MAP_T
RDRL	.7501	.6016	.2690	.2154
RDRL+SPP [30]	.7530	.6045	.2834	.2238
RDRL+SPP [30]+SVI	.7625	.6278	.2993	.2472

Spatial pyramid pooling (SPP) is another reasonable way to incorporate information from multiple resolutions. It was demon-

strated in the ImageNet challenge that it can improve recognition performance. However, SVI and SPP are essentially different. *SVI* focuses on the processing of inputs *before* feature extraction, while *SPP* concerns about the pooling thereafter. Therefore they can be seamlessly incorporated with each other.

We specifically conducted a set of experiments to study how SVI and SPP work together. In particular, we consider three settings: (1) AlexNet + RDRL loss, (2) AlexNet replaced by SPPNet, and (3) SPPNet incorporated with SVI. The results are shown in Table 3.1. First, the results confirm that SPP can lead to moderate improvement. Then, the incorporation of SVI can bring the performance to the next level. This suggests that SVI can consistently boost the tagging performance over a given base network, and more importantly, it can seamlessly work with other techniques like SPP.

Contrastive bundle loss. From the experimental results, we also observed performance gains due to the use of *contrastive bundle loss*, *e.g. Softmax* vs. *RDSL*, and *WARP* vs. *RDRL*. Particularly, we can see considerable increase in *per-tag precision* in most of the time. This shows that *bundle loss* can guide the model's attention towards discriminative visual patterns, while alleviating the undesirable impact of imbalanced tag distribution.

Among all the loss functions compared in the experiments, RDRL is often the best one. This shows that ranking based loss functions are more suited for tagging task, while the *bundling*

can further improve its performance, by contrasting across both *samples* and *tags*.

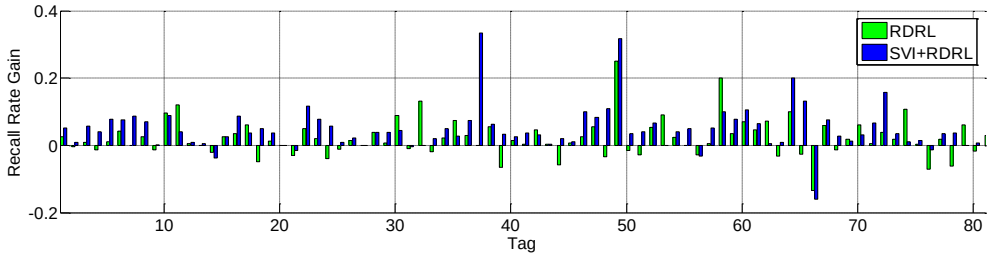


Figure 3.4: Per-tag recall rate gains over the baseline on NUS-WIDE with $k = 3$. Tags are arranged in the revert order of their frequencies.

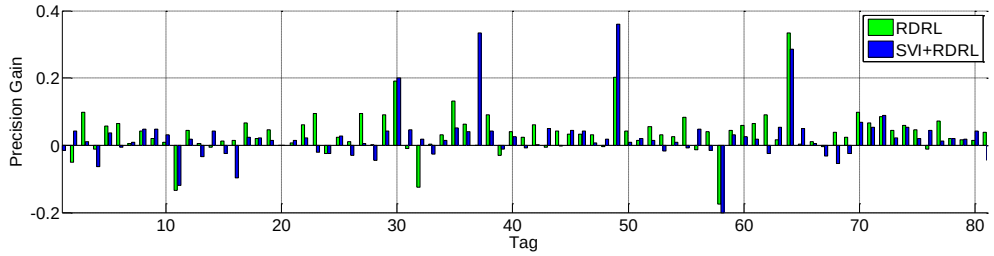


Figure 3.5: Per-tag precision gains over the baseline on NUS-WIDE with $k = 3$. Tags are arranged in the revert order of their frequencies.

Per-tag analysis. To obtain further insight, we visualize the per-tag recall and precision scores on the NUS-WIDE dataset in Figure 3.4 and Figure 3.5. In both figures, the performances are reported as the improvements relative to the baseline. We can see that *RDRL+SVI* improves the recall for 70 tag classes out of 81, and improves the precision for 56 classes. Clearly, most of the classes benefit from the contributions of *SVI* and *RDRL*. In other words, the overall performance gain is based on universal improvement instead of catering to a small number of frequent

classes.

Performances on different datasets. It is worth noting that the performances obtained on YFCC100M are clearly inferior to those obtained on NUS-WIDE. This is ascribed to a variety of reasons, which particularly include the nosiness of user tags and the substantially increased tag space. Both would add to the difficulties of accurate tagging.

On this more challenging data set, we can see even more obvious improvement when SVI and bundle losses are incorporated. For example, RDRL+SVI brings the MAP from 0.175 to 0.234 as compared to the baseline using Softmax loss. This clearly shows the significant role of the proposed techniques in tackling real-world challenges.

Scalability of the bundle formulation. We verified the scalability of the bundle formulation via experiments. Scalability requires that when scaling to large tag space, it can still achieve good performance, which is demonstrated in YFCC experiments, while maintaining reasonable runtime cost. We can verify the second requirement by examining the computation cost of a loss function and its bundle form. Using the settings in our experiments, we calculated the empirical computation costs of the WARP loss, which yield the best accuracy among non-bundle loss function, and its bundle form, the RDRL loss, as shown in Fig. 3.6. To better understand the growth of computation cost as the tag

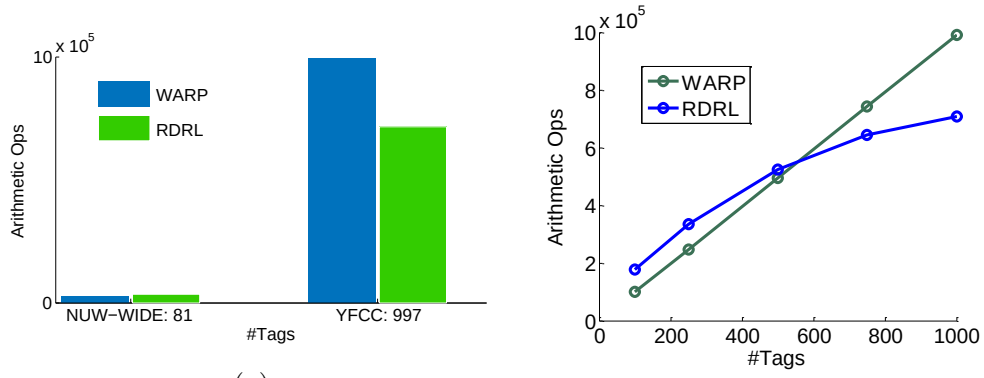


Figure 3.6: Computation cost analysis of WARP loss and RDRL loss. (a) shows the loss computation cost measured in arithmetic ops on NUW-WIDE and YFCC dataset. (b) shows an synthetic study of loss computation costs on YFCC dataset with reduced tag spaces.

space varies, we gradually reduce the tag space of YFCC dataset from 1000 to 100. Then we measure the loss computation costs of WARP and RDRL. For our RDRL, we use the bundle size that we empirically found to obtain reasonable tagging accuracies. We can see that as the tag spaces grow, the runtime cost grows *sublinearly* w.r.t the tag space size, demonstrating higher scalability.

3.5 Discussion and Summary

In this part of thesis work, an new framework for image tagging is proposed aiming to tackle two key challenges in real-world applications, namely *local association* and *large tag space*. It fol-

low the strategy of combining multiple aspect of data, which are scale, location, and conception categories in this case. Experimental results show that through scaled view integration and contrastive bundle loss, the proposed framework outperformed state-of-the-art on two large datasets, and demonstrated superior scalability in face of a growing tag space.

Table 3.2: Experimental results on the NUS-WIDE dataset dataset. Higher numbers are preferred in all columns. The upper half of the table reflects the per-image version of the evaluation metrics. The lower half deals with the per-tag ones. The best performing entries in each column are marked with bold fonts. Here "RDSL" and "RDRLrefer to models trained with restricted dual softmax losses and restricted dual ranking losses. "SVI" denotes that the model is trained with scaled-view integration.

Methods	NUS-WIDE - Per-Image				
	MAP_I	Rec_I@3	$Prec_I$@3	Rec_I@5	$Prec_I$@5
Random	0.002	0.04	0.03	0.07	0.03
Sigmoid	0.730	65.50	53.59	80.69	39.66
Softmax [26]	0.736	65.50	53.58	80.80	39.73
WARP [26]	0.750	66.05	54.03	81.43	40.03
RDSL	0.740	65.86	53.86	81.33	39.96
RDRL	0.750	66.23	54.17	81.65	40.11
Neighbor [55]+SVI	0.701	60.31	50.14	79.48	39.27
Softmax [26]+SVI	0.743	65.72	53.95	81.46	40.18
WARP [26]+SVI	0.759	66.76	**54.81**	82.25	**40.57**
RDSL+SVI	0.742	65.87	53.83	81.78	40.18
RDRL+SVI	**0.763**	**67.46**	54.73	**82.71**	40.44
Methods	NUS-WIDE - Per-Tag				
	MAP_T	Rec_T@3	$Prec_T$@3	Rec_T@5	$Prec_T$@5
Random	0.002	0.04	0.03	0.07	0.03
Sigmoid	0.553	44.23	37.22	53.21	25.18
Softmax [26]	0.568	45.81	38.31	57.33	25.37
WARP [26]	0.577	46.31	40.16	62.09	27.03
RDSL	0.603	46.10	40.11	62.13	27.93
RDRL	0.602	45.85	40.27	62.21	28.09
Neighbor [55]+SVI	0.573	42.82	37.15	56.61	24.19
Softmax [26]+SVI	0.621	47.30	40.30	62.30	28.10
WARP [26]+SVI	0.613	46.93	40.35	62.38	27.59
RDSL+SVI	**0.636**	**49.72**	40.96	64.46	**28.38**
RDRL+SVI	0.618	49.55	**41.30**	**66.50**	27.80

Table 3.3: Experimental results on the YFCC dataset dataset. Higher numbers are preferred in all columns. The upper half of the table reflects the per-image version of the evaluation metrics. The lower half deals with the per-tag ones. The best performing entries in each column are marked with bold fonts. Here "RDSL" and "RDRLrefer to models trained with restricted dual softmax losses and restricted dual ranking losses. "SVI" denotes that the model is trained with scaled-view integration.

Methods	YFCC - Per-Image				
	MAP_I	Rec_I@3	$Prec_I$@3	Rec_I@5	$Prec_I$@5
Random	0.000	0.01	0.00	0.00	0.00
Sigmoid	0.170	15.66	15.79	20.55	12.46
Softmax [26]	0.260	19.78	19.96	25.64	15.56
WARP [26]	0.265	20.47	20.66	26.58	16.12
RDSL	0.262	20.04	20.23	26.25	15.94
RDRL	0.269	21.08	21.32	27.81	16.90
Neighbor [55]+SVI	0.162	18.43	12.92	18.71	14.31
Softmax [26]+SVI	0.268	20.86	21.12	26.99	16.42
WARP [26]+SVI	0.274	20.95	21.22	27.21	16.55
RDSL+SVI	0.269	20.37	20.63	26.72	16.27
RDRL+SVI	**0.283**	**22.25**	**22.55**	**28.91**	**17.61**
Methods	YFCC - Per-Tag				
	MAP_T	Rec_T@3	$Prec_T$@3	Rec_T@5	$Prec_T$@5
Random	0.002	0.01	0.00	0.00	0.00
Sigmoid	0.103	6.15	12.30	8.31	10.11
Softmax [26]	0.175	9.79	19.53	12.86	16.11
WARP [26]	0.194	10.44	20.90	14.03	16.70
RDSL	0.183	9.39	21.30	13.10	16.70
RDRL	0.215	10.79	**24.03**	14.91	17.62
Neighbor [55]+SVI	0.154	10.33	19.43	12.67	11.36
Softmax [26]+SVI	0.188	11.03	20.20	14.20	15.00
WARP [26]+SVI	0.194	11.12	21.14	14.75	16.62
RDSL+SVI	0.188	9.70	22.90	13.16	17.65
RDRL+SVI	**0.234**	**12.51**	23.97	**16.86**	**17.72**

Chapter 4

Recognize Human Activity from Videos

Although we have achieved exciting results on still images. There is still a large gap between our computer vision systems and a truly practical one. As the visual information we perceive every moment are changing, it becomes necessary for the system to understanding videos. Human activity classification is a widely researched topic in this scenario. However, analysis demonstrates that it is still far from optimal even with the recent development of deep learning techniques. And the intrinsic property of this problem calls for a framework that can combine information of carried in appearance, motion, and more importantly, long-term temporal structures. Int this sense, we propose the temporal segment networks framework to enhance the modeling capacity of original convolutional neural networks from both spatial and temporal dimensions. Our approach obtains the state-the-of-art performance on the datasets of HMDB51 (69.4%) and UCF101

(94.2%).

4.1 From Images to Videos

Video-based action recognition has drawn a significant amount of attention from the academic community [76, 89, 93, 61, 96, 21], because of its applications in many areas like security and behavior analysis. Moving from images to videos, the new inputs of video streams bring about a new dimension to explore, time. For a video, the dynamics depicting the change of appearances over time thus become an indispensable factor in deciding its underlying action class. To achieve reasonable performance, an action recognition system must rely on both appearances and dynamics in the videos. However, extracting these information is non-trivial due to the factors such as scale variations, view point changes, and camera motions. Thus it becomes crucial to design effective representations that can deal with these challenges while preserve categorical information of action classes. Recently, Convolution Networks (ConvNets) [47] have witnessed great success in classifying images of objects, scenes, and complex events [43, 77, 82, 102]. ConvNets have also been introduced to solve the problem of video-based action recognition [40, 76, 86, 109]. Deep ConvNets come with great modeling capacity and are capable of learning discriminative representation from raw visual data with the help of large-scale supervised datasets. However, unlike image classification, end-

to-end deep ConvNets remain unable to demonstrate significant advantage over traditional hand-crafted features for video-based action recognition.

We analyze that there are two major obstacles to successfully applying ConvNets to video-based action recognition. *First*, a video is a kind of spatial-temporal signal. Long-range temporal structure plays an important role in understanding the dynamics in action videos [64, 20, 94, 19]. But current state-of-the-art ConvNet frameworks [76, 86] usually focus on modeling appearances and short-term motions, thus lacking the capacity to incorporate long-range temporal structure. Recently there are a few efforts [88, 61, 15] trying to deal with this problem. These methods, although working with longer video clips, are mostly based on dense temporal sampling strategies which lead to excessive computational cost. This inhibits them to take the entire video as input and incurs the risk of missing important information outside the sampled sequence. *Second*, in practice, training deep ConvNets requires a large volume of training samples to achieve optimal performance. Due to the difficulty in data collection and annotation, current action recognition datasets (e.g. UCF101 [78], HMDB51 [44]) possess quite limited training samples compared with those for image classification (e.g. ImageNet [11]). This can lead to severe over-fitting problems for the learned models.

In this thesis work, we aim to study these two problems: *1) how to design an effective and efficient video-level framework for*

learning ConvNets from video data; 2) how to learn the ConvNet models given limited training samples. In particular, we build our method on top of the successful two-stream architecture [76] while dealing with the problems mentioned above. In terms of temporal structure modeling, a key observation is the high redundancy between consecutive frames in a video. Due to this redundancy, densely sampling the frames, with the aim of modeling the long-range temporal structure, can cause unnecessary computational cost by repeatedly evaluating the ConvNets on very similar frames. Thus a sparse temporal sampling strategy will be more favorable in this case. Motivated by this observation, we develop a video-level framework, called *temporal segment network* (TSN). It adopts a sparse temporal sampling strategy, where it samples multiple short snippets at different temporal locations from the entire video. The sampling strategy is designed to make sampled snippets distribute uniformly along the temporal dimension. A segmental structure is employed to aggregate information from the sampled snippets. In this sense, temporal segment networks are capable of modeling long-range temporal structure over the whole video. In addition, this sparse temporal sampling strategy guarantees that the training computational cost of the temporal segment framework is independent of durations of the training videos. This makes it possible to perform end-to-end learning on the entire videos with reasonable computational cost.

To fully reveal the potential of temporal segment network frame-

work, we propose to use very deep ConvNet architectures [34, 77] introduced recently. However, applying these architectures to the action recognition datasets is troubled by the limited number of training samples. To this end, we study a series of good practices in training the ConvNets models on video data, including 1) cross-modality pre-training; 2) regularization; 3) enhanced data augmentation. Meanwhile, to fully utilize visual content from videos, we empirically study four types of input modalities to two-stream ConvNets, namely a single RGB image, stacked RGB difference, stacked optical flow field, and stacked warped optical flow field.

We perform experiments on two challenging action recognition datasets, namely UCF101 [78] and HMDB51 [44], to verify the effectiveness of our method. In experiments, models learned using the temporal segment network strongly outperform the state-of-the-art on these two challenging action recognition datasets. We also visualize the our learned two-stream models and try to provide some insights for future action recognition research.

4.2 Related Works

Action recognition has been extensively studied in past few years [89, 23, 69, 22, 19]. In this section we review previous works related to ours on two aspects: (1) convolutional networks for action recognition, (2) modeling temporal structure.

Convolutional Networks for Action Recognition. Several works have been trying to design effective ConvNet architectures for action recognition in videos [40, 76, 86, 35, 80]. Karpathy *et al.* [40] tested ConvNets with deep structures on a large dataset (Sports-1M). Simonyan *et al.* [76] designed two-stream ConvNets containing spatial and temporal net by exploiting ImageNet dataset for pre-training and calculating optical flow to explicitly capture motion information. Tran *et al.* [86] explored 3D ConvNets [35] on the realistic and large-scale video datasets, where they tried to learn both appearance and motion features with 3D convolution operations. Sun *et al.* [80] proposed a factorized spatio-temporal ConvNets and exploited different ways to decompose 3D convolutional kernels. Recently, several works focused on modeling long-range temporal structure with ConvNets [61, 88, 15]. However, these methods directly operated on a longer continuous video streams. This means heavier computational cost for longer video streams. So these methods usually process sequences of fixed lengths ranging from 64 to 120 frames. It is non-trivial for these methods to learn from entire video due to their limited temporal coverage. Our method differs from these end-to-end deep ConvNets by its novel adoption of a sparse temporal sampling strategy, which enables efficient learning using the entire videos without the limitation of sequence length.

Temporal Structure Modeling. Many research works have been devoted to model the temporal structure for action

recognition [64, 20, 94, 70, 95, 19]. Gaidon *et al.* [20] annotated each atomic action for each video and proposed Actom Sequence Model (ASM) for action detection. Niebles *et al.* [64] proposed to use latent variables to model the temporal decomposition of complex actions, and resorted to the Latent SVM [18] to learn the model parameters in an iterative approach. Wang *et al.* [94] and Pirsiavash *et al.* [70] extended the temporal decomposition of complex action into a hierarchical manner by using Latent Hierarchical Model (LHM) and Segmental Grammar Model (SGM), respectively. Wang *et al.* [95] designed a sequential skeleton model (SSM) to capture the relations among dynamic-poselets, and performed spatio-temporal action detection. Fernando [19] modeled the temporal evolution of BoVW representations for action recognition. These methods, however, are still not able to assemble an end-to-end learning scheme for modeling the temporal structure. The proposed temporal segment network, while also emphasizing this principle, is the first framework for end-to-end temporal structure modeling on the entire videos.

4.3 Action Recognition with Temporal Segment Networks

In this section, we give detailed descriptions of performing action recognition with temporal segment networks. Specifically, we first introduce the basic concepts in the framework of temporal segment network. Then, we study the good practices in

learning two-stream ConvNets within the temporal segment network framework. Finally, we describe the testing details of the learned two-stream ConvNets.

4.3.1 Temporal Segment Networks

As we discussed in Sec. 4.1, an obvious problem of current two-stream ConvNets is their inability in modeling long-range temporal structure. This is mainly due to their limited access to temporal context as they are designed to operate only on a single frame (spatial networks) or a single stack of frames in a short snippet (temporal network). However, complex actions, such as sports action, comprise multiple stages spanning over a relatively longer time. It would be quite a loss failing to utilize long-range temporal structures in these actions into ConvNet training. To tackle this issue, as shown in Figure 4.1, we propose temporal segment network, a video-level framework, to enable to model dynamics throughout the whole video.

Specifically, our proposed temporal segment network framework, aiming to utilize the visual information of entire videos to perform video-level prediction, is also composed of spatial stream ConvNets and temporal stream ConvNets. Instead of working on single frames or frame stacks, temporal segment networks operate on a sequence of short snippets sparsely sampled from the entire video. Each snippet in this sequence will produce its own preliminary prediction of the action classes. Then

a consensus among the snippets is attained as the video-level prediction. In the learning process, the loss values of video-level predictions, other than those of snippet-level predictions which were used in two-stream ConvNets, are optimized by iteratively updating the model parameters.

Formally, given a video V, we divide it into K segments $\{S_1, S_2, \cdots, S_K\}$ of equal durations. Then, the temporal segment network models a sequence of snippets as follows:

$$\mathrm{TSN}(T_1, T_2, \cdots, T_K) = \mathcal{H}(\mathcal{G}(\mathcal{F}(T_1; \mathbf{W}), \mathcal{F}(T_2; \mathbf{W}), \cdots, \mathcal{F}(T_K; \mathbf{W}))). \quad (4.1)$$

Here $(T_1, T_2, \cdots, T_K)$ is a sequence of snippets. Each snippet T_k is randomly sampled from its corresponding segment S_k. $\mathcal{F}(T_k; \mathbf{W})$ is the function representing a ConvNet with parameters $\mathbf{W}$ which operates on the short snippet T_k and produces class scores for all the classes. The segmental consensus function $\mathcal{G}$ combines the outputs from different short snippets to obtain a consensus of class hypothesis among them. Based on this consensus, the prediction function $\mathcal{H}$ predicts the probability of each action class for the whole video. Here we choose the widely used Softmax function for $\mathcal{H}$. Combining with standard categorical cross-entropy loss, the final loss function regarding the segmental consensus $\mathbf{G} = \mathcal{G}(\mathcal{F}(T_1; \mathbf{W}), \mathcal{F}(T_2; \mathbf{W}), \cdots, \mathcal{F}(T_K; \mathbf{W}))$ is formed as

$$\mathcal{L}(y, \mathbf{G}) = -\sum_{i=1}^{C} y_i \left(G_i - \log \sum_{j=1}^{C} \exp G_j \right), \quad (4.2)$$

where C is the number of action classes and y_i the groundtruth

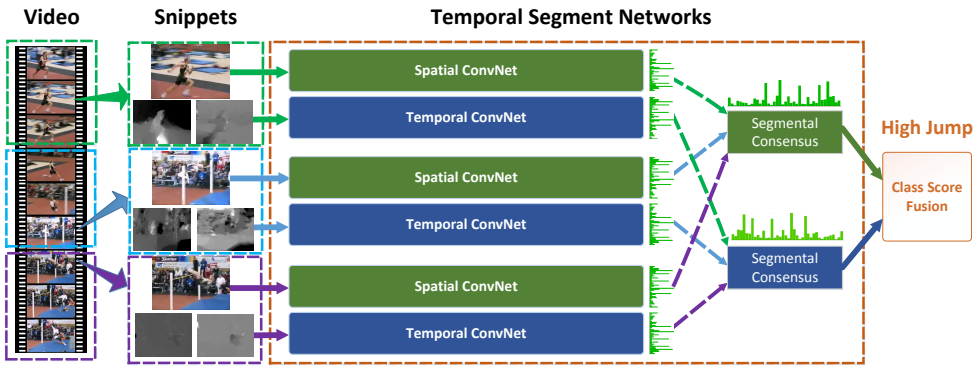


Figure 4.1: Temporal segment network: One input video is divided into K segments and a short snippet is randomly selected from each segment. The class scores of different snippets are fused by an the segmental consensus function to yield segmental consensus, which is a video-level prediction. Predictions from all modalities are then fused to produce the final prediction. ConvNets on all snippets share parameters.

label concerning class i. In experiments, the number of snippets K is set to 3 according to previous works on temporal modeling [20, 94]. The form of consensus function $\mathcal{G}$ remains an open question. Here we use the simplest form of $\mathcal{G}$, where $G_i = g(\mathcal{F}_i(T_1), \ldots, \mathcal{F}_i(T_K))$. It is therefore the class-independent aggregation from snippet-level predictions, depicted by the aggregation function g. We empirically evaluated several forms form aggregation function g including evenly averaging, maximum, and weighted averaging in our experiments. Among them, evenly averaging is used to report our final recognition accuracies.

This temporal segment network is differentiable or at least has subgradients, depending on the aggregation function g, depending on the choice of g. This allows us to utilize the multi-

ple snippets to jointly optimize the model parameters $\mathbf{W}$ with standard back-propagation algorithms. In the back-propagation process, the gradients of model parameters $\mathbf{W}$ with respect to the loss value $\mathcal{L}$ can be derived as

$$\frac{\partial \mathcal{L}(y, \mathbf{G})}{\partial \mathbf{W}} = \frac{\partial \mathcal{L}}{\partial \mathbf{G}} \sum_{k=1}^{K} \frac{\partial \mathcal{G}}{\partial \mathcal{F}(T_k)} \frac{\partial \mathcal{F}(T_k)}{\partial \mathbf{W}}, \tag{4.3}$$

where K is number of segments temporal segment network uses.

When we use a gradient-based optimization method, like stochastic gradient descent (SGD), to learn the model parameters, Eq. 4.3 guarantees that the parameter updates are utilizing the segmental consensus $\mathbf{G}$ derived from all snippet-level prediction. Optimized in this manner, temporal segment networkcan learn model parameters from the entire video rather than a short snippet. Meanwhile, by fixing K for all videos we assemble a sparse temporal sampling strategy, where the sampled snippets contain only a small portion of the frames. It drastically reduces the computational cost for evaluating ConvNets on the frames, compared with previous works using densely sampled frames [61, 88, 15].

4.3.2 Learning Temporal Segment Networks

temporal segment network provides a solid framework to perform video-level learning, but to achieve optimal performance, a few practical concerns have to be taken care of, for example the limited numberof training samples. To this end, we study a se-

ries of good practices in training deep ConvNets on video data, which are also directly applicable in learning temporal segment networks.

Network Architectures. Network architecture is an important factor in neural network design. Several works have verified that deeper structures improve object recognition performance [77, 82]. However, the original two-stream ConvNets [76] employed a relatively shallow network structure (ClarifaiNet [108]). In this work, we choose the Inception with Batch Normalization (BN-Inception) [34] as building block, due to its good balance between accuracy and efficiency. We adapt the original BN-Inception architecture to the design of two-stream ConvNets. Like in the original two-stream ConvNets [76], the spatial stream ConvNet operates on a single RGB images, and the temporal stream ConvNet takes a stack of consecutive optical flow fields as input.

Network Inputs. We are also interested in exploring more input modalities to enhance the discriminative power of temporal segment networks. Originally, the two-stream ConvNets used RGB images for the spatial stream and stacked optical flow fields for the temporal stream. Here, we propose to study two extra modalities, namely *RGB difference* and *warped optical flow fields*.

A single RGB image usually encodes static appearance at a specific time point and lacks the contextual information about previous and next frames. As shown in Figure 4.2, RGB differ-

ence between two consecutive frames describe the appearance change, which may correspond to the motion salient region. Inspired by [80], We experiment with adding stacked RGB difference as another input modality and investigate its performance in action recognition.

The temporal stream ConvNets take optical flow field as input and aim to capture the motion information. In realistic videos, however, there usually exists camera motion, and optical flow fields may not concentrate on the human action. As shown in Figure 4.2, a remarkable amount of horizontal movement is highlighted in the background due to the camera motion. Inspired by the work of improved dense trajectories [89], we propose to take warped optical flow fields as additional input modality. Following [89], we extract the warped optical flow by first estimating homography matrix and then compensating camera motion. As shown in Figure 4.2, the warped optical flow suppresses the background motion and makes motion concentrate on the actor.

Network Training. As the datasets for action recognition are relatively small, training deep ConvNets is challenged by the risk of over-fitting. To mitigate this problem, we design several strategies for training the ConvNets in temporal segment networks as follows.

Cross Modality Pre-training. Pre-training has turned out to be an effective way to initialize deep ConvNets when the target dataset does not have enough training samples [76]. As spa-

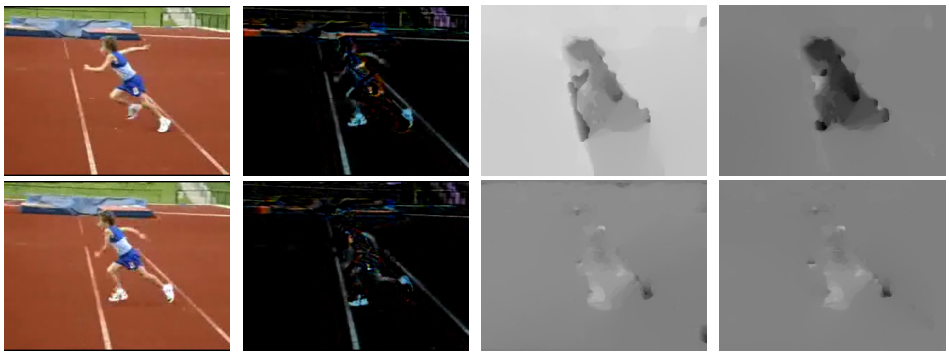


Figure 4.2: Examples of four types of input modality: RGB images, RGB difference, optical flow fields (x,y directions), and warped optical flow fields (x,y directions)

tial networks take RGB images as input, it is natural to exploit models trained on the ImageNet [11] as initialization. For other modalities such as optical flow field and RGB difference, they essentially capture different visual aspects of video data and their distributions are different from that of RGB images. We come up with a cross modality pre-training technique in which we utilize RGB models to initialize the temporal networks. First, we discretize optical flow fields into the interval from 0 to 255 by a linear transformation. This step makes the range of optical flow fields to be the same with RGB images. Then, we modify the weights of first convolution layer of RGB models to handle the input of optical flow fields. Specifically, we average the weights across the RGB channels and replicate this average by the channel number of temporal network input. This initialization method works pretty well for temporal networks and reduce the effect of over-fitting in experiments.

Regularization Techniques. Batch Normalization [34] is an important component to deal with the problem of covariate shift. In the learning process, batch normalization will estimate the activation mean and variance within each batch and use them to transform these activation values into a standard Gaussian distribution. This operation speeds up the convergence of training but also leads to over-fitting in the transferring process, due to the biased estimation of activation distributions from limited number of training samples. Therefore, after initialization with pre-trained models, we choose to freeze the mean and variance parameters of all Batch Normalization layers except the first one. As the distribution of optical flow is different from the RGB images, the activation value of first convolution layer will have a different distribution and we need to re-estimate the mean and variance accordingly. We call this strategy as **partial BN**. Meanwhile, we add a extra **dropout** layer after the global pooling layer in BN-Inception architecture to further reduce the effect of over-fitting. The dropout ratio is set as 0.8 for spatial stream ConvNets and 0.7 for temporal stream ConvNets.

Data Augmentation. Data augmentation can generate diverse training samples and prevent severe over-fitting. In the original two-stream ConvNets, random cropping and horizontal flipping are employed to augment training samples. We exploit two new data augmentation techniques: corner cropping and scale-jittering. In corner cropping technique, the extracted regions are only selected from the corners or the center of the

image to avoid implicitly focusing on the center area of a image. In multi-scale cropping technique, we adapt the scale jittering technique [77] used in ImageNet classification to action recognition. We present an efficient implementation of scale jittering. We fix the size of input image or optical flow fields as 256×340, and the width and height of cropped region are randomly selected from $\{256, 224, 192, 168\}$. Finally, these cropped regions will be resized to 224×224 for network training. In fact, this implementation not only contains scale jittering, but also involves aspect ratio jittering.

4.3.3 Testing Temporal Segment Networks

Finally, we present our testing method for temporal segment networks. Due to the fact that all snippet-level ConvNets share the model parameters in temporal segment networks , the learned models can perform frame-wise evaluation as normal ConvNets. This allows us to carry out fair comparison with models learned without the temporal segment network framework. Specifically, we follow the testing scheme of the original two-stream ConvNets [76], where we sample 25 RGB frames or optical flow stacks from the action videos. Meanwhile, we crop 4 corners and 1 center, and their horizontal flipping from the sampled frames to evaluate the ConvNets. For the fusion of spatial and temporal stream networks, we take a weighted average of them. When learned within the temporal segment networkframework,

the performance gap between spatial stream ConvNets and temporal stream ConvNets is much smaller than that in the original two-stream ConvNets. Based on this fact, we give more credits spatial stream by setting its weight as 1 and the one of temporal stream as 1.5. When both normal and warped optical flow fields are used, the weight of temporal stream is divided to 1 for optical flow and 0.5 for warped optical flow. It is described in Sec. 4.3.1 that the segmental consensus function is applied before the Softmax normalization. To test the models in compliance with their training, we fuse the prediction scores of 25 frames and different streams before Softmax normalization.

4.4 Experiments

In this section, we first introduce the evaluation datasets and the implementation details of our approach. Then, we explore the proposed good practices for learning temporal segment networks. After this, we demonstrate the importance of modeling long-term temporal structures by applying the temporal segment network framework. We also compare the performance of our method with the state-of-the-art. Finally, we visualize our learned ConvNet models.

4.4.1 Datasets and Implementation Details

We conduct experiments on two large action datasets, namely HMDB51 [44] and UCF101 [78]. The UCF101 dataset con-

tains 101 action classes and 13,320 video clips. We follow the evaluation scheme of the THUMOS13 challenge [38] and adopt the three training/testing splits for evaluation. The HMDB51 dataset is a large collection of realistic videos from various sources, such as movies and web videos. The dataset is composed of 6,766 video clips from 51 action categories. Our experiments follow the original evaluation scheme using three training/testing splits and report average accuracy over these splits.

We use the mini-batch stochastic gradient descent algorithm to learn the network parameters, where the batch size is set to 256 and momentum set to 0.9. We initialize network weights with pre-trained models from ImageNet [11]. We set a smaller learning rate in our experiments. For spatial networks, the learning rate is initialized as 0.01 and decreases to its $\frac{1}{10}$ every 2,000 iterations. The whole training procedure stops at 4,500 iterations. For temporal networks, we initialize the learning rate as 0.005, which reduces to its $\frac{1}{10}$ every 12,000 iterations. The maximum iteration is set as 30,000. Concerning data augmentation, we use the techniques of location jittering, horizontal flipping, corner cropping, and scale jittering, as specified in Section 4.3.2. For the extraction of optical flow and warped optical flow, we choose the TVL1 optical flow algorithm [107] implemented in OpenCV with CUDA. To speed up training, we employ a data-parallel strategy with multiple GPUs, implemented with

our modified version of Caffe [37] and OpenMPI [1].

4.4.2 Exploration Study

In this section, we focus on the investigation the good practices described in Sec. 4.3.2, including the training strategies and the input modalities. In this exploration study, we use the two-stream ConvNets with very deep architecture adapted from [34] and perform all experiments on the split 1 of UCF101 dataset.

We propose two training strategies in Section 4.3.2, namely cross modality pre-training and partial BN with dropout. Specifically, we compare four settings: (1) training from scratch, (2) only pre-train spatial stream as in [76], (3) with cross modality pre-training, (4) combination of cross modality pre-training and partial BN with dropout. The results are summarized in Table 4.1. First, we see that the performance of training from scratch is much worse than that of the original two-stream ConvNets (baseline), which implies carefully designed learning strategy is necessary to reduce the risk of over-fitting, especially for spatial networks. Then, We resort to the pre-training of the spatial stream and cross modality pre-training of the temporal stream to help initialize two-stream ConvNets and it achieves better performance than the baseline. We further utilize the partial BN with dropout to regularize the training procedure, which boosts the recognition performance to 92.0%.

[1] `https://github.com/yjxiong/caffe`

Table 4.1: Exploration of different training strategies for two-stream ConvNets on the UCF101 dataset (split 1).

Training setting	Spatial ConvNets	Temporal ConvNets	Two-Stream
Baseline [76]	72.7%	81.0%	87.0%
From Scratch	48.7%	81.7%	82.9%
Pre-train Spatial(same as [76])	84.1%	81.7%	90.0%
+ Cross modality pre-training	84.1%	86.6%	91.5%
+ Partial BN with dropout	84.5%	87.2%	92.0%

We propose two new types of modalities in Section 4.3.2: RGB difference and warped optical flow fields. Results on comparing the performance of different modalities are reported in Table 4.2. These experiments are carried out with all the good practices verified in Table 4.1. We first observe that the combination of RGB images and RGB differences boosts the recognition performance to 87.3% . This result indicates that RGB images and RGB difference may encode complementary information. Then it is shown that optical flow and warped optical flow yield quite similar performance (87.2% vs. 86.9%) and the fusion of them can improve the performance to 87.8%. Combining all of four modalities leads to an accuracy of 91.7%. As RGB difference may describe similar but unstable motion patterns, we also evaluate the performance of combining the other three modalities and this brings better recognition accuracy (92.3% vs 91.7%). We conjecture that the optical flow is better at capturing motion information and sometimes RGB difference may be unstable for describing motions. On the other hand, RGB

Table 4.2: Exploration of different input modalities for two-stream ConvNets on the UCF101 dataset (split 1).

Modality	Performance
RGB Image	84.5%
RGB Difference	83.8%
RGB Image + RGB Difference	87.3%
Optical Flow	87.2%
Warped Flow	86.9%
Optical Flow + Warped Flow	87.8%
Optical Flow + Warped Flow + RGB	**92.3**%
All Modalities	91.7%

Table 4.3: Exploration of different segmental consensus functions for temporal segment networks on the UCF101 dataset (split 1).

Consensus Function	Spatial ConvNets	Temporal ConvNets	Two-Stream
Max	85.0%	86.0%	91.6%
Average	85.7%	87.9%	**93.5**%
Weighted Average	86.2%	87.7%	92.4%

difference may serve as a low-quality, high-speed alternative for motion representations.

4.4.3 Evaluation of Temporal Segment Networks

In this subsection, we focus on the study of the temporal segment network framework. We first study the effect of segmental consensus function and then compare different ConvNet architectures on the split 1 of UCF101 dataset. For fair comparison, we only use RGB images and optical flow fields for input modalities in this exploration. As mentioned in Sec 4.3.1, the number

of segments K is set to 3.

In Eq. (4.1), a segmental consensus function is defined by its aggregation function g. Here we evaluate three candidates: (1) max pooling, (2) average pooling, (3) weighted average, for the form of g. The experimental results are summarized in Table 4.3. We see that average pooling function achieves the best performance. So in the following experiments, we choose average pooling as the default aggregation function. Then we compare the performance of different network architectures and the results are summarized in Table 4.4. Specifically, we compare three very deep architectures: BN-Inception [34], GoogLeNet [82], and VGGNet-16 [77], all these architectures are trained with the good practices aforementioned. Among the compared architectures, the very deep two-stream ConvNets adapted from BN-Inception [34] achieves the best accuracy of 92.0%. This is in accordance with its better performance in the image classification task. So we choose BN-Inception [34] as the ConvNet architecture for temporal segment networks.

With all the design choices set, we now apply the temporal segment network (TSN) to the action recognition. The result is illustrated in Table 4.4. A component-wise analysis of the components in terms of the recognition accuracies is also presented in Table 4.5. We can see that temporal segment networkis able to boost the performance of the model even when all the discussed good practices are applied. This corroborates that modeling long-term temporal structures is crucial for better un-

Table 4.4: Exploration of different very deep ConvNet architectures on the UCF101 dataset (split 1). "BN-Inception+TSN" refers to the setting where the temporal segment networkframework is applied on top of the best performing BN-Inception [34] architecture.

Training setting	Spatial ConvNets	Temporal ConvNets	Two-Stream
Clarifai [76]	72.7%	81.0%	87.0%
GoogLeNet	77.1%	83.9%	89.0%
VGGNet-16	79.8%	85.7%	90.9%
BN-Inception	84.5%	87.2%	92.0%
BN-Inception+TSN	85.7%	87.9%	**93.5**%

Table 4.5: Component analysis of the proposed method on the UCF101 dataset (split 1). From left to right we add the components one by one. BN-Inception [34] is used as the ConvNet architecture.

Component	Basic Two-Stream [76]	Cross-Modality Pre-training	Partial BN with dropout	Temporal Segment Networks
Accuracy	90.0%	91.5	92.0%	93.5%

derstanding of action in videos. And it is achieved by temporal segment networks.

4.4.4 Comparison with the State of the Art

After exploring of the good practices and understanding the effect of temporal segment network, we are ready to build up our final action recognition method. Specifically, we assemble three input modalities and all the techniques described as our final recognition approach, and test it on two challenging datasets: HMDB51 and UCF101. The results are summarized in Ta-

Table 4.6: Comparison of our method based on temporal segment network(TSN) with other state-of-the-art methods. We separately present the results of using two input modalities (RGB+Flow) and three input modalities (RGB+Flow+Warped Flow).

HMDB51		UCF101	
DT+MVSV [5]	55.9%	DT+MVSV [5]	83.5%
iDT+FV [89]	57.2%	iDT+FV [90]	85.9%
iDT+HSV [69]	61.1%	iDT+HSV [69]	87.9%
MoFAP [97]	61.7%	MoFAP [97]	88.3%
Two Stream [76]	59.4%	Two Stream [76]	88.0%
VideoDarwin [19]	63.7%	C3D (3 nets) [86]	85.2%
MPR [63]	65.5%	Two stream +LSTM [61]	88.6%
F_{ST}CN (SCI fusion) [80]	59.1%	F_{ST}CN (SCI fusion) [80]	88.1%
TDD+FV [96]	63.2%	TDD+FV [96]	90.3%
LTC [88]	64.8%	LTC [88]	91.7%
KVMF [110]	63.3%	KVMF [110]	93.1%
TSN (2 modalities)	68.5%	TSN (2 modalities)	94.0%
TSN (3 modalities)	**69.4**%	TSN (3 modalities)	**94.2**%

ble 4.6, where we compare our method with both traditional approaches such as improved trajectories (iDTs) [89], MoFAP representations [97], and deep learning representations, such as 3D convolutional networks (C3D) [86], trajectory-pooled deep-convolutional descriptors (TDD) [96], factorized spatio-temporal convolutional networks (F_{ST}CN) [80], long term convolution networks (LTC) [88], and key volume mining framework (KVMF). Our best result outperforms other methods by 3.9% on the HMDB51 dataset, and 1.1% on the UCF101 dataset. The superior performance of our methods demonstrates the effectiveness of temporal segment networkand justifies the importance of long-

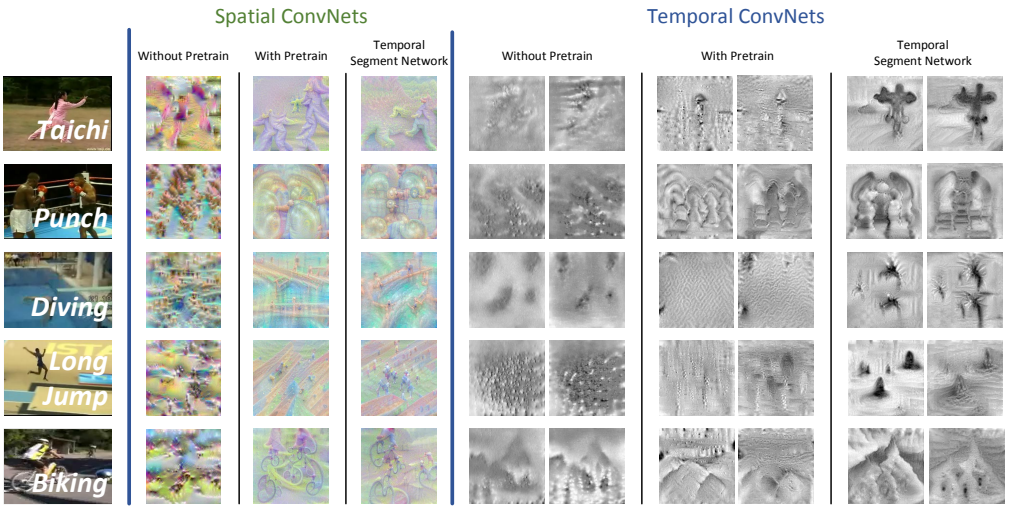


Figure 4.3: Visualization of ConvNet models for action recognition using DeepDraw [1]. We compare three settings: (1) without pre-train; (2) with pre-train; (3) with temporal segment network. For spatial ConvNets, we plot three generated visualization as color images. For temporal ConvNets, we plot the flow maps of x (left) and y (right) directions in gray-scales. Note all these images are generated from purely random pixels.

term temporal modeling.

4.4.5 Model Visualization

Besides recognition accuracies, we would like to attain further insight into the learned ConvNet models. In this sense, we adopt the DeepDraw [1] toolbox. This tool conducts iterative gradient ascent on input images with only white noises. Thus the output after a number of iterations can be considered as class visualization based solely on class knowledge inside the ConvNet model. The original version of the tool only deals with RGB data. To conduct visualization on optical flow based models, we adapt the

tool to work with our temporal ConvNets. As a result, we for the first time visualize interesting class information in action recognition ConvNet models. We randomly pick five classes from the UCF101 dataset, *Taichi*, *Punch*, *Diving*, *Long Jump*, and *Biking* for visualization. The results are shown in Fig. 4.3. For both RGB and optical flow, we visualize the ConvNet models learned with following three settings: (1) without pre-training; (2) only with pre-training; (3) with temporal segment network.

Generally speaking, models with pre-training are more capable of representing visual concepts than those without pre-training. One can see that both spatial and temporal models without pre-training can barely generate any meaningful visual structure. With the knowledge transferred from the pre-training process, the spatial and temporal models are able to capture structured visual patterns.

It is also easy to notice that the models, trained with only short-term information such as single frames, tend to mistake the scenery patterns and objects in the videos as significant evidences for action recognition. For example, in the class "Diving", the single-frame spatial stream ConvNet mainly looks for water and diving platforms, other than the person performing diving. Its temporal stream counterpart, working on optical flow, tends to focus on the motion caused by waves of surface water. With long-term temporal modeling introduced by temporal segment network, it becomes obvious that learned models focus more on humans in the videos, and seem to be modeling

the long-range structure of the action class. Still consider "Diving" as the example, the spatial ConvNet with temporal segment networknow generate a image that human is the major visual information. And different poses can be identified in the image, depicting various stages of one diving action. This suggests that models learned with the proposed method may perform better, which is well reflected in our quantitative experiments.

4.5 Discussion and Summary

In this thesis work, we have introduced temporal segment network, a video level-framework aiming to model long-term temporal structure in action videos. The ideas of sparse temporal sampling strategy and video-level supervision make it an efficient and effective framework for video-based action recognition. The framework is a strict realization the idea of combining multiple aspect data, as it utilizes both appearances, motion, and temporal structures of videos. The superior performance of the temporal segment networkframework is also justified by carefully designed visualization of the learned models. Compared with previous works on this problem, the described method not only delivers convincing performance, but also shows many possibilities where new research can be conducted upon.

Chapter 5

Conclusion

This thesis work mainly focus on high-level visual understanding, with emphasize on combining multiple aspect data in the learning and prediction. Three projects are finished in this thesis: 1)recognize complex events from images by fusing deep channels, 2)multi-label image tagging by uniting different scales, locations, and concept categories, 3)human activity recognition from videos by combining motion, appearances, and temporal structures. We hope this thesis serves as an good example how to deal with high-level visual understanding tasks and benefits other research works.

In the first part of the thesis work, a novel framework for event recognition from still images is proposed. It combines multiple channels of deep networks to effectively analysis the many semantic aspects of a complex event. The utilized information ranges from apparent appearances to complicated interactions between human and surrounding environment. A unified

deep network architecture is devised to perform the combination. Quantitative experimental evaluations on multiple dataset demonstrate that it outperforms state-of-the-art methods.

Then in the second part of the thesis, we apply the idea of combining multiple aspect of data to the problem of image tagging. A novel framework is proposed to unite visual information scattered in various scales and locations on the images, and contrastively learn the visual concepts to be tagged. The proposed techniques of *Scaled View Integration* and *Contrast Based Learning* results to a simple yet powerful deep architecture. Experimental results show improved performance over state-of-the-art methods, which corroborates the superiority of our framework.

As the last part of the thesis work, the idea of combining multiple aspect of information is extended to the temporal side, when we study the problem of human activity recognition from videos. We propose a unified framework, called temporal segment networks (TSN), to incorporate appearances, short-term motion patterns, and long-term temporal structures. An end-to-end learning scheme is devised to learn from all these aspects of video data, resulting in a method that is efficient in both learning and inference. Significant improvement over other state-of-the-art methods is observed in experimental analysis. Further visualization of the learned models also justifies superiority of the proposed framework and demonstrates the importance of combining the information from the aspects above.

5.1 Future Works

The works demonstrated in this thesis work show that our approaches, by combining multiple aspects of data, can lead to promising results in various tasks of high-level visual understanding in real world. But there are still a lot of open questions to be explored. In our future works, we will further investigate how to devise a principled approach to the integration of information in the context of deep learning. Specifically, we would like to propose a framework that can jointly learn from the existing large corpus of image and videos. Starting from current supervised learning scheme, we would like go beyond to exploring the possibility of performing weakly-supervised or unsupervised learning by utilizing the coherence of multiple data aspects.

Bibliography

[1] Deep draw. `https://github.com/auduno/deepdraw`.

[2] K. Barnard and D. Forsyth. Learning the semantics of words and pictures. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 408–415. IEEE, 2001.

[3] C. M. Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995.

[4] W. Brendel, A. Fern, and S. Todorovic. Probabilistic event logic for interval-based event recognition. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3329–3336, June 2011.

[5] Z. Cai, L. Wang, X. Peng, and Y. Qiao. Multi-view super vector for action recognition. In *CVPR*, pages 596–603, 2014.

[6] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image

annotation and retrieval. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(3):394–410, 2007.

[7] G. Chechik, V. Sharma, U. Shalit, and S. Bengio. Large scale online learning of image similarity through ranking. *The Journal of Machine Learning Research*, 11:1109–1135, 2010.

[8] L. Chen, D. Xu, I. W. Tsang, and J. Luo. Tag-based web photo retrieval improved by batch mode re-tagging. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3440–3446. IEEE, 2010.

[9] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 539–546. IEEE, 2005.

[10] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng. Nus-wide: A real-world web image database from national university of singapore. In *Proc. of ACM Conf. on Image and Video Retrieval (CIVR'09)*, Santorini, Greece., July 8-10, 2009.

[11] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li. ImageNet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[12] C. Desai and D. Ramanan. Detecting actions, poses, and objects with relational phraselets. In *Computer Vision–ECCV 2012*, pages 158–172. Springer, 2012.

[13] P. Dollar, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(8):1532–1545, Aug 2014.

[14] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on*, pages 65–72. IEEE, 2005.

[15] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, pages 2625–2634, 2015.

[16] L. Duan, D. Xu, I.-H. Tsang, and J. Luo. Visual event recognition in videos by learning from web data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(9):1667–1680, Sept 2012.

[17] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *Pattern Anal-*

ysis and Machine Intelligence, IEEE Transactions on, 35(8):1915–1929, 2013.

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

[19] B. Fernando, E. Gavves, J. O. M., A. Ghodrati, and T. Tuytelaars. Modeling video evolution for action recognition. In *CVPR*, pages 5378–5387, 2015.

[20] A. Gaidon, Z. Harchaoui, and C. Schmid. Temporal localization of actions with actoms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(11):2782–2795, 2013.

[21] C. Gan, N. Wang, Y. Yang, D.-Y. Yeung, and A. G. Hauptmann. Devnet: A deep event network for multimedia event detection and evidence recounting. In *CVPR*, pages 2568–2577, 2015.

[22] C. Gan, Y. Yang, L. Zhu, D. Zhao, and Y. Zhuang. Recognizing an action using its name: A knowledge-based approach. *International Journal of Computer Vision*, pages 1–17, 2016.

[23] C. Gan, T. Yao, K. Yang, Y. Yang, and T. Mei. You lead, we exceed: Labor-free video concept learning by jointly

exploiting web videos and images. In *CVPR*, pages 923–932, 2016.

[24] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 580–587, June 2014.

[25] R. Girshick, F. Iandola, T. Darrell, and J. Malik. Deformable part models are convolutional neural networks. *arXiv preprint arXiv:1409.5403*, 2014.

[26] Y. Gong, Y. Jia, T. Leung, A. Toshev, and S. Ioffe. Deep convolutional ranking for multilabel image annotation. *arXiv preprint arXiv:1312.4894*, 2013.

[27] Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. In *ECCV 2014*, pages 392–407. Springer, 2014.

[28] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 309–316. IEEE, 2009.

[29] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik. Learning rich features from rgb-d images for object detection and

segmentation. In *Computer Vision – ECCV 2014*, volume 8695 of *Lecture Notes in Computer Science*, pages 345–360. Springer International Publishing, 2014.

[30] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Computer Vision–ECCV 2014*, pages 346–361. Springer, 2014.

[31] C.-L. Huang, H.-C. Shih, and C.-Y. Chao. Semantic analysis of soccer video using dynamic bayesian network. *Multimedia, IEEE Transactions on*, 8(4):749–760, 2006.

[32] S. J. Hwang and K. Grauman. Learning the relative importance of objects from tagged images for retrieval and cross-modal search. *International journal of computer vision*, 100(2):134–153, 2012.

[33] N. Ikizler, R. G. Cinbis, S. Pehlivan, and P. Duygulu. Recognizing actions from still images. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4. IEEE, 2008.

[34] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015.

[35] S. Ji, W. Xu, M. Yang, and K. Yu. 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(1):221–231, 2013.

[36] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[37] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *CoRR*, abs/1408.5093.

[38] Y.-G. Jiang, J. Liu, A. Roshan Zamir, I. Laptev, M. Piccardi, M. Shah, and R. Sukthankar. THUMOS challenge: Action recognition with a large number of classes, 2013.

[39] J. Johnson, L. Ballan, and F.-F. Li. Love thy neighbors: Image annotation by exploiting image metadata. *ICCV*, 2015.

[40] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, pages 1725–1732, 2014.

[41] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In

Advances in Neural Information Processing Systems 25, pages 1097–1105. Curran Associates, Inc., 2012.

[42] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[43] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012.

[44] H. Kuehne, H. Jhuang, E. Garrote, T. A. Poggio, and T. Serre. HMDB: A large video database for human motion recognition. In *ICCV*, pages 2556–2563, 2011.

[45] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169–2178. IEEE, 2006.

[46] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169–2178. IEEE, 2006.

[47] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[48] J. Li and J. Z. Wang. Real-time computerized annotation of pictures. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(6):985–1002, 2008.

[49] J. Li and Y. Zhang. Learning surf cascade for fast and accurate object detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3468–3475. IEEE, 2013.

[50] L.-J. Li and L. Fei-Fei. What, where and who? classifying events by scene and object recognition. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, Oct 2007.

[51] L.-J. Li, R. Socher, and L. Fei-Fei. Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2036–2043, June 2009.

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

[53] M. Li, X.-B. Xue, and Z.-H. Zhou. Exploiting multi-modal interactions: A unified framework. In *IJCAI*, pages 1120–1125, 2009.

[54] W. Li, L. Niu, and D. Xu. Exploiting privileged information from web data for image categorization. In *Computer Vision – ECCV 2014*, volume 8693 of *Lecture Notes in Computer Science*, pages 437–452. Springer International Publishing, 2014.

[55] X. Li, C. Snoek, and M. Worring. Learning social tag relevance by neighbor voting. *Multimedia, IEEE Transactions on*, 11(7):1310–1322, Nov 2009.

[56] X. Li, T. Uricchio, L. Ballan, M. Bertini, C. G. Snoek, and A. Del Bimbo. Socializing the semantic gap: A comparative survey on image tag assignment, refinement and retrieval. *arXiv preprint arXiv:1503.08248*, 2015.

[57] P. Luo, X. Wang, and X. Tang. Pedestrian parsing via deep decompositional network. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2648–2655. IEEE, 2013.

[58] S. Maji, L. Bourdev, and J. Malik. Action recognition from a distributed representation of pose and appearance. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3177–3184. IEEE, 2011.

[59] J. McAuley and J. Leskovec. Image labeling on a network: using social-network metadata for image classification. In *Computer Vision–ECCV 2012*, pages 828–841. Springer, 2012.

[60] M. Naphade, J. R. Smith, J. Tesic, S.-F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis. Large-scale concept ontology for multimedia. *MultiMedia, IEEE*, 13(3):86–91, 2006.

[61] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *CVPR*, pages 4694–4702, 2015.

[62] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 689–696, 2011.

[63] B. Ni, P. Moulin, X. Yang, and S. Yan. Motion part regularization: Improving action recognition via trajectory group selection. In *CVPR*, pages 3698–3706, 2015.

[64] J. C. Niebles, C.-W. Chen, and F.-F. Li. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, pages 392–405, 2010.

[65] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

[66] P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, and G. Quéenot. Trecvid 2014 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2014*. NIST, USA, 2014.

[67] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1307–1314, Nov 2011.

[68] D. Parikh and K. Grauman. Relative attributes. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 503–510, Nov 2011.

[69] X. Peng, L. Wang, X. Wang, and Y. Qiao. Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice. *Computer Vision and Image Understanding*, 150:109–125, 2016.

[70] H. Pirsiavash and D. Ramanan. Parsing videos of actions with segmental grammars. In *CVPR*, pages 612–619, 2014.

[71] M. Raptis and S. Soatto. Tracklet descriptors for action modeling and video analysis. In *Computer Vision–ECCV 2010*, pages 577–590. Springer, 2010.

[72] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*, pages 512–519. IEEE, 2014.

[73] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

[74] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified ding for face recognition and clustering. *arXiv preprint arXiv:1503.03832*, 2015.

[75] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. *CoRR*, abs/1406.2199, 2014.

[76] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576, 2014.

[77] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, pages 1–14, 2015.

[78] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.

[79] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *Advances in Neural Information Processing Systems 25*, pages 2222–2230. Curran Associates, Inc., 2012.

[80] L. Sun, K. Jia, D. Yeung, and B. E. Shi. Human action recognition using factorized spatio-temporal convolutional networks. In *ICCV*, pages 4597–4605, 2015.

[81] Y. Sun, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. *CoRR*, abs/1406.4773, 2014.

[82] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015.

[83] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Computer Vision and Pattern Recognition*

(CVPR), 2014 IEEE Conference on, pages 1701–1708. IEEE, 2014.

[84] K. Tang, L. Fei-Fei, and D. Koller. Learning latent temporal structure for complex event detection. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1250–1257, June 2012.

[85] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li. The new data and new challenges in multimedia research. *arXiv preprint arXiv:1503.01817*, 2015.

[86] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015.

[87] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

[88] G. Varol, I. Laptev, and C. Schmid. Long-term temporal convolutions for action recognition. *CoRR*, abs/1604.04494, 2016.

[89] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, pages 3551–3558, 2013.

[90] H. Wang and C. Schmid. LEAR-INRIA submission for the thumos workshop. In *ICCV Workshop on THUMOS Challenge*, pages 1–3, 2013.

[91] H. Wang, M. M. Ullah, A. Klaser, I. Laptev, C. Schmid, et al. Evaluation of local spatio-temporal features for action recognition. In *BMVC 2009-British Machine Vision Conference*, 2009.

[92] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1386–1393. IEEE, 2014.

[93] L. Wang, Y. Qiao, and X. Tang. Motionlets: Mid-level 3D parts for human motion recognition. In *CVPR*, pages 2674–2681, 2013.

[94] L. Wang, Y. Qiao, and X. Tang. Latent hierarchical model of temporal structure for complex activity classification. *IEEE Trans. Image Processing*, 23(2):810–822, 2014.

[95] L. Wang, Y. Qiao, and X. Tang. Video action detection with relational dynamic-poselets. In *ECCV*, pages 565–580, 2014.

[96] L. Wang, Y. Qiao, and X. Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *CVPR*, pages 4305–4314, 2015.

[97] L. Wang, Y. Qiao, and X. Tang. MoFAP: A multi-level representation for action recognition. *International Journal of Computer Vision*, pages 1–18, 2015.

[98] X.-J. Wang, L. Zhang, X. Li, and W.-Y. Ma. Annotating images by mining image search results. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(11):1919–1932, 2008.

[99] J. Weston, S. Bengio, and N. Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *IJCAI*, volume 11, pages 2764–2770, 2011.

[100] S. Wu, S. Bondugula, F. Luisier, X. Zhuang, and P. Natarajan. Zero-shot event detection using multi-modal fusion of weakly supervised concepts. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 2665–2672. IEEE, 2014.

[101] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3485–3492, June 2010.

[102] Y. Xiong, K. Zhu, D. Lin, and X. Tang. Recognize complex events from static images by fusing deep channels. In *CVPR*, pages 1600–1609, 2015.

[103] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden markov model. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, pages 379–385. IEEE, 1992.

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

[105] Y. Yang and M. Shah. Complex events detection using data-driven concepts. In *Computer Vision – ECCV 2012*, volume 7574, pages 722–735. Springer Berlin Heidelberg, 2012.

[106] B. Yao and L. Fei-Fei. Recognizing human-object interactions in still images by modeling the mutual context of objects and human poses. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(9):1691–1703, 2012.

[107] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-L^1 optical flow. In *29th DAGM Symposium on Pattern Recognition*, pages 214–223, 2007.

[108] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833, 2014.

[109] B. Zhang, L. Wang, Z. Wang, Y. Qiao, and H. Wang. Real-time action recognition with enhanced motion vector CNNs. In *CVPR*, pages 2718–2726, 2016.

[110] W. Zhu, J. Hu, G. Sun, X. Cao, and Y. Qiao. A key volume mining deep framework for action recognition. In *CVPR*, pages 1991–1999, 2016.

[111] C. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *Computer Vision – ECCV 2014*, volume 8693, pages 391–405. Springer International Publishing, 2014.