UNIVERSITY OF CALIFORNIA, SAN DIEGO

Deep Learning for Image Understanding

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Electrical Engineering (Signal and Image Processing)

by

Yufei Wang

Committee in charge:

Professor Garrison W. Cottrell, Chair
Professor Nuno Vasconcelos, Co-Chair
Professor Kenneth Kreutz-Delgado
Professor Bhaskar D. Rao
Professor Lawrence K. Saul

2017

The dissertation of Yufei Wang is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____
Co-Chair

_____
Chair

University of California, San Diego

2017

DEDICATION

To my family.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

The pursuit of Ph. D. is an unforgettable experience, and along the way, there are so many people that I want to thank.

The most important person I would like to thank is of course my Ph. D. advisor, Dr. Garrison W. Cottrell. Gary is very supportive in any way I can imagine, and his insightful ideas and concrete supervision are of great help for me. His serious attitude towards academia influenced my entire graduate study. Not only is Gary a great advisor in research, he is also a mentor and cares like a friend. Whenever I have troubles and doubts, he is always there listening, and ready to give advice.

I also want to thank my co-advisor, Dr. Nuno Vasconcelos, for giving me valuable advice on my research throughout my Ph. D. study. I am honored to have Dr. Bhaskar D. Rao, Dr. Lawrence K. Saul, and Dr. Kenneth Kreutz-Delgado to serve as my doctoral committee. Their insightful discussion and invaluable advice are great asset to my academia career.

I feel thankful to my collaborators from Adobe Research, Dr. Zhe Lin, Dr. Xiaohui Shen, Dr Scott Cohen, Dr. Jianming Zhang, Dr. Radomir Měch, and Dr. Gavin Miller. Through the internships and long time collaboration with Adobe Research, every deep discussion is very valuable to my research. I would like to express my special thanks to Zhe for his mentorship. His inspiration and insights as well as his detailed advice and patient supervision are invaluable to me. Discussion with Xiaohui is always inspiring, and I've learnt a lot from him. I also want to thank Scott for not only giving me valuable academic advice, but also cares me greatly in my personal life, which helped me go through my toughest time. My works greatly benefit from Jianming's insightful suggestions, and Radomir and Gavin's mentorship.

I would like to thank my boyfriend, Si Chen. He was also a graduate student in UCSD. Before that, we studied in the same University in China. For the many years I've known him, he has always been the first person I go to when I have troubles, and he not only comforts me but also gives concrete advice on the steps I can take. With him always being there for me, I know

that there is nothing I cannot conquer. When I get too satisfied with my life and get lazy, he is there to remind me of my ambitions and goals; when I have doubts of myself, he always helps me regain the courage to continue fighting. He makes me a better person.

I also want to thank my parents, Suli Wu and Dr. Jianhua Wang, for their unconditional support and love for me throughout my life. Their upright, honest and hardworking personalities shaped my character. They have been strict with me, but they never fail to express how proud they are of me. When it comes to life-changing decisions such as pursuing a Ph. D., they give suggestions and express concerns, but fully support me once I make my decision. They never speak up for their love, but I can feel it every second, and I am grateful for it.

Along the journey, there are a lot of colleagues and friends that I would like to thank for their support on my research and life. My wonderful labmates from Gary's Unbelievable Research Unit (GURU) who make our lab a warm family: Honghao Shan, Ben Cipollini, Tomoki Tsuchida, Vicente Malave, Mohsen Malmir, Panqu Wang, Amanda Song, William Fedus, Yao Qin, Yan Shu, Davis Liang, Sanjeev Rao, Sandy Wiraatmadja, and Angel Zhang. A lot of thanks to my friends and colleagues: Yingwei Li, Shuai Tang, Ning Ma, Mengting Wan, Zhaowei Cai, Weixin Li, and Xiaodi Hou. Also, my friends outside of my research that made the four years of my life colorful: Wei Huang, Yilun Zhang, Lijuan Huang, Pengfei Chen, Zhiyuan Sun, Jiacong Li, Yao Peng, Yi Yang, Jingxin Ye, Gufeng Zhang, Yuan Fang, Chuan Wang, Qiao Zhang, Dongjin Song, Rui Hua, Huan Hu and Qian Yao.

I would like to thank the funders that supported my research, including the Temporal Dynamics of Learning Center (TDLC), NSF (SMA 1041755 to the TDLC, and NSF grant # IIS-1219252 to GWC), UC San Diego (provided a Fellowship to Yufei Wang), Adobe Research (provided gift money to Garrison W. Cottrell).

Chapter 2, in full, is an edited reprint of the material as it appears in the following publication: Wang, Y., Zhe, L., Shen, X., Měch, R., Miller, G., Cottrell, G. W., "Event-specific Image Importance", In *Computer Vision and Pattern Recognition (CVPR)*, 2016. The dissertation

author was a primary researcher and an author of the cited material.

Chapter 3, in full, is an edited reprint of the material as it appears in the following publication: Wang, Y., Zhe, L., Shen, X., Měch, R., Miller, G., Cottrell, G. W. (2017), "Recognizing and Curating Photo Albums via Event-Specific Image Importance", In *British Machine Vision Conference (BMVC)*, 2017. The dissertation author was a primary researcher and an author of the cited material.

Chapter 4, in full, is an edited reprint of the material as it appears in the following publication: Wang, Y., Zhe, L., Shen, X., Cohen, S., Cottrell, G. W., "Skeleton Key: Image Captioning by Skeleton-Attribute Decomposition", In *Computer Vision and Pattern Recognition (CVPR)*, 2017. The dissertation author was a primary researcher and an author of the cited material.

VITA

2013	B. S. in Electrical Engineering, University of Science and Technology of China, China
2017	M. S. in Electrical Engineering (Signal and Image Processing), University of California, San Diego
2017	Ph. D. in Electrical Engineering (Signal and Image Processing), University of California, San Diego

PUBLICATIONS

Wang, Y., Zhe, L., Shen, X., Zhang, J., Cohen, S., "Concept Mask: Large Scale Segmentation from Semantic Concepts", Under Review in *Computer Vision and Pattern Recognition (CVPR)*, 2018

Wang, Y., Zhe, L., Shen, X., Měch, R., Miller, G., Cottrell, G. W., "Recognizing and Curating Photo Albums via Event-Specific Image Importance", In *British Machine Vision Conference (BMVC)*, 2017.

Wang, Y., Zhe, L., Shen, X., Cohen, S., Cottrell, G. W., "Skeleton Key: Image Captioning by Skeleton-Attribute Decomposition", In *Computer Vision and Pattern Recognition (CVPR)*, 2017

Wang, Y., Zhe, L., Shen, X., Měch, R., Miller, G., Cottrell, G. W., "Event-specific Image Importance", In *Computer Vision and Pattern Recognition (CVPR)*, 2016

Rao, S., Wang, Y., G., Cottrell, G. W., "A Deep Siamese Neural Network Learns the Human-Perceived Similarity Structure of Facial Expressions Without Explicit Categories." In *Proceedings of the 38th annual conference of the cognitive science society*, 2016.

Wang, Y., and Cottrell, G. W., "Bikers are like tobacco shops, formal dressers are like suits: Recognizing Urban Tribes with Caffe". In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2015.

Tang, A., Lu, K., Wang, Y., Huang, J., Li, H., "A real-time hand posture recognition system using deep neural networks.", In *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2015

ABSTRACT OF THE DISSERTATION

Deep Learning for Image Understanding

by

Yufei Wang

Doctor of Philosophy in Electrical Engineering (Signal and Image Processing)

University of California, San Diego, 2017

Professor Garrison W. Cottrell, Chair
Professor Nuno Vasconcelos, Co-Chair

Computer vision and image understanding is the problem of interpreting images by locating, recognizing objects, attributes and other higher level features in an image. In this thesis, I seek to tackle this broad problem using deep learning techniques. More specifically, I build deep neural network based models to solve two specific problems to understand images in a high level: album wise image understanding with event-specific image importance score, and description generation for an image.

I first focus on the understanding of a collection of images in an event album. In an event album, some images are more important or interesting to save or present than others, and I show

that with an event-specific image importance property, we can learn the interestingness of an image given an album, and the performance of the model generated importance score is very close to human preference. I build a siamese network that can predict image importance score given the event type of that image, using novel objective function and learning scheme. Next, to make the process fully automated, I propose an iterative updating procedure for event type and image importance score prediction, that can simultaneously decide the event type of the album and the importance score of every image. It consists of a Convolutional Neural Network that recognizes the event type, a Long-Short Term Memory (LSTM) that uses sequential information for event type recognition, and a siamese network that predicts image importance score.

Furthermore, not just limited to describing an image with a score or by a classified type, I seek the possibility to describe it with a phrase or sentence. I propose a coarse-to-fine LSTM based method that decomposes the original image description into a skeleton sentence and its notable attributes, and demonstrate that in this way the language model can generate better descriptions, with the capability to generate image descriptions that better accommodates user preference.

Chapter 1

Introduction

Computer vision and image understanding is one of the main problem of artificial intelligence. It involves many attempts to help computer "see" images better. Early study for image understanding mostly focused on extracting the low level features, such as feature extraction for edges, corners, and optical flow [47, 14, 52]. The understanding of middle level features such as image segmentation, object detection and recognition then became major focus for many research studies [21, 36, 78, 119, 8]. More recently, with the access to large scale images with high quality annotations through the internet, and the speed up of computing with hardware innovation (GPUs), deep neural networks [43, 69] have brought great innovation into many research areas. Convolutional Neural Networks (CNN) has especially inspired great advance for many problems in image understanding [70, 98, 108, 38, 41]. Recurrent Neural Networks (RNN) and Long-Short Term Memory (LSTM) are widely used in sequence learning, such as machine translation [105] and image captioning [126].

In this thesis, I seek to use deep learning techniques to solve two problems in image understanding. First, I use a siamese network based model and LSTM based model to simultaneously predict album-wise event type and image importance for personal album organization. Second, I propose a coarse-to-fine LSTM based model for image caption generation. In this chapter, I provide relevant background knowledge for the topics relevant to this thesis.

1.1 Deep Learning

Most recently, thanks to the easy access to large scale image set via internet and great efforts researchers take to collect high quality annotations [94], the advance in network architecture [65, 100, 107, 48, 54], and development of faster computing hardware (GPUs), deep learning has been a great success, and has brought large performance boost to many areas in computer vision and image understanding, including object recognition [65, 48], object detection [41, 40, 93, 91], semantic segmentation [98, 64, 17, 130, 22], image captioning [117, 126, 81], and so on.

Deep convolutional neural networks (DCNN) are a type of feed forward network especially designed for image related task. They are advantageous over traditional multilayer perceptron networks in that they are much deeper, with tens or even hundreds of layers, and can learn the image from low level features to very high level features. The basic structure of unit in a DCNN consists of three layers: 1) a two dimensional convolutional layers that learns directly from the input image or from the activation of the previous layers. It preserves the spatial information of the input image and learns translation invariant features; 2) a spatial pooling layer which shrinks the size of features and at the same time enlarges the receptive field of the network; 3) a non-linear activation layer which improves the complexity and expressiveness of the network. With the stack of such units, the network is able to learn different level of features, from the low level features like corner and edges in the early layers, to the high level features like object parts and attributes in the late layers. The output of the stack of units is a high level feature vector representing the input image. In addition to the basic units, there are many variations of the network architecture to enhance the network's ability to interpret images [54, 48, 106, 49, 53].

On top of the feature extraction layers, the features are used for different tasks. For example, for object recognition, the final layer is an aggregated layer over different locations followed by a Softmax layer with cross-entropy loss function, and the output of the layer is the probability distribution of each object category given the input; for semantic segmentation, the output will be probability of each image pixel being in each object/stuff category.

With the use of back-propagation [71], DCNN's can learn the features from the image data directly, and greatly exceeds the performance of human designed features.

Recurrent neural networks (RNN), on the other hand, is different from feed forward networks in that the network not only takes its current input example as input, but also what it has perceived previously. It is designed for understanding a sequence of data, such as texts, handwriting, and spoken words. For each time step of an RNN, it has two sources of input: the present input data, and the output hidden state of the network in the previous time-step. The

learning of RNN relies on back-propagation through time [84], the extension of back-propagation.

In this thesis, I seek to use deep learning techniques for image understanding problems.

1.2 Album-wise Image Understanding

The first problem I aim to tackle is to understand personal photo albums. A personal photo album is a collection of photos that we take in an event, for example a wedding event, or a trip event. The high level understanding of such photo collection involves two stages: recognizing the event type of the photo album, and suggesting the most important/interesting images in the collection to represent the album or to save for future use.

For event recognition, there are three types of approaches. The most popular approach takes videos as input and uses spatiotemporal features for event recognition [122]. The second approach uses single image as cue to recognize event type. This approach does not use temporal information or relevant frame importance, and only uses object level and scene level features from a single image [73]. In between the two approaches, album-wise event recognition has useful album-wise temporal information, but the images in an album are very sparse in time and is not temporally continuous. Bossard *et al*.[3] found the sequential information of the albums is helpful for learning the event type of the albums, despite their sparsity.

On the other hand, image importance is a complex image property that correlates with various factor, such as aesthetics [23], image interestingness [45, 28], and image memorability [55]. In this thesis, I propose a novel image property named event-specific image importance. To study this property, we collected the CUration of Flickr Events Dataset (CUFED), and let the human annotator to decide the image importance score given an event album. We intentionally gave vague instructions on how annotators decide the importance of an image, to encourage them to rate based on their intuition. We found out that the image importance is indeed highly related to the event type of the album it is in, and although the image importance is a highly subjective

property, there is significant consistency across different annotators on the importance score they give in an album.

In this thesis, in Chapter 2, I propose a deep siamese architecture that learns the relative importance score of an image given the album event type it is from, assuming the event type of an album is given in advance. Further, in Chapter 3, I propose an iterative procedure that jointly learns the event type of an album and the importance score for each image. Thus, the two tasks for personal album understanding can be solved simultaneously with our framework.

1.3 Image Captioning

With the advance of image understanding with the development of deep learning, the research on image understanding is not constrained to the interpretation of an image with classification scores or detected tags, and the task of automatically describing the images with a sentence has drawn great attention. The problem is more challenging than conventional computer vision task in that the description generation requires high level understanding of the image beyond simple object recognition. It also requires the organization of a sentence that correctly conveys the notable information in the image.

The dominant approach for image captioning is inspired by the machine translation task [105]. For machine translation, an Encoder-Decoder network is used to map the input sequence to a vector of a fixed dimensionality, and then to decode the target sequence from the vector. The popular network used for encoding/decoding is Recurrent Neural Network (RNN), in which each element of the text sequence share the same unit parameters, and is sequentially fed into the network. RNN can deal with sequences with arbitrary length. Specifically, Long-Short Term Memory (LSTM), a variation of RNN, is commonly used [50]. It is capable to learn long-term dependencies with a cell state.

Similar to machine translation, an image can be viewed as a sentence in the source

language, and an Encoder-Decoder network is used to translate it from the source language to the target sentence. Since the source "sentence" is in fact an image in the image captioning task, a CNN is used as Encoder, and LSTM is used as a Decoder.

Despite the great success in image captioning, most of the existing LSTM based methods suffer from two problems: 1) they tend to parrot back the sentences from the training corpus; and 2) the nature of predicting sentence words one by one means the attributes of a sentence is predicted before the object they are referring to, which is counter-intuitive.

To solve these two problems, in Chapter 4, I propose a coarse-to-fine model which decomposes the original caption into two parts: skeleton sentence which contains the main objects and structure in the sentence, and notable attributes for each object in the skeleton sentence.

1.4 Organization of the Thesis

In this thesis, I aim to tackle the two problems in high level image understanding. The rest of the thesis is organized as follows:

In Chapter 2, I introduce the problem of event-specific image importance. I collected a dataset for the study of this image property, and collect annotations of album-wise event type and image-wise importance score for the dataset, using Amazon Mechanical Turk (AMT). With the dataset, we show that the event-specific image importance property is subjective yet learnable. Furthermore, we propose a siamese network based architecture that can learn the image importance score with performance close to human perception, and the model assumes the event type information is know in advance.

In Chapter 3, I further extend our model to learn image importance score with no prior knowledge, by proposing an iterative procedure to learn the two album properties at the same time: album-wise event type recognition and image-wise importance score prediction. We show

that these two components of our algorithm help each other in turn, and with the algorithm, we can automatically organize and recognize a personal event album without any extra input, with performance close to that in Chapter 2.

In Chapter 4, I focus on the image captioning problem, which aims to use a sentence to describe a given image. I propose a coarse-to-fine LSTM based model which decomposes the original caption into two parts: skeleton sentence and its attributes. It is able to generate better and more unique descriptions for images, and has the ability to adjust the amount of information the caption conveys according to user preference.

Finally, in Chapter 5, I conclude the thesis by discussing the possible directions and future works.

Chapter 2

Event-specific Image Importance

This chapter, together with Chapter 3, aims to tackle the problem of personal album understanding. Two aspects are studied: image-wise importance prediction, and album-wise event recognition. This chapter focuses on the first aspect.

When creating a photo album of an event, people typically select a few important images to keep or share. There is some consistency in the process of choosing the important images, and discarding the unimportant ones. Modeling this selection process will assist automatic photo selection and album summarization. In this paper, we show that the selection of important images is consistent among different viewers, and that this selection process is related to the event type of the album. We introduce the concept of event-specific image importance. We collected a new event album dataset with human annotation of the relative image importance with each event album. We also propose a Convolutional Neural Network (CNN) based method to predict the image importance score of a given event album, using a novel rank loss function and a progressive training scheme. Results demonstrate that our method significantly outperforms various baseline methods.

2.1 Introduction

With the proliferation of cameras (in cell phones and other portable cameras), taking photographs is practically effortless, and happens frequently in everyday life. When attending an event, for instance, a Thanksgiving holiday, participants often take many photos recording every interesting moment during the event. This leads to an oversized album at the end of the event. When we need to simplify the album before saving to a device, or if we want to make a photo collage or a photo book to share our important moment with others, we have to go through the tedious and time-consuming work of selecting important images from a large album. Therefore, it is desirable to perform this task automatically.

Automatic photo selection or album summarization has been studied by some researchers

[115, 95, 101, 15, 120]. They aim at personal event albums, and visual content information as well as diversity and coverage is often considered jointly to obtain a summarization. However, these works ignored the role of the event type in the selection process. Intuitively, the event type of the album is an important criterion when we select important images. For example, if we need to select important photos from a vacation to Hawaii, the photo of the volcano on the Big Island is definitely important to keep, whereas if the album is a wedding ceremony, beautiful scenes are only background and are not likely to be more important than the shot of the bride and groom.

In this paper, we introduce the concept of event-specific image importance. It is different from general image interestingness or aesthetics, in that it is contextual, and is based on the album the image is in. We focus on the event-specific importance score of a single image, and do not consider summarization problems where diversity and coverage are also important: Image importance prediction is the most challenging and crucial part of the event curation/album summarization process; Moreover, the importance score can be directly applied to to any album summarization algorithm. We collect an event-specific image importance dataset from human annotators, and we show that the event-specific importance is subjective yet predictable. Finally, we provide a method for predicting event-specific image importance using Convolutional Neural Network (CNN). We propose a new loss function and training procedure, and our CNN method greatly outperforms different baselines.

2.2 Related Work

Image properties.

Importance of an image is a complex image property, and is related to many other image properties. Many image properties can be viewed as cues when selecting important images, such as memorability [56, 55], specificity [57], popularity [60], aesthetics and interestingness [28, 45]. Those image properties are correlated to image contents, such as high level features:

object and scene categories [56, 55, 57, 60, 28], and low level features: texture, edge distribution, etc.[45, 60]. In this work, rather than the general image properties mentioned above, we study event-specific image importance, which summarizes human preferences related to images within the context of an album, where the album is of a known event type.

Convolutional Neural Networks(CNNs).

The development of methods for training deep CNNs has led to rapid progress in many computer vision tasks in recent years. Substantial improvements have been made in basic computer vision problems such as image classification [65, 107], object detection [41, 16] and scene recognition [134, 29]. Now, there is a greater focus on learning higher-level image properties. One example closely related to our project is Xiong *et al*.'s work on event recognition from static images [125]. In this work, the network is divided into different channels, creating human and object maps that are then fused with the original images to jointly train a deep architecture that predicts the event type from a single image. Our model also uses deep representations to capture event features, but our focus is on event curation rather than event recognition. In fact, our model assumes that the event type is known. Event curation then requires choosing the most important images for the event in question.

Album summarization and photo selection.

The most closely related work to our project is on summarization and selection from an album or several albums.

Event summarization of public photo/video collections involves selecting the most important moments of a social event from a variety sources on the web[24, 99]. Here, the goal is to retrieve all of the important moments (diversity), while covering the whole event (coverage). More relevant to this project is work that attempts to summarize a single album [115, 95, 101]. Again, coverage and diversity of the albums are considered, and single image importance is used

as a cue[95, 101]. Sinha *et al*.aim at summarization of personal photo collections taken over a long time span, take the event type as one photo descriptor to calculate diversity and coverage of the photo subset [101].

For the photo selection problem, Yeh *et al*.proposed a ranking system for photographs based on a set of aesthetic rules and personal preferences [129]. Walber *et al*.use gaze information from user's photo viewing process to assist the automatic photo selection algorithm, so this work requires eyetracking[120]. The work by Ceroni *et al*.[15] is probably most relevant to our work. It focuses on selection of important photos from a single event album, and different factors are considered: image quality, presence of faces, concept features, and collection based features such as album size. However, each album used for training and testing in this work is collected from a single participant, and the important subset is picked by the same person: it does not focus on common human preferences. Moreover, the prediction algorithm is tested on unseen images in the same album used for training, and it does not focus on new album prediction.

Our work differs from all the above in that we focus on: i) whether humans have common preferences for image importance/preference scores, ii) whether image importance can be predicted for unseen albums with widely varying content, and iii) whether event type information is important for the prediction. To summarize, we are introducing a subjective but predictable image property: event-specific image importance, and we propose a method to predict this property.

2.3 The Curation of Flickr Events Dataset

Are people's ratings for images in albums representing particular events predictable? Our intuition is that in an album of a certain event type, there will be a consistent subset of images that will be preferred by most people. However, there is no available dataset to verify this intuition, or to test the degree of people's agreement on this highly subjective task. In this section, we describe the collection of the CUration of Flickr Events Dataset (CUFED), and measure the consistency of

human subjects' preferences on this dataset. CUFED provides a ground truth dataset that allows us to measure the predictability of human rated image importance scores, and to develop our prediction model. CUFED will be made available to public.

2.3.1 Album collection

In order to collect a dataset of albums of different event types, we segmented albums from the Yahoo Flickr Creative Commons 100M Dataset (YFCC100M) [113]. The YFCC100M Dataset has 100 million images and videos from Flickr. In this collection, each image has the following metadata: the user ID who uploaded this photo; the time the image was taken; and often there are user tags. We took advantage of the metadata to segment dataset into albums: For each photo uploader, events are segmented based on timestamps and tags: images taken within short time interval (3 hours) and with more than 1/3 common tags belong to one event. Using tags to filter the data was inspired by the observation that users tend to give the same tags to an event album instead of individually tagging every single image in it. Using this approach, we segmented 1.8 million albums from the YFCC100M dataset. Here, we randomly selected 20,000 albums to work with.

To get the event type of those albums, we presented the albums to workers on Amazon Merchanical Turk (AMT) and asked them to classify the albums into 23 event types. Aside from these event types, the workers could choose "Other events", "Not an event", "More than a single event" or "Cannot decide" instead of available event types. We chose our 23 event types so that they cover the most common events in our lives, ranging from weddings to sports games.

All 23 event types are shown in Table 2.1. Each album was labeled by 3 workers. Over 82% of the 20k albums received the same labels from at least 2 of the 3 workers. We kept the albums which were given the same label by 2 or more workers, and this label was given to the album. This resulted in 16,489 albums.

We further randomly selected 50-200 albums from each of the event types (except for

Table 2.1: 23 Event types, their corresponding number of albums, and percentage of significant albums at level $q = 0.05$ using Kendall's W statistics. The event types fall into four categories.

Categories	Important Personal Event	Personal Activity	Personal Trip	Holiday
Event types and # albums	Wedding:198 (98%) Birthday:180 (91%) Graduation:178 (88%)	Protest:50 (92%) Personal Music Activity:25 (92%) Religious Activity:50 (90%) Casual Family Gather:50 (84%) Group Activity:50 (82%) Personal Sports:100 (78%) Business Activity:50 (76%) Personal Art Activity:54 (70%)	Architecture/Art:50 (92%) Urban Trip:100 (89%) Cruise Trip:50 (88%) Nature Trip:50 (86%) Theme Park:100 (86%) Zoo:99 (85%) Museum:50 (84%) Beach Trip:50 (82%) Show:100 (82%) Sports Game:50 (58%)	Christmas:100 (87%) Halloween:99 (86%)

Personal Music Activity, which has 25 albums), resulting in a dataset of 1883 albums. The number of events of each type is shown in Table 2.1. The size of the albums varies between 30 and 100 images. We chose these parameters by hand to emphasize our intuition that some event types will have more consistent ratings, and hence more predictability, than others. Therefore, in this dataset, we emphasized those events in hope of learning more from them.

2.3.2 Data annotation

In order to get the rating for each image in an album, we presented an album together with its event type to AMT workers and let them rate each image in that album as very important, important, neutral, or irrelevant. The four ratings are mapped to scores $\{2,1,0,-2\}$ when creating ground truth. We intentionally did not give specific criteria for the rating levels, to encourage the workers to rate based on their intuition. In our pilot study, workers on AMT tended to mark a large proportion of the images as very important/important. This is understandable, since most of the albums are of high quality, but it leads to a ceiling effect on the ratings. To control the size

of images marked as important, we forced the workers to label 5%-30% of the images as very important, and 10%-50% as important. The average time to rate each image was 7.7 seconds. Each album was annotated by 5 distinct workers. 292 workers participated in the tasks. Over 90% of our data was annotated by 93 workers.

The image ratings collected from AMT differed in quality among different AMT workers. To avoid low quality work, only workers who passed an event recognition test using single images could proceed to the real task. In addition, we added two distractor images per album which were clearly not related to the event in order to screen workers who were not paying attention. However, it is not possible to assure the quality of an individual submission because of the subjective nature of the image importance rating task. Therefore, in order to filter "bad" submissions, we found workers who consistently gave scores far from others and filtered out their submissions. If more than 30% of his/her submissions had a euclidean distance from the average of other workers' submissions greater than a threshold, that worker's submissions were filtered out. Only two workers were filtered out in this way.

Figure 2.1 shows an example of the ground truth we obtained from AMT. Scores are normalized so that the range is (0,1), 1 being most important and 0 being totally irrelevant. It's best viewed electronically. Note that all images are stretched and distorted for viewing.

2.3.3 Consistency analysis

To examine the consistency of the human ratings of images, we split our subjects into two independent groups of two and three raters for each album, and used Spearman's rank correlation (ρ) to evaluate their consistency. ρ ranges from -1 (perfectly inverse correlation) to 1 (perfect correlation), while 0 indicates no correlation. For each album, we averaged the correlation scores of all possible random splits. The average correlation over all albums was 0.40.

We further evaluated the annotation consistency with Kendall's W, which directly calculates the agreement among multiple raters, and accounts for tied ranks. Kendall's W ranges from

Figure 2.1: Example of a *Wedding* album with ground truth obtained from 5 AMT workers. The ground truth score of each image is given under it. The images are sorted by the ground truth scores. The average Spearman's correlation ρ over all possible two workers-three workers splits for this album is 0.49.

0 (no agreement) to 1 (complete agreement). Note that in our workers' rating of one album, tied ranks are very frequent, since there are only 4 possible ratings, and the average album size is 52. Coincidentally, the average Kendall's W over all albums was also 0.40. Both Spearman's rank correlation ρ and Kendall's W showed significant consistency across subjects despite the high subjectivity of this problem.

To test the statistical significance of Kendall's W score, we did a permutation test over W to obtain the distribution of W under the null hypothesis, and for each event type, we used the Benjamini-Hochberg procedure to control the false discovery rate (FDR) for multiple comparisons [9]. At level $q = 0.05$, 86% of albums had significant agreement on average. Table 2.1 shows the percentage of albums with significant agreement for each event type. The different percentages of significant albums in different event types confirmed our intuition that some event types would be more consistently rated than others. The wedding event was the most consistently rated, with 98% of albums being significantly consistent, while for the sports game category, only 58% of the albums received significant consistency scores, the lowest among the 23 events.

Table 2.2: 23 Event types, and the (average Kendall's W score / average Spearman's correlation ρ) for each album. The event types fall into four categories.

Categories	**Important Personal Event**	**Personal Activity**	**Personal Trip**	**Holiday**
Event types and # albums	Wedding (0.486/0.548) Birthday (0.418/0.423) Graduation (0.413/0.427)	Personal Music Activity (0.425/0.447) Protest (0.418/0.446) Religious Activity (0.401/0.406) Casual Family Gather (0.383/0.369) Personal Sports (0.372/0.347) Business Activity (0.368/0.335) Group Activity (0.366/0.357) Personal Art Activity (0.339/0.280)	Architecture/Art (0.428/0.452) Theme Park (0.391/0.385) Museum (0.391/0.384) Cruise Trip (0.383/0.371) Urban Trip (0.372/0.349) Beach Trip (0.370/0.368) Show (0.366/0.336) Zoo (0.366/0.337) Nature Trip (0.357/0.321) Sports Game (0.349/0.288)	Halloween (0.395/0.397) Christmas (0.386/0.379)

In addition to Table 2.1, Table 2.2 shows the average Kendall's W as well as Spearman's correlation ρ for each event type. We can see that *Wedding* albums receive on average the highest correlation/agreement, while *PersonalArtActivity* albums receive the lowest score.

We also show some examples of albums in Figure 2.2 - Figure 2.5 with their average ρ and W. Note that the average ρ and W are for an individual album.

Figure 2.2-2.3 show two examples of albums that receive relatively high correlation and agreement from 5 AMT workers, and Figure 2.4-2.5 show two examples of albums that receive low correlation and agreement from 5 AMT workers. Figure 2.5 shows an example in which all the images are of similar quality and semantics, and it's hard for people to agree on the ranking.

2.4 Approach

In this section, we propose a Convolutional Neural Network (CNN) based method for estimating an event-specific image importance score in an album, given the event type of this album. We use a siamese network [103] with a novel rank loss function to take two images at a time and rank them relative to one another based on their scores.

2.4.1 CNN structure

The design of our siamese CNN architecture is shown in Fig. 2.6. It has several properties described in the following subsections.

Feature sharing among event types

We train a single siamese network with albums from all event types. The last layer, however, has separate outputs for each event type. The reasons are as follows. First, there exists strong visual similarity among different event types in terms of image importance, therefore for a specific event type, labeled data from other event types will help as implicit data augmentation.

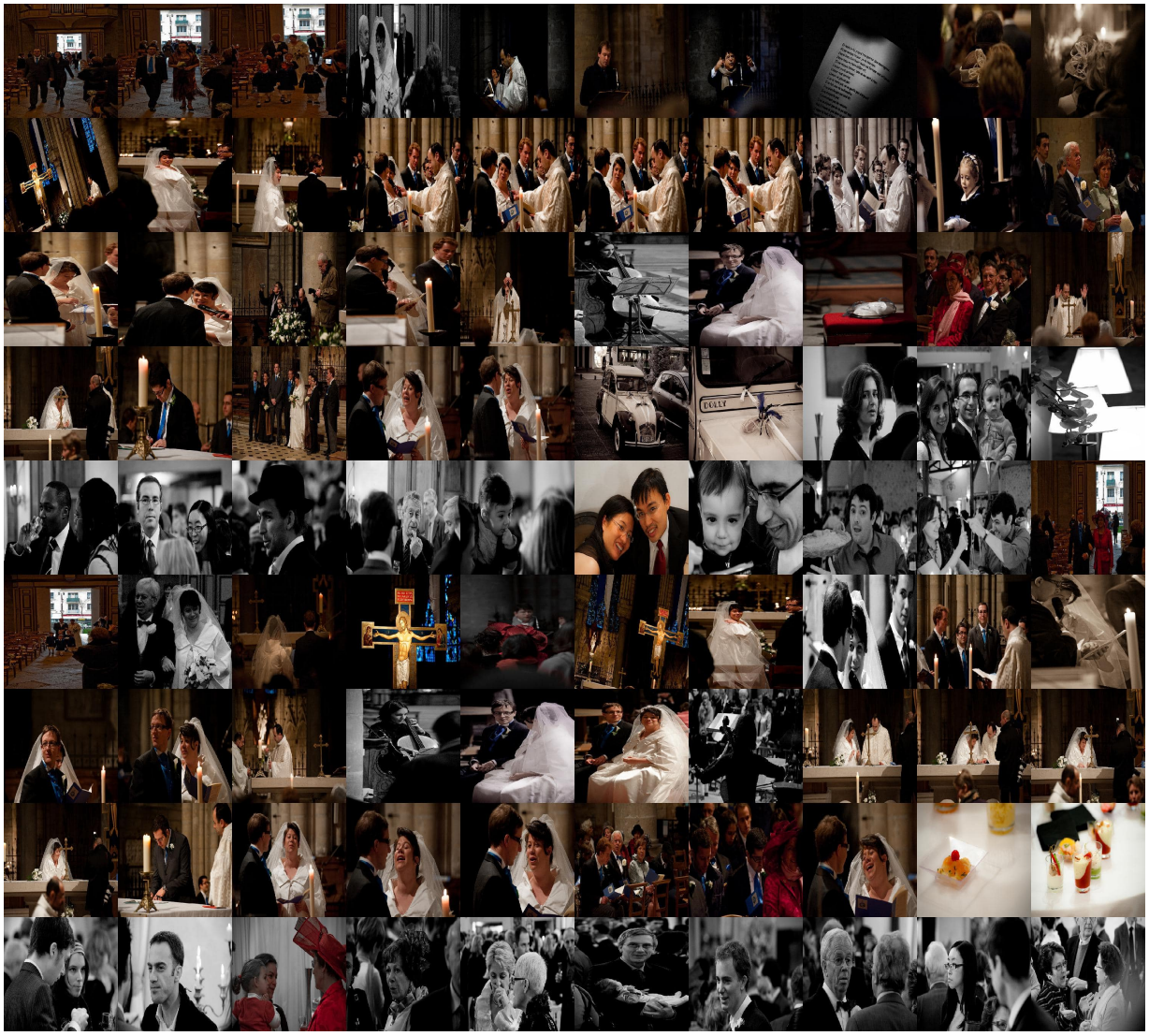


Figure 2.2: An example of albums in our dataset and the Spearman's Correlation ρ and Kendall's W from worker's rating for each album: A *Wedding* album. Spearman's Correlation $\rho = 0.78$, Kendall's $W = 0.64$

Figure 2.3: An example of albums in our dataset and the Spearman's Correlation ρ and Kendall's W from worker's rating for each album: A *Birthday* album. Spearman's Correlation $\rho = 0.61$, Kendall's $W = 0.49$

Figure 2.4: An example of albums in our dataset and the Spearman's Correlation ρ and Kendall's W from worker's rating for each album: A *Zoo/Botanic garden* album. Spearman's Correlation $\rho = 0.02$, Kendall's $W = 0.19$

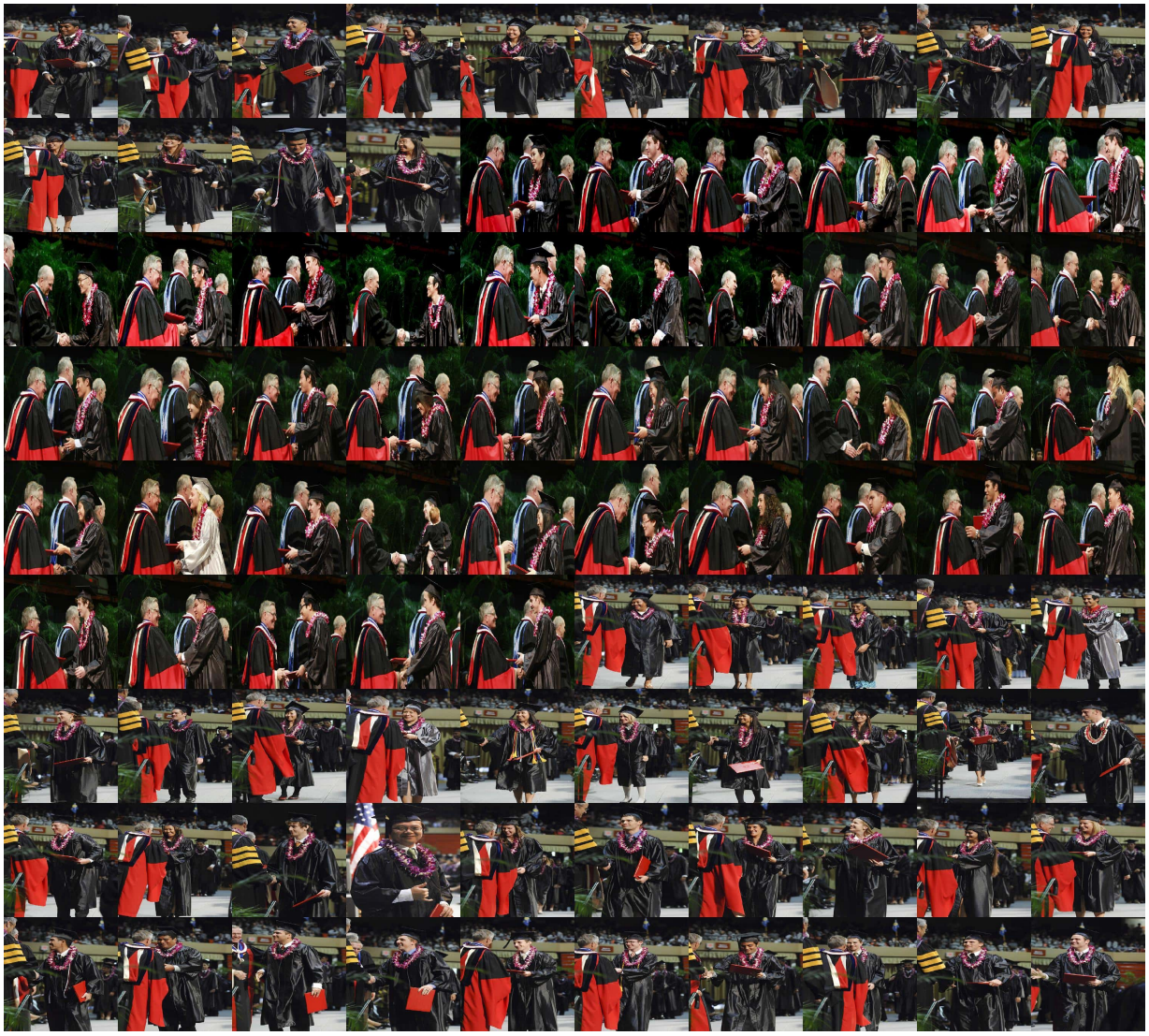


Figure 2.5: An example of albums in our dataset and the Spearman's Correlation ρ and Kendall's W from worker's rating for each album: A *Graduation* album. Spearman's Correlation $\rho = -0.09$, Kendall's $W = 0.17$

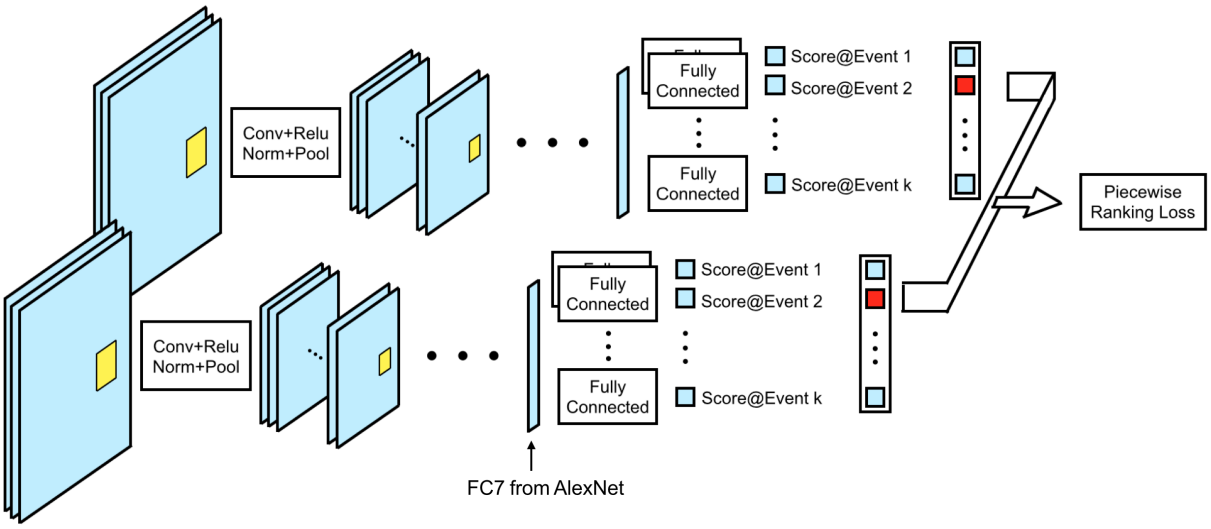


Figure 2.6: A siamese CNN architecture for joint training over events. A pair of images from the same album is the input to the two pathways. The network computes an importance score for its input image; only the units corresponding to the correct event type are activated and back-propagated through.

Second, feature sharing will significantly reduce the number of parameters in the network and regularize the network training. Especially for our problem, high variance among albums within each event type and relatively small datasets make this even more necessary. Therefore, in our network, all event types share the features, while the output level has event-specific ratings. During the training process, only the output corresponding to the event type of an image pair receives an error signal, and we assume that we know the event type at test time.

2-stage progressive training

Due to the large variation among albums and the relatively small scale of the dataset (especially for some event types such as *casual family/friends gathering*), directly training a CNN for separate event types as in Section 2.4.1 may lead to over-fitting for some event types with less training data. Therefore, we use a 2-stage progressive learning method: we train all images with one output for the whole network; and then switch to training with a separate output

for each event type. Initialization of the second stage is done using the network from the first stage. This helps in that i) the features that are useful for all event types are learned first, using all of the data; ii) the individual event-type output units are initialized with the weights from the one-output unit, so they already have some knowledge of what makes an important image; and iii) the discrimination is then refined based on the properties of individual event types. Some pictures are just excellent no matter what the occasion; our two-stage learning system leverages that intuition[1].

Siamese architecture

There is large variation in the quality of the albums within an event type, which might bias the judgment of participants in our AMT task. Therefore it is difficult to learn a reliable absolute image importance score that is suitable for different albums. Meanwhile, the relative importance ranking of images within the same album is more meaningful and more practical in applications. Hence, rather than training on an absolute image score, we use the average score difference between a pair of images from the same album to train the network. This is the motivation for using the siamese network architecture [103], which processes pairs of images. In the siamese network, the two pathways share weights, so a common representation is learned (see Fig. 2.6).

Piecewise ranking loss

For each input image pair to the network $(\mathrm{I}_1, \mathrm{I}_2)$, $G(\mathrm{I}_i)$ is the ground truth score of image I_i, and $P(\mathrm{I}_i)$ is its predicted score from the network. We use a piecewise ranking loss (PR loss) to train the network:

[1] We also tried to cluster the event types into k "superclasses" according to their similarity, and to use the superclass information for the first stage training. However, that didn't lead to a better result. One possible reason is that our event type clustering algorithm does not perform well.

$$\text{PR} = \begin{cases} \frac{1}{2}\max(0, \left|D_\text{p}\right| - \text{m}_\text{s})^2 & \text{if } D_\text{g} < \text{m}_\text{s} \\ \frac{1}{2}\left\{\max(0, \text{m}_\text{s} - D_\text{p})^2 + \max(0, D_\text{p} - \text{m}_\text{d})^2\right\} & \\ & \text{if } \text{m}_\text{s} \leq D_\text{g} \leq \text{m}_\text{d} \\ \frac{1}{2}\max(0, \text{m}_\text{d} - D_\text{p})^2 & \text{if } D_\text{g} > \text{m}_\text{d} \end{cases} \tag{2.1}$$

where $D_\text{g} = G(\text{I}_1) - G(\text{I}_2)$ is the ground truth score difference between the input image pair, and $D_\text{p} = P(\text{I}_1) - P(\text{I}_2)$ is the predicted score difference. m_s and m_d are predefined values for similar and different margins. In Equation 2.1, several conditions are considered:

- When $D_\text{g} > \text{m}_\text{d}$, the loss function reduces to a variation of ranking SVM hinge loss [18]. We use L-2 loss which penalizes high errors more heavily than traditional hinge loss [112]. This is similar to contrastive loss function when the input pair of images are deemed dissimilar [46], but we are not using the euclidean distance of the output of the network, since the sign of D_p is important here.

- When $D_\text{g} < \text{m}_\text{s}$, the loss function reduces to a variation of contrastive loss when the input pair is deemed similar [46]. In addition to the contrastive loss in [46], we introduce a margin: m_s. The margin serves as a slack term. The reason to have it is that the ground truth importance score is acquired from a group of humans, and the variance is relatively high among the humans, as shown in Section 2.3.3. The introduction of relaxation with m_s makes the network less sensitive to this variance in our ground truth.

- When $\text{m}_\text{s} < D_\text{g} < \text{m}_\text{d}$, the loss function will only penalize the D_p not being in the same range with D_g. This pulls D_p towards D_g when the image pair is similar in rating, reducing the loss function's vulnerability to the variance in our ground truth.

The PR objective loss function has the following advantages: Rather than training only on images with different ratings, it provides an error signal even when image pairs have the same

rating, moving them closer together in representational space. This makes full use of the training dataset. Our piecewise version also introduces relaxation in the ground truth score, thus making the network more stable, which is beneficial when the ratings are subjective.

2.4.2 Incorporating face heatmaps

Images with faces tend to be more interesting than images without them[101]. Moreover, our intuition is that in an event album, important people will appear more frequently. This across-album feature cannot be captured by a CNN trained with image pairs. In order to incorporate face information, we generate face heatmaps, and use them to train a shallow CNN to independently predict the importance score of the photos. A separate face heatmap-based score enables flexible tuning of the relative strength of the two scores from original images and face heatmaps.

To generate the face heatmaps, we use a state-of-the-art face detection network [72]. In order to modulate the heatmaps according to face frequency, we need facial identity information. We train 18 CNN models for different face parts and concatenate the final fully-connected layers as the final face descriptor, following a similar pipeline as [104]. We then do agglomerative identity clustering to obtain the frequency of faces in an album. In the face heatmap, faces are represented with Gaussian kernels, and the two most frequent faces are emphasized by doubling their peak values. These are used as input to a shallow siamese CNN trained from scratch, with one convolutional layer and two fully connected hidden layers, in the same manner as the image network. In Figure 2.7, we show the architecture we used for Face Heatmap network.

Examples of face heatmaps are shown in Fig. 2.8. In the testing stage, the prediction from the original image and the face heatmap network are combined according to the following formula:

$$P = P_{\mathrm{I}} + \lambda \cdot \min\left\{\max\left\{P_{\mathrm{f}}, \beta\right\}, \alpha\right\} \tag{2.2}$$

where $(\mathrm{P_I}, \mathrm{P_f})$ are predicted scores from the original photo network and face heatmap network

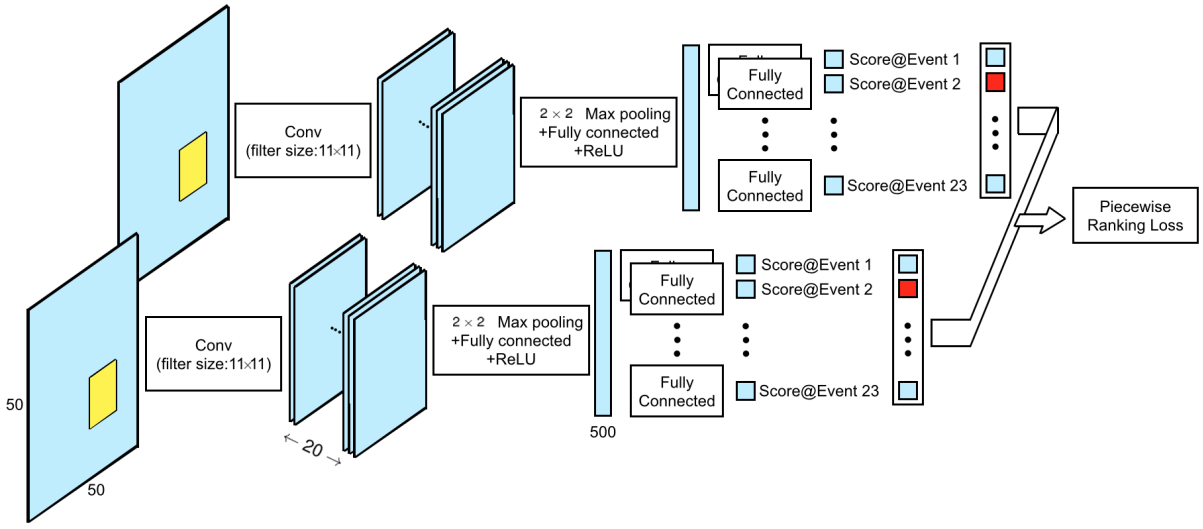


Figure 2.7: Face Heatmap CNN architecture.

respectively. The face heatmap contains a limited information, therefore we constrain the effect of the face heatmap for the final prediction with (α, β), so that extreme predictions from the face heatmap are eliminated; λ is also used to further control the effect of the face heatmap-based prediction. These parameters are set using cross-validation and a grid search.

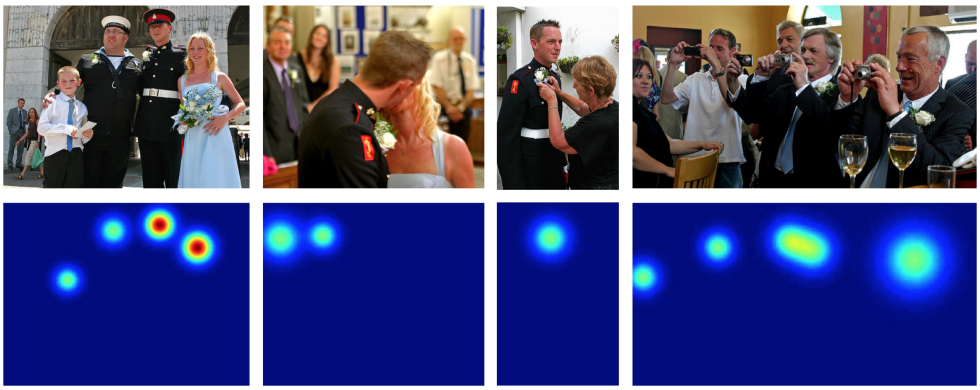


Figure 2.8: Face heatmaps from a wedding event album. First row: original images; Second row: face heatmaps. Faces of the two most important people have higher peak values (red dots). The second column shows that face detection is not ideal; the third column shows that identity clustering is not perfect.

2.5 Experimental Results

In this section, we compare our results with several baseline methods.

2.5.1 Experimental settings

Dataset

For training and testing, we randomly split the Curation of Flickr Events Dataset into 3:1 albums for every event type. The training set consists of 1404 albums, and the test set has 479 albums.

Parameter setting

We use Alexnet to initialize the CNN architecture and then fine-tune it [65, 58]. In Fig. 2.6, FC7 is from Alexnet, driving the event-specific sigmoidal score prediction layer. We assume we know the event type, and the teaching signal is masked by the correct event. For PR loss, we set $m_s = 0.1$ and $m_d = 0.3$. For training parameters, we use the default settings for pre-training in Caffe [58], but we start from a smaller learning rate of 0.001 [41].

We follow [65]'s data augmentation approach: Input images are resized to 256×256. During the training stage, images are randomly cropped to 227×227 crops, and there is a 50% probability that input images are horizontally flipped. In the test stage, predictions are averaged on five crops (four corners and the center) and their horizontal reflections. We train five different CNNs with 5-fold cross validation, and use an ensemble of the five networks for the final prediction.

Evaluation metrics

We use two evaluation methods to compare the different approaches. For both evaluation methods, we assume that given an event album, we view the top $t\%$ images as relevant images,

and measure the metric at various values of t.

First, we use mean average precision (MAP) to evaluate our models. MAP is a common evaluation method for information retrieval [4]. It is the averaged area under the precision-recall curve over all albums. Given the collection of albums, and top $t\%$ of the images as being relevant images, MAP@$(t\%)$ can be calculated:

$$\text{AP(S)}@t\% = \int_0^1 p(r)d(r) \approx \frac{\sum_{k=1}^{n} p(k) \times \text{rel}(k)}{\lceil n \cdot t\% \rceil} \tag{2.3}$$

$$\text{MAP(U)}@t\% = \frac{1}{N}\sum_{i=1}^{N} \text{AP}(\text{S}_i)@t\% \tag{2.4}$$

where S_i is the ith album, and U is the collection of all albums. n is the size of album S, $p(k)$ is the precision at rank k, and rel is an indicator of whether the kth ranked image from our algorithm is a relevant image, i.e. among the top t % ground truth.

Second, we calculate the precision (P), the ratio between the number of relevant photos in the retrieved images over the total number of relevant images at each level of t. Unlike MAP, P cares entirely about how many important images can be retrieved at a cut-off level, and does not care about the position they are in the retrieval list, or where the rest of important images are in the ranking system. Although less informative than MAP, P is also an intuitive way to demonstrate the effectiveness of our predicted image ranking result. Since we are solving an image selection problem, we care more about MAP and P for small $t\%$, so we only present results for $t \leq 30$.

2.5.2 Results and analysis

In this section, we compare our method, Piecewise Ranking-CNN trained progressively (PR-CNN(Progressive)), on all event types to various baselines, and demonstrate the advantages of our method (see Table 2.3 and Figure 2.10).

Figure 2.9: Example results for one wedding album. Top 5 images of the album from different methods are shown here. First row: Ground truth acquired from AMT workers; Second row: Our prediction using Ensemble-CNN; Third row: Random selection.

Figure 2.9 is an example to show how our algorithm performs intuitively. Our result clearly learns meaningful concepts for the wedding event.

Table 2.3: Comparison of predictions using different methods. Evaluation metric here is MAP@t% and P@t%. Random ranking score is also shown as a lower bound.

	MAP@t%						P@t%					
t%	5	10	15	20	25	30	5	10	15	20	25	30
Random	0.122	0.164	0.211	0.260	0.305	0.350	0.058	0.093	0.141	0.195	0.251	0.298
Worker	0.328	0.410	0.476	0.531	0.580	0.624	0.242	0.371	0.448	0.505	0.552	0.591
Aesthetic	0.139	0.191	0.242	0.290	0.338	0.384	0.060	0.121	0.176	0.228	0.284	0.335
Pre-KNN	0.220	0.276	0.326	0.373	0.419	0.465	0.138	0.216	0.275	0.326	0.372	0.419
Pre-SVM	0.252	0.320	0.370	0.420	0.466	0.512	0.169	0.262	0.318	0.363	0.410	0.458
Euclidean	0.266	0.329	0.389	0.444	0.494	0.540	0.173	0.260	0.328	0.391	0.439	0.485
SVM-CNN	0.266	0.337	0.396	0.451	0.500	0.546	0.172	0.280	0.345	0.402	0.447	0.491
NoEvent-CNN	0.261	0.318	0.369	0.422	0.474	0.520	0.167	0.247	0.310	0.372	0.425	0.468
PR-CNN(Direct)	0.296	0.358	0.410	0.462	0.511	0.557	0.199	0.293	0.352	0.403	0.454	0.498
PR-CNN(Progressive)	0.302	0.361	0.415	0.469	0.517	0.563	0.214	0.296	0.356	0.410	0.458	0.502
Ensemble-CNN	0.305	**0.364**	0.417	0.471	0.519	**0.563**	**0.216**	0.301	**0.360**	0.411	0.459	**0.504**
Ensemble-CNN + face	**0.306**	**0.364**	**0.418**	**0.472**	**0.520**	**0.563**	0.215	**0.303**	**0.360**	**0.413**	**0.460**	0.503

In Figure 2.10, we show the comparison of MAP@t%5 by six methods for each of the 23 event types. The six methods being compared are: random ranking, aesthetics, K nearest neighbors with pre-trained CNN features (KNN), single network with Euclidean loss (Euclidean), siamese network with ranking SVM loss (Ranking-SVM), and our method using Ensemble of siamese CNNs (Ensemble). We will explain the baseline methods in detail in the following section. We also show a "worker" method here for comparison. It is calculated as follows: for each album, we have 5 rankings from 5 workers, and we can calculate the MAP score for each worker's rating against the ground truth. Then all the MAPs over all albums are averaged for one event type. The "worker" method is to measure how workers did on those albums.

As shown, our method outperforms all the other methods in most cases, except for *Personal Art Activity*, *Architecture*, *Business Activity*, *Protest* and *Nature Trip*. Our method can even beat "worker" in some cases.

In the following sections, we describe the various baseline methods we benchmark our system against. To make a long story short, we achieve our best result using an ensemble of the five PR-CNN(Progressive) networks with face information (See Table 2.3).

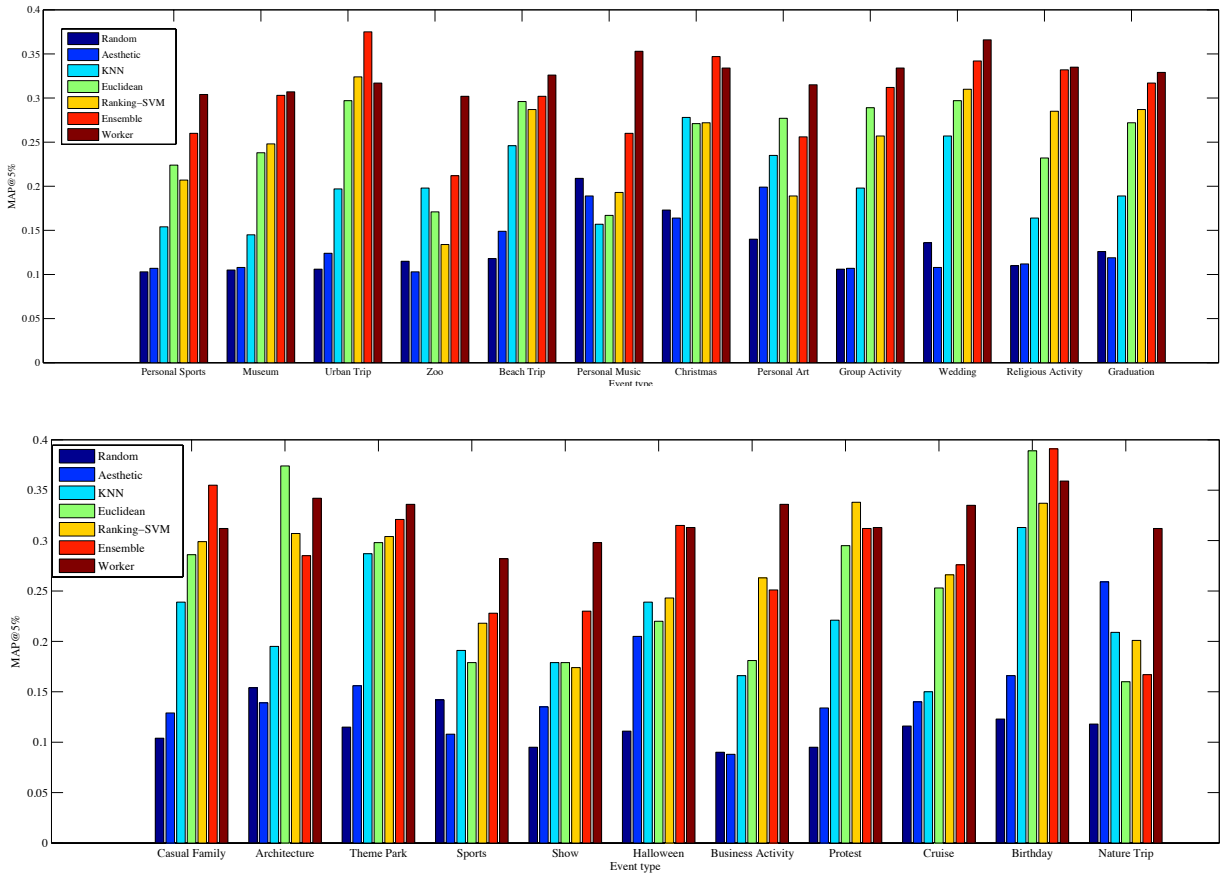


Figure 2.10: Comparison of six methods for 23 event types respectively. Individual worker's performance is also included as comparison. Results of MAP@t%5 are shown.

Does aesthetics play an important role?

In a user study, Walber *et al*.show that that humans use the visual appeal of an image as a criterion for selecting important images in an album [120]. Here, in order to quantify the role attractiveness plays in the selection, we use an aesthetic score prediction method instead of the importance score. We train a CNN classifier similar to [79], using aesthetic scores collected using AMT.

Table 2.3 shows that the aesthetic score of images is only slightly better than random. We conclude that aesthetics, at least using this method, is not a very important criterion for human selection of important images in event albums.

As shown in Figure 2.10, we observe that the aesthetic score is more predictive for some events than others, e.g. *Nature trip*, *Personal art activity* (in which many photos are portrait shots). Especially for *Nature Trip*, aesthetics achieves the best performance over all methods. This is consistent with our intuition: aesthetics is an important criterion for human selection in events without strong narrative structure.

Are pre-trained CNN features useful?

Pre-trained CNN features have been shown to have a high generalization ability to new tasks [16, 41]. Using the FC7 layer of Alexnet [58, 65] as our feature vector, we apply a K-NN classifier and a Ranking-SVM classifier.

For the KNN approach, we perform a 10-nearest neighbors search against all training images in the same event type, and use the weighted average of the 10 images' ground truth importance score, where the weight is the image's similarity score to the query test image. We denote this method as Pre-KNN. We also train 23 Ranking-SVMs (one for each event type) on pairs of the 4096-d feature vectors. This method is denoted Pre-SVM.

Table 2.3 shows the results of using pre-trained CNN features. The KNN method significantly outperforms the aesthetic score and random ranking. However, it is still much lower

than our proposed method. This shows that the high variation of albums makes the direct score prediction using images in other albums with similar visual appearance unreliable. The Pre-SVM method performs better than the KNN method, but the improvement is limited.

The results of the above two experiments verify that the pre-trained CNN features can generalize to some extent to the event-based image importance prediction problem.

Is Piecewise Ranking loss necessary?

In order to show the advantage of PR loss, we compare our results with the results trained from a conventional ranking SVM hinge loss. For the SVM ranking loss, the network architecture is exactly the same as our proposed method except for the loss function:

$$\mathrm{L}(\mathrm{I}_1, \mathrm{I}_2) = \max(0, 1 - D_{\mathrm{p}}) \tag{2.5}$$

where $D_{\mathrm{p}} = P(\mathrm{I}_1) - P(\mathrm{I}_2)$ is the predicted score difference between the image pair.

This method is denoted as SVM-CNN. As shown in Table 2.3, PR loss (PR-CNN(direct)) outperforms Ranking SVM hinge loss (SVM-CNN) especially when $t < 20$. Ranking SVM uses 87% of image pairs as the training data compared to PR loss, because it does not use the image pairs with the same score. The reason for PR's better performance may be due to differences in the loss function or because it has 15% more training data.

We also tried a single network with Euclidean Loss to directly predict the importance of a single image. As shown in Table 2.3, the result is denoted as Euclidean, and they are consistently worse than SVM-CNN by about 0.6%.

Is event information useful?

In the previous work on album summarization or photo selection, a common approach is to use general image interestingness/quality to represent the image importance score irrespective of

the event type of the album [15, 95, 101]. We propose that event type information is an important factor in determining the image importance score, and that using 2-stage learning will help with the prediction. In this section, we verify our proposal by comparing the performance of CNNs trained i) without the event type information, ii) with 2-stage learning, and iii) with only the second stage learning on 23 event types.

We train a CNN with exactly the same architecture and training parameters except that the last layer of each of the halves of the siamese network in Fig 2.6 is one unit, so there is essentially one ”superclass” event type. This method is denoted as No Event CNN (NoEvent-CNN). As shown in Table 2.3, although trained with the same loss, without event type information, the network performs worse than PR-CNN(Progressive) by a large margin of 4% over the MAP scores. In addition, the difference of $P@t\%$ is especially large for smaller t, which is the region of most importance.

We also train a CNN with only the second stage directly on 23 event types, as PR-CNN(Direct). Table 2.3 shows the performance gain using 2-stage learning is about 0.6% on MAP score. This difference is consistent across our experiments. Again, our best result is with an ensemble of the PR-CNN(Progressive) networks (Ensemble-CNN).

Incorporation of face information

In order to incorporate the face information, we use 5-fold cross validation on the training set to set the parameters $\{\alpha, \beta, \lambda\}$ in Equation 2.2 using a grid search.

Among 23 event types, only 10 event types showed a performance gain after face information was incorporated in the validation set, and therefore face information was only used for these 10 event types. Table 2.4 shows the effect of incorporating face information for these 10 event types.

Among 23 event types, only 10 event types show a performance gain after face information is incorporated in the validation set, and thus the face information is used for only these 10 event

types on the test set. Table 2.4 shows the effect of incorporating face information for these 10 event types. As shown, for some event types, face information substantially helps performance, while for other event types, face information has little impact, or even harms performance. In summary, counter to our expectation, our method for incorporating face information has little effect on performance, increasing it by about 0.1%, which is not likely to be significant.

Table 2.4: For a given event type, MAP@t% for the Ensemble-CNN after using the face information. The difference between before v.s. after face information is shown in parentheses. All the 10 event types for which face information is used are shown here.

t%	5	15	25
Beach Trip	0.353(+0.051)	0.455(+0.022)	0.555(+0.011)
Nature Trip	0.167(+0.008)	0.272(+0.008)	0.369(+0.007)
Group Activity	0.315(+0.003)	0.489(+0.001)	0.586(+0.003)
Halloween	0.315(+0.000)	0.424(+0.001)	0.529(+0.002)
Personal Art Activity	0.256(+0.000)	0.361(0.002)	0.449(+0.000)
Religious Activity	0.320(-0.012)	0.416(0.000)	0.503(+0.005)
Graduation	0.317(+0.001)	0.444(0.002)	0.548(+0.001)
Sports	0.228(+0.001)	0.322(0.002)	0.420(+0.002)
Show	0.232(+0.002)	0.356(0.002)	0.473(+0.001)
Museum	0.293(-0.010)	0.367(-0.010)	0.453(-0.006)

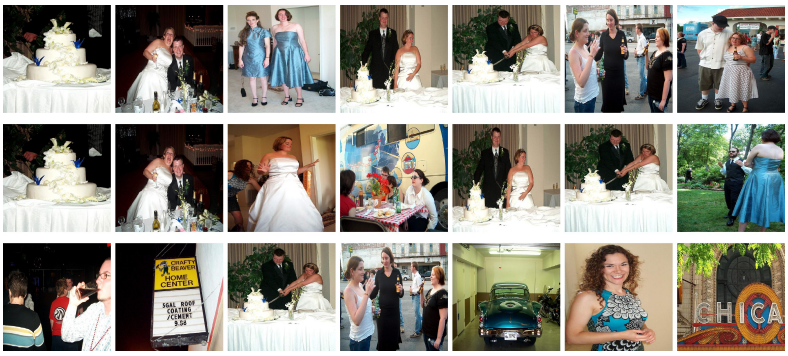
2.5.3 Qualitative results

In addition to the visual example of our method's performance, we show more examples of our method. Here we present 64 examples from all 23 event types from Figure 2.11a to Figure 2.13d. For each album, we show top 10-20% images of the album from three methods. (Each album has different size, while we want to constrain the number of images we show to make it easier to view.) First row is the ground truth we acquired from AMT worker; second row is our prediction using Ensemble-CNN; third row is the result from random selection. Note that the images are distorted for viewing.

We can see that for most albums that have strong narrative structure or albums that consist of images that vary much in quality or semantics, our method's results are close to, though do

not perfectly match the ground truth result; on the contrary, the results from random selection are obviously less appealing (for example, Figure 2.11a, 2.18a, 2.18b, etc.). For instance, in Figure 2.14a, our method captures the important moments of the wedding event, similar to those people picked (in the ground truth); however random selection has many images that are less important, for example, photos of people eating, or photos of guests talking, while not looking at the camera.

There are also some albums in which most of the images are of similar quality or semantics, for example, Figure 2.12a, 2.21b.



(a) Top 20% of a *Wedding* album.

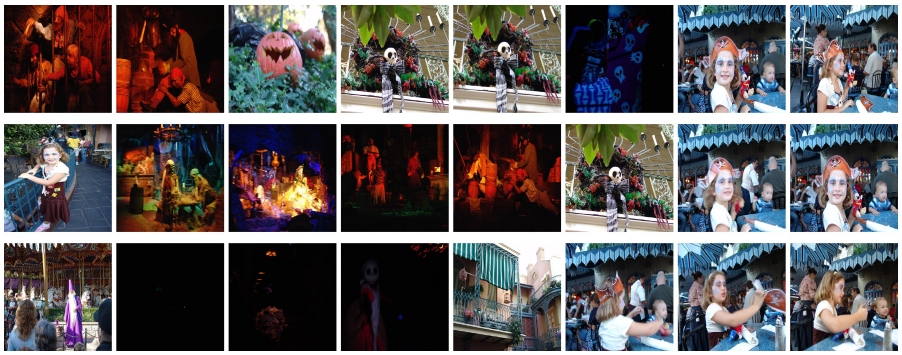
(b) Top 20% of a *Museum* album.

(c) Top 10% of a *Graduation* album.

Figure 2.11: Examples of results. For each album, top 10-20% images of the album from three methods are shown. First row is the ground truth we acquired from AMT worker; second row is our prediction using Ensemble-CNN; third row is the result from random selection.

(a) Top 20% of a *Personal Sports* album.

(b) Top 20% of a *Birthday* album.



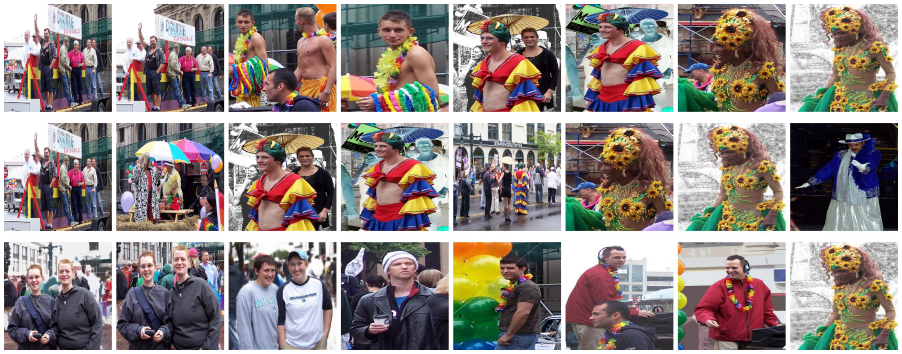
(c) Top 10% of a *Halloween* album.

(d) Top 20% of a *Sports* album.

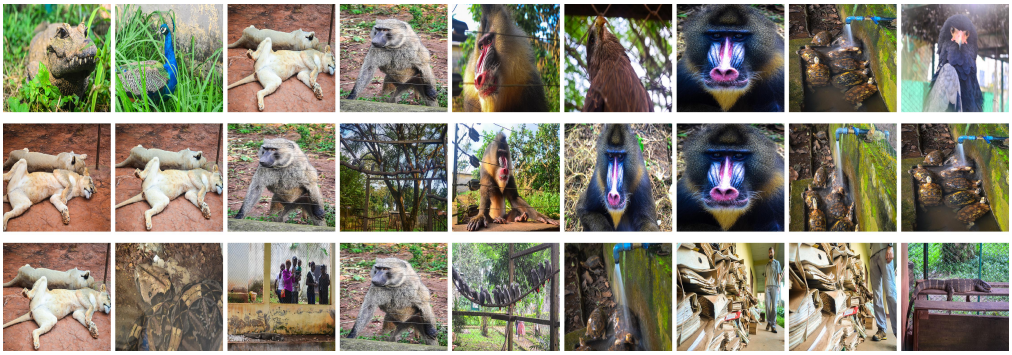
Figure 2.12: Examples of results.

(a) Top 20% of a *Personal Music Activity* album.

(b) Top 20% of a *Halloween* album.



(c) Top 20% of a *Show* album.



(d) Top 20% of a *Zoo* album.

Figure 2.13: Examples of results.

(a) Top 20% of a *Wedding* album.

(b) Top 15% of a *Cruise Trip* album.

(c) Top 20% of a *Wedding* album.

(d) Top 20% of a *Religious Activity* album.

Figure 2.14: Examples of results.

(a) Top 15% of a *Causal Family/Friends Gathering* album.

(b) Top 20% of a *Wedding* album.

(c) Top 20% of a *Birthday* album.

(d) Top 20% of a *Halloween* album.

Figure 2.15: Examples of results.

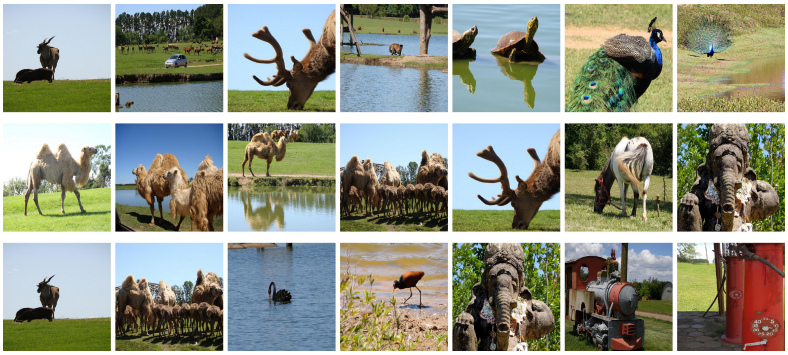
(a) Top 20% of a *Birthday* album.

(b) Top 20% of a *Wedding* album.

(c) Top 20% of a *Personal Art Activity* album.

(d) Top 15% of a *Theme Park* album.

Figure 2.16: Examples of results.



(a) Top 20% of a *Zoo* album.

(b) Top 20% of a *Halloween* album.

(c) Top 15% of a *Graduation* album.

(d) Top 10% of a *Graduation* album.

Figure 2.17: Examples of results.

(a) Top 15% of a *Urban Trip* album.

(b) Top 20% of a *Causal Family/Friends Gathering* album.

(c) Top 10% of a *Graduation* album.

(d) Top 20% of a *Architecture* album.

Figure 2.18: Examples of results.

(a) Top 20% of a *Urban Trip* album.

(b) Top 20% of a *Business Activity* album.

(c) Top 20% of a *Business Activity* album.

(d) Top 20% of a *Wedding* album.

Figure 2.19: Examples of results.

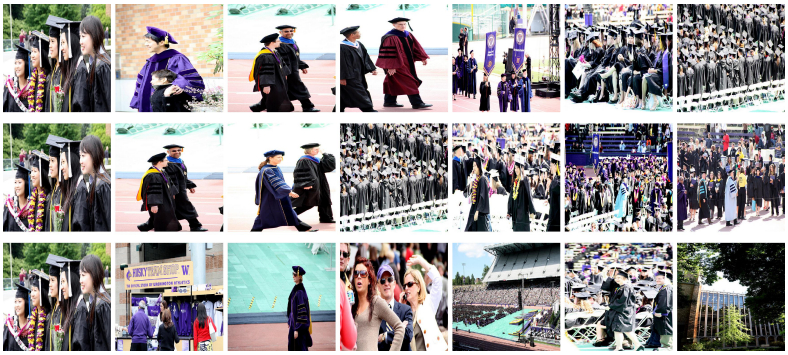
(a) Top 20% of a *Birthday* album.

(b) Top 20% of a *Architecture* album.

(c) Top 20% of a *Wedding* album.

(d) Top 20% of a *Zoo* album.

Figure 2.20: Examples of results.



(a) Top 10% of a *Graduation* album.

(b) Top 20% of a *Museum* album.

(c) Top 20% of a *Beach Trip* album.

(d) Top 10% of a *Nature Trip* album.

Figure 2.21: Examples of results.

(a) Top 10% of a *Zoo* album.

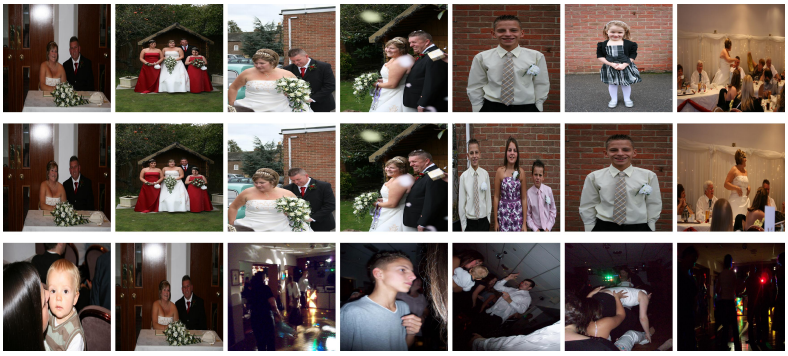
(b) Top 20% of a *Zoo* album.

(c) Top 15% of a *Zoo* album.

(d) Top 10% of a *Zoo* album.

Figure 2.22: Examples of results.

(a) Top 15% of a *Protest* album.



(b) Top 20% of a *Wedding* album.

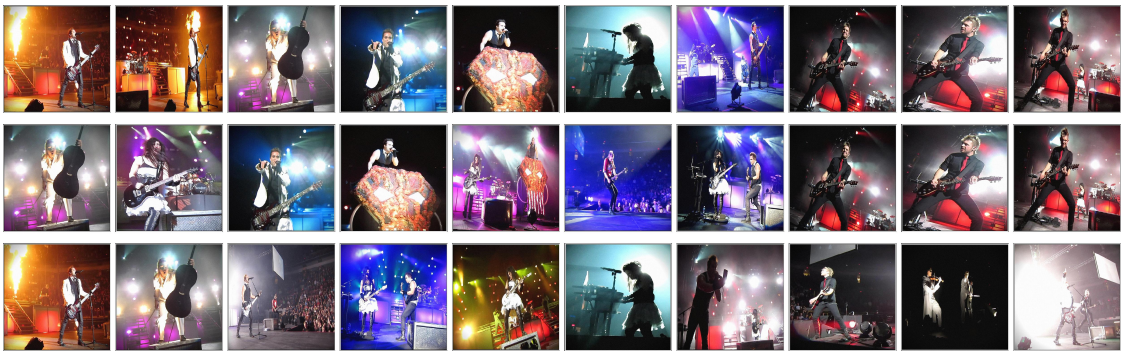
(c) Top 20% of a *Christmas* album.

(d) Top 20% of a *Museum* album.

Figure 2.23: Examples of results.

(a) Top 15% of a *Birthday* album.

(b) Top 20% of a *Personal Sports* album.



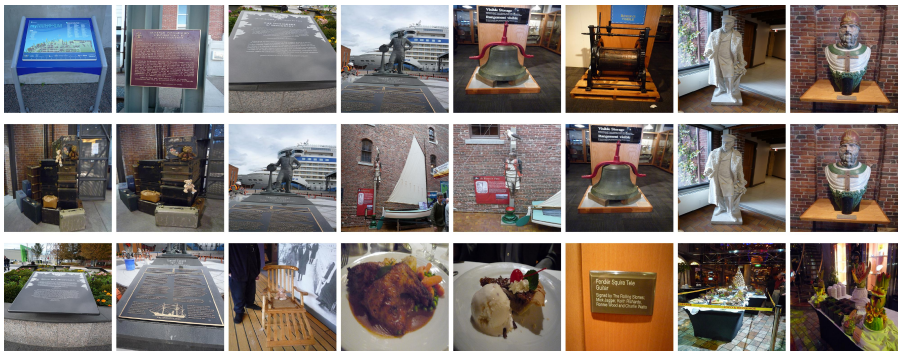
(c) Top 20% of a *Show* album.

(d) Top 15% of a *Theme Park* album.

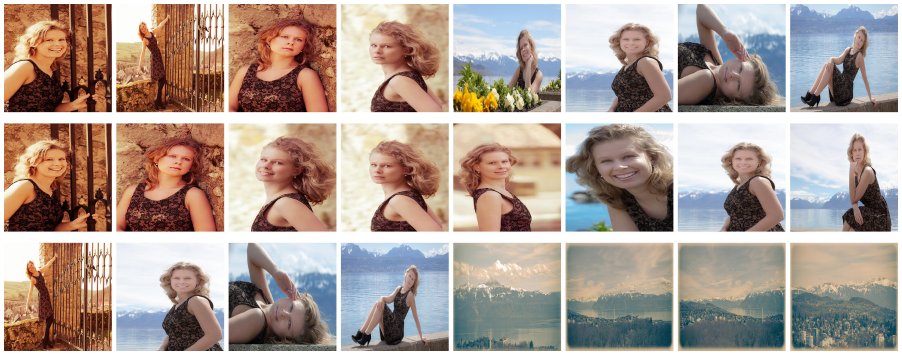
Figure 2.24: Examples of results.

(a) Top 20% of a *Sports* album.

(b) Top 20% of a *Wedding* album.



(c) Top 10% of a *Museum* album.



(d) Top 20% of a *Personal Art Activity* album.

Figure 2.25: Examples of results.

(a) Top 15% of a *Nature Trip* album.

(b) Top 20% of a *Theme Park* album.

(c) Top 20% of a *Theme Park* album.

Figure 2.26: Examples of results.

2.6 Conclusion

In this work, we introduce a new image property: event-specific image importance. We provide a new dataset consisting of common personal life events, and we provide human generated image importance score ground truth for the dataset. We provide evidence that although the event-specific image importance score is subjective, it is a well-defined and predictable property: there is consistency among different subjects. We develop a CNN-based system to predict event-specific image importance. We show that although aesthetics is usually considered in an image selection system, it is not the most important criterion for people. More importantly, we also show that the event information is an important criterion when people select important images in an album. In our prediction system, we design a Piecewise Ranking Loss for a dataset with subjective or high variance ground truth, and we use a 2-stage progressive training process to train the network. We show that our system is advantageous over the conventional Ranking SVM loss and training procedure.

This work is the first attempt to predict event-specific image importance. This image property is especially useful in album summarization and image selection from an album. In future work, it will be interesting to further investigate the relationship between event types, and to deal with albums with multiple/ambiguous event types. Also, we plan to develop a curation system based on the image importance score, taking diversity and coverage into consideration. Our Curation of Flickr Events Dataset will be made public to facilitate the study of this topic.

2.7 Acknowledgements

Chapter 2, in full, is an edited reprint of the material as it appears in the following publication: Wang, Y., Zhe, L., Shen, X., Měch, R., Miller, G., Cottrell, G. W., "Event-specific Image Importance", In *Computer Vision and Pattern Recognition (CVPR)*, 2016. The dissertation author was a primary researcher and an author of the cited material.

This work is supported in part by NSF grant SMA 1041755 to the Temporal Dynamics of Learning Center (TDLC), an NSF Science of Learning Center (GWC and IG), NSF grant IIS-1219252 to GWC, and a gift from Adobe Research.

Chapter 3

Recognizing and Curating Photo Albums via Event-Specific Image Importance

Chapter 2 shows that in a personal event album, image importance score is related to the event type of the album it is in, and is predictable by statistical models. However, in Chapter 2, to obtain the image importance score of images in an album, the event type information is assumed known in advance. In this chapter, I develop an iterative model that can learn the event type of the album and image importance score simultaneously.

Automatic organization of personal photos is a problem with many real world applications, and can be divided into two main tasks: recognizing the event type of the photo collection, and selecting interesting images from the collection. In Chapter 2, we describe our model to find interesting images from an album collection, with learning of the image property: event-specific image importance. In this chapter, we attempt to simultaneously solve both tasks: album-wise event recognition and image-wise importance prediction. We collected an album dataset with both event type labels and image importance labels, refined from an existing CUFED dataset. We propose a hybrid system consisting of three parts: A siamese network-based event-specific image importance prediction, a Convolutional Neural Network (CNN) that recognizes the event type, and a Long Short-Term Memory (LSTM)-based sequence level event recognizer. We propose an iterative updating procedure for event type and image importance score prediction. We experimentally verified that image importance score prediction and event type recognition can each help the performance of the other.

3.1 Introduction

With the advent of cheap cameras in nearly all of our devices, automated uploading to the cloud, and practically unlimited storage, it has become painless to take photos frequently in daily life, resulting in an explosion of personal photo collections. However, the oversized image collections make it difficult to organize the photos, and thus automatic organization algorithms are highly desirable. The organization of personal photo collections can be decomposed into

two stages: recognizing the event type of a photo collection, and suggesting the most interesting/important images in the photo collection to represent the album. The two stages can assist users in keeping the photo collections organized and free of irrelevant images, and can be further used to pick photos for an album cover or to make a photo collage.

Both event recognition and image importance prediction have been studied independently in previous literature. Studies of event recognition fall into three types. The most popular approach uses videos as input [122, 88, 111, 127, 132, 85, 37], and spatiotemporal features are commonly used. Further, event recounting which aims to find the event-specific spatial/temporal discriminative parts of a video is also studied [37]. This is relevant to event-specific image importance, but the image importance of an image is not decided by how discriminative it is. At the other end of the spectrum, event recognition for single images has also been studied [73, 90, 97, 121]. There is no temporal information or relevant frame importance to consider, and both object and scene level features have been used [73].

Album-wise event recognition lies between single-image-based and video-based event recognition, and is most related to our work. Images in an album can be thought of as very sparse samples from an event video, and consecutive images from the photo album are no longer continuous. A common approach is to aggregate evidence from single images to classify the album type [82, 114, 3, 124]. For example, Wu *et al*.[124] fine-tune Alexnet to extract features from single image, and then aggregate the features from each image and train a multi-layer network to recognize the event type of the album. The above work treats albums as unordered collection of images. On the other hand, Bossard *et al*.[11] exploit the sequential nature of personal albums, using an HMM-based sub-event approach (Stopwatch HMM) for event recognition. They use temporal sequence of the images, and model an album with successive latent sub-events to boost the recognition performance, and show that the temporally-sensitive HMM outperforms simply aggregating the predictions from all the images in an album. This indicates that the sequential information in an album is useful for album-wise event recognition.

Image importance is a complex image property which is related to various factors, such as aesthetics [79], interestingness [28] and image memorability [56]. In Chapter 2, we show that image importance is modulated by the context it is in, i.e., image importance is event-specific. For example, a photo of a beautiful work of architecture is important in an album of an urban trip, yet not so important in a wedding event. We showed that a siamese-network-based model can reasonably predict this highly subjective image property. However, our previous work assumed that the event type of the album is already known. This is undesirable if we want to build an end-to-end photo organization algorithm. In this work, we train a system simultaneously for event recognition and image curation, so that user input of the event type is not required.

Event recognition and image importance prediction are inherently related to each other: 1) importance is event-specific, so we need to know the event type to better predict importance; 2) albums often contain "outlier" photos that aren't directly related to the event. If we can reduce effects from the outliers by discovering the important/key images in an album, we can better recognize the event. Therefore, we ask the question: can we simultaneously recognize the event type of an album, and discover important images in it? And more importantly, can we improve the performance of each task by forming a joint solution?

In answering this question, this work makes the following contributions: 1) We develop a joint event recognition and image importance prediction algorithm.We use a CNN for image level event recognition, and a Siamese Network for event-specific image importance prediction. Then An iterative update scheme is used during the test stage, and we find that event recognition and image importance prediction can indeed improve each other's performance; 2) We further boost the performance of event recognition with an LSTM network that leverages sequential information in labeling the album; 3) We also refine the CUFED dataset by collecting more human annotations for the event types, allowing raters to apply multiple labels to the events. This improves the reliability of the ground-truth, accounting for the ambiguity between event types.

3.2 Related Work

Our work is closely related to the study of event recognition for a personal album. Mattivi *et al*.[82] classify a personal album into 18 events simply by aggregating SVM classification results from single images in the album. Tsai *et al*.[114] learn object patterns from single images, and then an album-wise SVM is trained on the frequency distribution of different object patterns appearing in an album. Bacha *et al*.[3] proposed a probabilistic graphical model to combine scene and object features for album classification. CNN-based models are also explored to aggregate features of an album for event prediction [124].

The above works treat albums as unordered collections of images. On the other hand, in [11], Bossard *et al*.exploit the sequential nature of personal albums and use an HMM based sub-event approach for event recognition. They use temporal sequence of the images, and model an album with successive latent sub-events to boost the recognition performance, and show that the temporally-sensitive HMM outperforms the simple aggregation of predictions from all the images in an album.

Event recognition for single photos has also been studied. Li *et al*.[73] use a generative graphical model to recognize event types of a database with 8 sports events. Their model integrates cues from scene and object categorization to classify the sports events. Salvador *et al*.[97] apply CNNs to cultural event recognition. They integrate cues from visual features extracted by a CNN with the time-stamp of a photo, inspired by the fact that photos of a cultural event are mostly taken in the same period of time. However, in personal photo collections, the relevance of an image within an event album varies a great deal. These approaches for single images are useful, but not sufficient for album-wise event recognition.

CNN methods have greatly boosted performance in image understanding tasks, such as image classification, object detection and scene recognition [65, 107, 41, 134]. Now many researchers have switched their focus to higher-level image properties, such as event recognition

[125], semantic segmentation [77], multilabel image annotation [42], and image captioning [30].

Long Short-Term Memory (LSTM) networks [50] have been proposed for sequence prediction and sequence labeling, and have achieved success for tasks such as handwritten text recognition [44] and speech recognition [96]. The success of LSTMs for sequence prediction tasks have also been extended to video-based event recognition. Reiter *et al*.[92] also combine LSTM and HMMs for video meeting analysis. Relevant to our work, Donahue *et al*.[30] proposed the Long-term Recurrent Convolutional Network (LRCN) model to stack CNN feature extractors and LSTM networks for sequential learning of videos or images.

3.3 The ML-CUFED Dataset

In order to train and evaluate the joint curation-recognition model, we use the Curation of Flickr Events Dataset (CUFED), and refine it by collecting additional human opinions on the event types in the dataset. We call the new dataset MultiLabel-CUFED (ML-CUFED). In this section, we describe the dataset, and provide a consistency analysis of the labels collected from Amazon Mechanical Turk (AMT). The dataset is available to the public.

3.3.1 The CUFED dataset

The CUFED dataset is an image curation dataset extracted from the Yahoo Flickr Creative Commons 100M dataset. It contains 1883 albums over 23 common event types, with 50 to 200 albums for each event type. The event type of each album was decided by 3 AMT workers' annotations. Meanwhile, within each album, the event-specific importance of each image is obtained by averaging 5 AMT workers' votes when the event type is given to them.

One problem with CUFED is that the event type of an album is decided by only 3 workers, who were constrained to give a single label to each album. However, some of the event types in that dataset are related (e.g., architecture and urban trip). For an album with ambiguous or

multiple event types, such a constraint is overly restrictive. For example, the two albums in Fig 3.1 are both birthday events, but they can also fall into the category of casual friends gathering. These two event types are not mutually exclusive. Moreover, intuitively, we would consider the album on the right to be a more typical birthday event, with distinguishable elements such as birthday hats and cakes, while the album on the left is more of a casual friends gathering rather than an obvious birthday event. Therefore, collecting the event types and their proportion in one album from more peoples' views is necessary. This results in a multi-label event recognition dataset with richer information.

Figure 3.1: Example of two birthday albums (both have the photo uploader's tag "birthday").

3.3.2 Data collection

In addition to the 3 votes the dataset already includes, we collected 9 more workers' opinions for each album, and allowed them to select up to 3 event types. There were 299 distinct workers who participated in the task.

Quality control was performed for each AMT worker in order to collect high quality annotations. Before the real task, only workers who passed a test that was very similar to the actual task were allowed to proceed. During the tasks, the results workers turned in were compared with other workers' submissions, and submissions that highly diverged from others were further manually inspected. If the divergence was unreasonable, the submission was rejected. After all the annotations from workers were collected, we further cleaned the annotations by eliminating the labels with only one vote. To get the final ground-truth event types, we converted the votes to

a probability distribution over event types for an album.

3.3.3 Dataset analysis

To check the validity of the dataset we collected, we analyzed the annotations in several ways. Each album has between 9 and 27 votes (because we allow for multiple choices from one worker). 76% of the albums received votes for two or fewer event types. 95% of the albums received votes for three or fewer event types. To check the consistency among workers, we randomly split the 299 workers into two halves, and for each album we checked whether the annotations from one half were consistent with the other half. We repeated the random split 100 times, and on average, for 89.6% of the albums, the event type receiving the most votes were the same for both groups. This suggests that despite the ambiguity of some album types, the opinions of different AMT workers are consistent.

Table 3.1: 23 Event types of ML-CUFED, and most frequent event type pairs of 2-label albums with their occurrence.

Categories	Event Types
All Event Types	Wedding, Birthday, Graduation, Protest, Personal Music Activity, Religious Activity, Casual Family/Friends Gathering, Group Activity, Personal Sports, Business Activity, Personal Art Activity, Architecture/Art, Urban Trip, Cruise Trip, Nature Trip, Theme Park, Zoo, Museum, Beach Trip, Show, Sports Game, Christmas, Halloween
Top 10 event types of two-label albums	(Personal Sports, Sports): 68, (Urban Trip, Architecture/Art): 27, (Zoo, Nature Trip): 22, (Show, Personal Music Activity): 22, (Casual Family/Friends Gathering, Group Activity): 17, (Birthday, Casual Family/Friend Gather): 16, (Halloween, Group Activity): 12, (Beach Trip, Cruise Trip): 8, (Show, Group Activity): 8

Table 3.1 shows all the 23 event types in ML-CUFED dataset, and the most frequent event type pairs of 2-label albums. Overall, there are 363 albums with multiple labels, about 20% of the ML-CUFED dataset.

In Figure 3.2, we show three examples of albums with multiple labels. These albums

(a) An example of a Birthday & Casual Family Gathering album. It was originally labeled as Birthday in CUFED.

(b) An example of a Christmas & Theme Park album. It is originally labeled as Christmas in CUFED.

(c) An example of an Urban Trip & Architecture/Art album. It is originally labeled as Architecture/Art in CUFED.

Figure 3.2: Examples of albums with multi-label in ML-CUFED, the original labels in CUFED are also shown. It is better to view digitally.

contain a mixture of different event types. For example, Figure 3.2(b) is a Christmas night event in a theme park (the fourth image in the second row shows ”Merry Christmas” with the Christmas lights, better seen if zoomed in), therefore the multi-label: (Christmas & Theme Park) is more reasonable than the single label Christmas.

3.4 Joint Event Recognition and Image Curation

In this section, we describe our approach to jointly attain image importance prediction and album event recognition. It is intuitive that important images contribute more to the identity of an event, and should be emphasized when deciding the event type of the album from the images. On the other hand, the identity of the event is needed for accurate individual image importance prediction, as shown in Chapter 2. Moreover, it has been shown that sequential information in an album is useful for event prediction [11]. Therefore, we build a joint system that can simultaneously predict the event type and image importance for an album. The system is shown in Figure 3.3. We elaborate on the different parts of the system in this section.

3.4.1 Event curation network

For event curation, we use a similar approach as in Chapter 2, using Piecewise Ranking Loss to train a Siamese network to predict the importance score difference between an image pair given the ground-truth event type. The Siamese network outperforms a traditional CNN that directly predicts the absolute image importance score. Compared to the architecture in Chapter 2, we added a pathway to directly predict the score difference between the image pair, rather than looking at the two images separately. This makes the training process faster and improves the results.

In Figure 3.4, we show the Curation-Siamese network used in the training stage for image importance prediction. Similar to Chapter 2, we use a siamese network to predict the importance

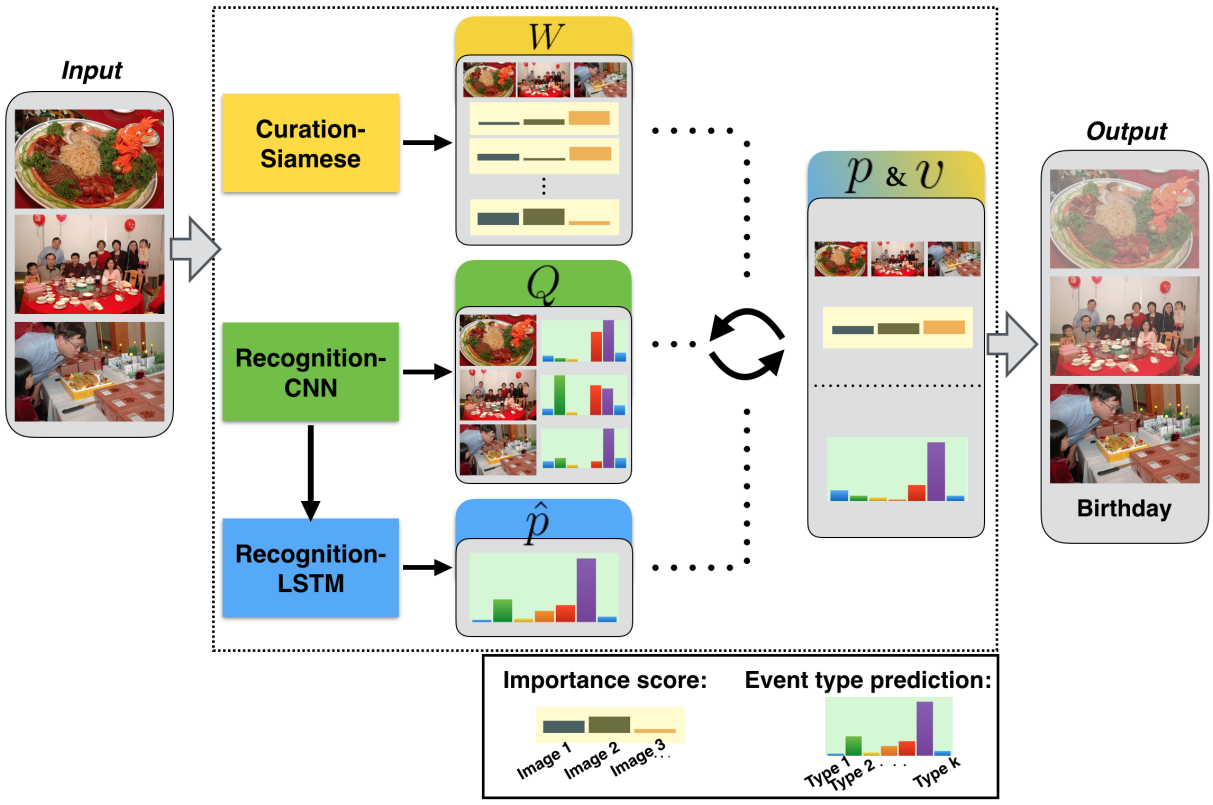


Figure 3.3: The joint album recognition-curation system. $\{W, Q, \hat{p}, p, v\}$ are described in Section 3.4.3. W, Q, and $\hat{p}$ are computed once and then used to iteratively update p and v.

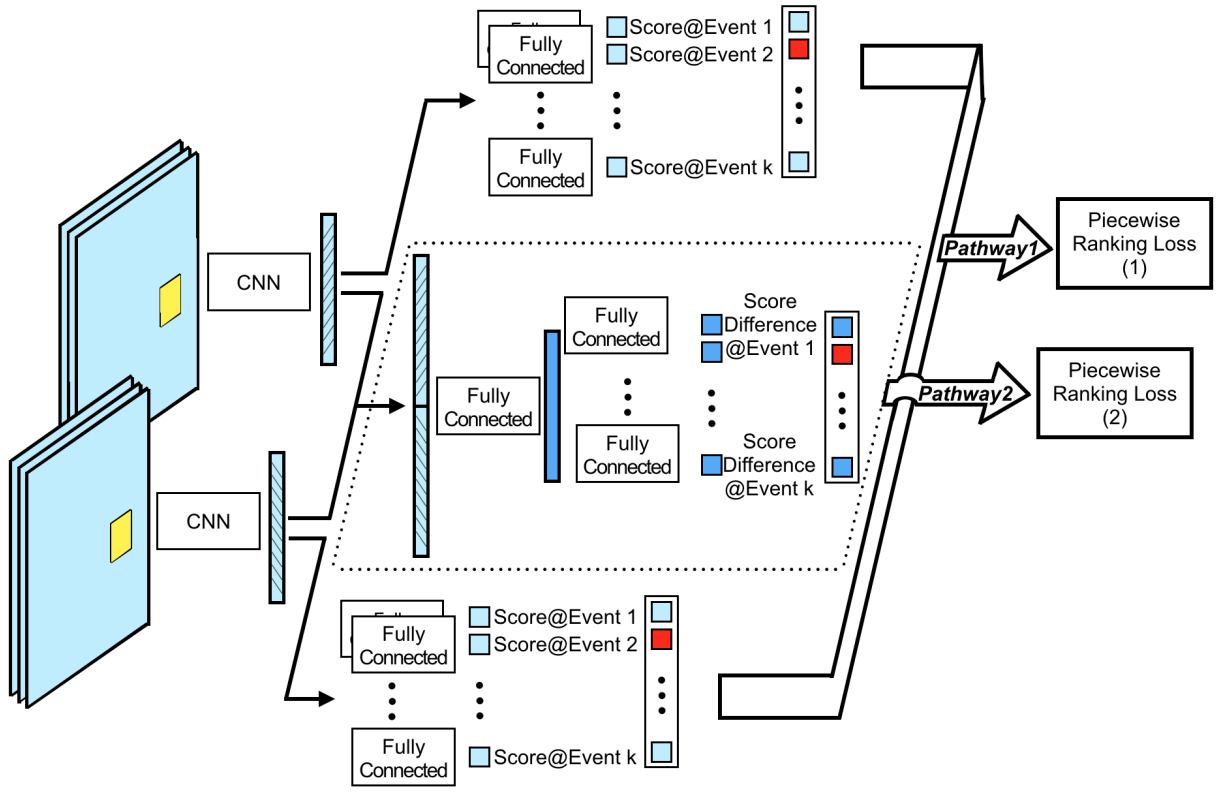


Figure 3.4: Architecture of the event curation siamese network (Curation-Siamese) during training. The "CNN" parts are the standard siamese network, the middle pathway that predicts score differences directly is novel in this application.

difference between an input image pair from an album given the ground-truth event type.

In Chapter 2, the siamese network predicts the absolute image score for each input image first, and then calculates the importance score difference, and a Piecewise Ranking Loss (PRL) is used as objective (shown in Figure 3.4 as Piecewise Ranking Loss(1)). This pathway is preserved in our architecture, and is denoted as **Pathway1**.

Unlike the architecture in Figure 2.6, we add another pathway to directly predict the score difference between the image pair (as shown in the dotted box in the middle in Figure 3.4). We denote this extra pathway as **Pathway2**. This pathway concatenates the image features extracted from both input images (*fc7* layer features for AlexNet, or the 500-unit fully connected layer features after the *pool5* layer when using ResNet), and adds a 300-unit fully connected layer

on top of the concatenated features, followed by a ReLU nonlinearity and dropout layer with 0.5 dropout rate. Then, the score difference between the image pair is directly predicted. The piecewise ranking loss is also used for this pathway, denoted Piecewise Ranking Loss (2).

In Pathway1, the siamese networks only see the two images separately, and predict the absolute importance score independently. However, Pathway2 adds a single network that sees both of the images, and directly predicts the score difference.

During the test stage, only one test image is fed into the trained network, with one importance score as the prediction from Pathway1. Though not used in the test stage, Pathway2 helps with the training of the network shared between both pathways, and effectively improves the performance of the network.

For each training image pair, the "ground-truth" event is sampled from the label distribution, and used to gate the output and gradient of the network. We denote this network as **Curation-Siamese**.

3.4.2 Event recognition networks

One of the properties of an "event album" that makes it distinct from a simple collection of images is that it is a sequence, and this provides us with the temporal relationship between the images. LSTMs have been successfully applied to sequential tasks [117, 50, 44, 96, 30], and their ability to remember long-range temporal context is suitable for our task. Therefore, we use the LSTM network to capture the sequential information, in addition to a classical CNN that captures the visual features of a single image.

We start with a CNN pre-trained on ImageNet [58, 65], and optionally fine-tune it on ML-CUFED to recognize the event type from a single image. We call this network **Recog-CNN**.

The architecture of the **Recog-CNN** is shown in Figure 3.5. The album's images are first fed into the trained CNN for single images. For AlexNet, *fc7* features are extracted, and for ResNet, *pool5* features are extracted. The dimensionality of the features is then reduced to 512

with PCA, and the sequence of compressed features is fed into the LSTM network. The LSTM network we use is the same as the one described in [30]. The dimensionality of the hidden units is 512. The hidden-unit features for all the time frames are then averaged by the mean pooling layer over time. The mean hidden features are used as the features of the whole album, and are fed into the prediction layer for the final event type prediction. The target for both the CNN and LSTM network is a one-hot encoding, but we treat each training example as from one of the possible event types according to the distribution of ground-truth event labels.

AdaDelta is used to train the network. There are 1404 training albums in ML-CUFED. To overcome the overfitting problem, we subsample 20 sub-albums from one album. The sub-albums contain no less than 75% images of the original album.

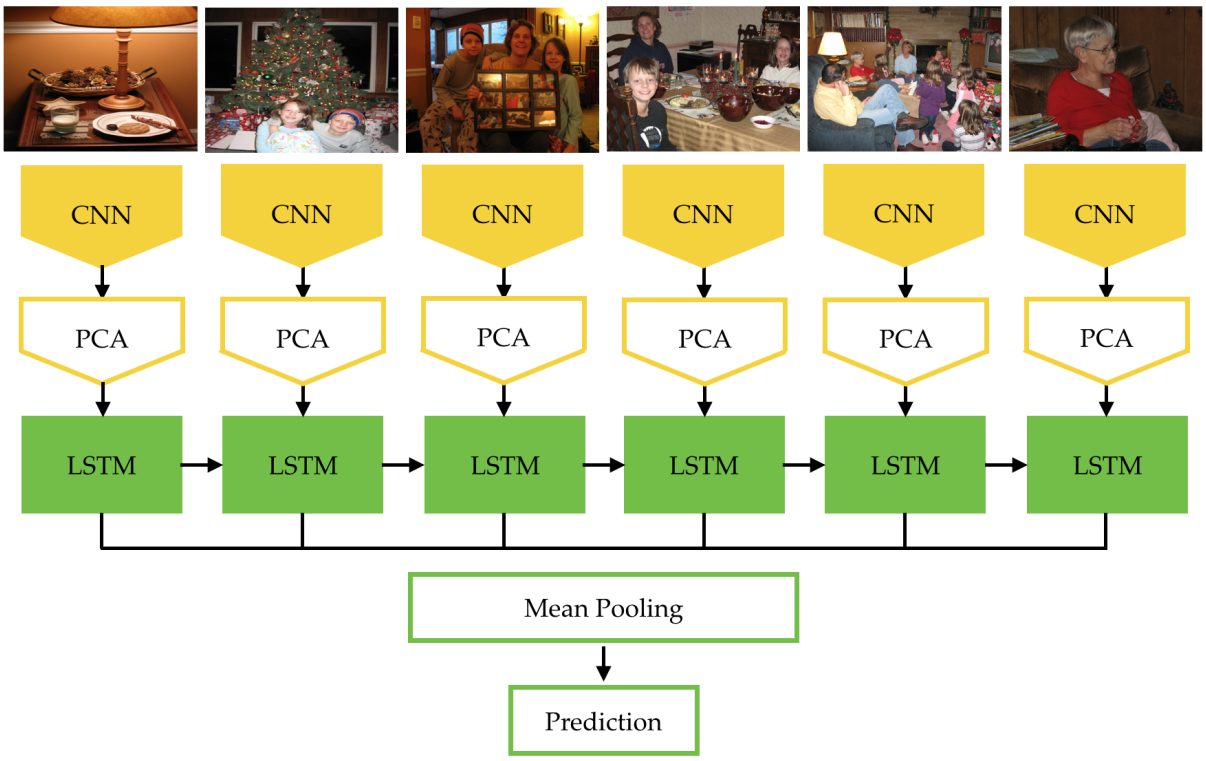


Figure 3.5: Architecture of the LSTM network for album-wise event recognition (Recog-LSTM)

3.4.3 The iterative curation-recognition procedure

For an "event album", more important images give us more information about the event type. For example, although a candle blowing image may only appear in an album once, it is a critical clue for revealing the event type of the album. However, as shown in Chapter 2, the importance of an image is event-type dependent. Therefore, we propose an iterative update procedure to demonstrate that the image importance score and event recognition of an album can be used to improve each other's performance.

We denote an N-image album as $\mathbf{A} = \{I_1, ..., I_N\}$. We assume C different event types. The input to the algorithm is the output of the above three networks: 1) Recog-CNN produces an N-by-C matrix Q, where each row is a probability distribution over event-types, given the image; 2) Recog-LSTM produces a 1-by-C row vector of probabilities of event types, $\hat{p}$, given the image sequence; 3) Curation-Siamese produces a N-by-C matrix W, where each row is the importance score of an image, given the event-type. The output of the algorithm is the N-dimensional column vector $v = [v_1, ..., v_N]^T$, which is the prediction of the importance score for all images in album $\mathbf{A}$, and the C-dimensional row vector $p = [p_1, p_2, ..., p_C]$, the distribution over the possible event types.

The iterative curation-recognition procedure is as follows:

1. **Re-weight Recog-CNN event prediction by image importance.**

$$p'(k+1) \propto (v^T(k))^{\alpha} \cdot Q \tag{3.1}$$

where $v(k)$ is the k-th step prediction for all images' importance scores in album $\mathbf{A}$ (initialized to a uniform distribution) and α is a parameter that controls the strength of the importance score for the update. p' denotes the updated album event type prediction, normalized to a distribution. Thus, p' is a distribution over event types that is the average of each image's event distribution weighted by the image's predicted importance score.

2. **Combine event type predictions** with $\hat{p}$.

$$p(k+1) = \frac{1}{2}(p'(k+1) + \hat{p}) \tag{3.2}$$

where $\hat{p}$ is the probability distribution of event types predicted by Recog-LSTM. Thus, $\hat{p}$ serves as an"anchor" for the prediction.

3. **Update image importance score with the updated event type distribution.**

$$v(k+1) \propto \left\{ W \circ \mathrm{I}\left\{ p_c \geq m \cdot \max_{c'}(p_{c'}) \right\}_{(1,c)} \right\} \cdot p(k+1)^T \tag{3.3}$$

where W is the importance scores of all the images given the event type from Curation-Siamese, $\circ$ denotes element-wise multiplication of each row, and I is an indicator that returns 1 if its argument is true and 0 otherwise. Hence, $\mathbf{I}$ forms a binary mask that zeros out the importance scores for columns of W that correspond to low-probability events, computed as a fraction m (a parameter) of the maximum probability event. Thus, the updated image importance is the average of the importance score given different events, weighted by the event type probability. Elements of $v(k+1)$ are normalized to range from 0 to 1.

By iterating Equations 4.4-3.3, we obtain the album-wise event prediction p and image importance score prediction v. Note that this procedure is not guaranteed to converge, hence we set a maximum number of iterations, and if this maximum number is reached before convergence, the predictions for p and v are obtained by averaging over last three steps.

3.5 Experiments

In this section, we evaluate our approach for both event recognition and image importance prediction on ML-CUFED, and we compare our event recognition result with Bossard *et al.*[11]

on another album-wise event recognition dataset they collected called PEC.

3.5.1 Baselines

Our joint recognition-curation method produces two outputs: an album event type prediction, and an image importance prediction. For event recognition, we compare our result with the baseline from Recog-CNN. In addition, the intermediate result of our algorithm can also be compared with the final result to validate the necessity of each part of our system. Therefore, we compare our method with the following methods:

- **CNN-recognition**: Use Recog-CNN to predict the event type for each image, and average the results.
- **CNN-LSTM**: The prediction by Recog-LSTM. Note this uses Recog-CNN's feature representation as input.
- **CNN-Iterative**: Use the proposed method as described in Section 3.4.3, but without step 2. Therefore, Recog-LSTM result is not involved.
- **CNN-LSTM-Iterative**: Our full proposed method as described in Section 3.4.3.

To evaluate our image importance prediction, we compare with several baselines:

- **CNN-Noevent**: Train a Siamese Network to predict the importance score difference of an input image pair without any event-type information. All albums are considered to be part of the same "uber" event type.
- **CNN-Noevent(test)**: Use Curation-Siamese that is trained using the ground-truth event type information to gate the output error and back-propagation signal, while during testing, average the predicted importance score for all possible event types.
- **CNN-LSTM-Iterative**: As above.

3.5.2 Experimental details

Dataset

For ML-CUFED, we split the albums into training and test in a ratio of 4:1. The test set has 368 albums. To decide the hyper-parameters (α, m) in our iterative model, a validation set with 111 albums is extracted from the training set. For the PEC Event Recognition Dataset [11], we use directly the test set consisting of 10 albums for each event type as described in [11], so that we can directly compare their results with ours.

Parameter setting

For both the Recog-CNN and the Curation-Siamese, we use two architectures: 8-layer AlexNet [65] and 101-layer ResNet [48]. Both networks are pre-trained on ImageNet, and we fine-tune AlexNet on ML-CUFED. We use a similar training scheme to [58], but with a lower learning rate of 0.001. For Recog-LSTM, we use high-level features from the Recog-CNN as input. For fine-tuned AlexNet, we use *fc7* layer features, while for ResNet, we use the *pool5* layer features. We reduce the feature dimension to 512 with PCA. For the Recog-LSTM, the dimensionality of the LSTM is 512, and we use AdaDelta as the optimization method [133, 10, 7]. For Curation-Siamese, we follow the settings in Chapter 2 and choose the two margins as $m_s = 0.1$ and $m_d = 0.3$. We set the number of iterations of our joint recognition curation algorithm to 10.

Evaluation

For event recognition on ML-CUFED, we use two metrics to evaluate the models: average accuracy and F_1 Score. F_1 Score is the harmonic mean of precision and recall, and can account for multi-label ground-truth. Both accuracy and F_1 are calculated with top-1 prediction. For event recognition on PEC, only average accuracy is used. For image importance prediction, we follow Chapter 2 using MAP@(t%) and Precision@(t%). Precision is the ratio between the number of

retrieved relevant images and the number of retrieved images. MAP is the averaged area under the precision-recall curve.

3.5.3 Results on the ML-CUFED Dataset

Performance over iterations

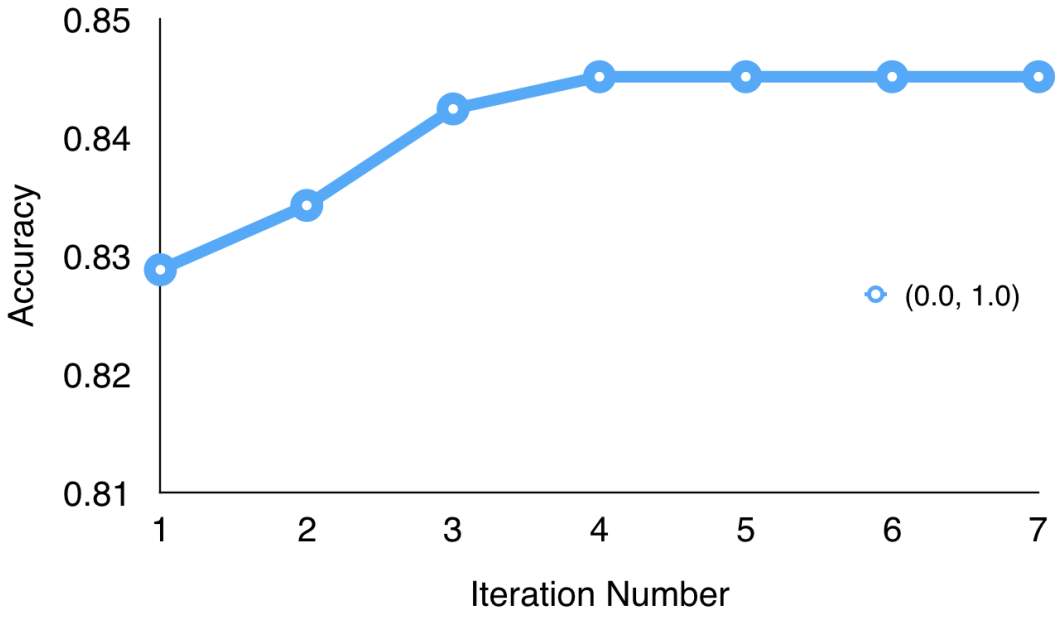
Our curation-recognition procedure iteratively updates the album-wise event type prediction and the image-wise importance score prediction. There are two hyper-parameters for the iterative procedure: $\theta = (m, \alpha)$. Here, m is a threshold (a fraction of the maximum probability) used to eliminate event types with low probability by setting their probability to 0. These are then ignored by the image importance prediction procedure; α is the emphasis we put on the image importance score for event type prediction. When $m = 0$, all event types are considered for image importance calculation; when $m = 1$, only one event type with highest probability is considered.

These hyper-parameters are determined using a 111-album validation set and running a grid search on choices of $\theta = (m, \alpha)$. Using ResNet features, the hyper-parameters are as follows: for ML-CUFED, $\theta = (0.0, 1.0)$. For PEC, $\theta = (1.0, 1.4)$. Using fine-tuned AlexNet features: $\theta = (0.3, 1.9)$ for ML-CUFED, and $\theta = (0.6, 1.1)$ for PEC. In Figure 3.6, we show the system performance with respect to the iteration number on ML-CUFED using ResNet features.

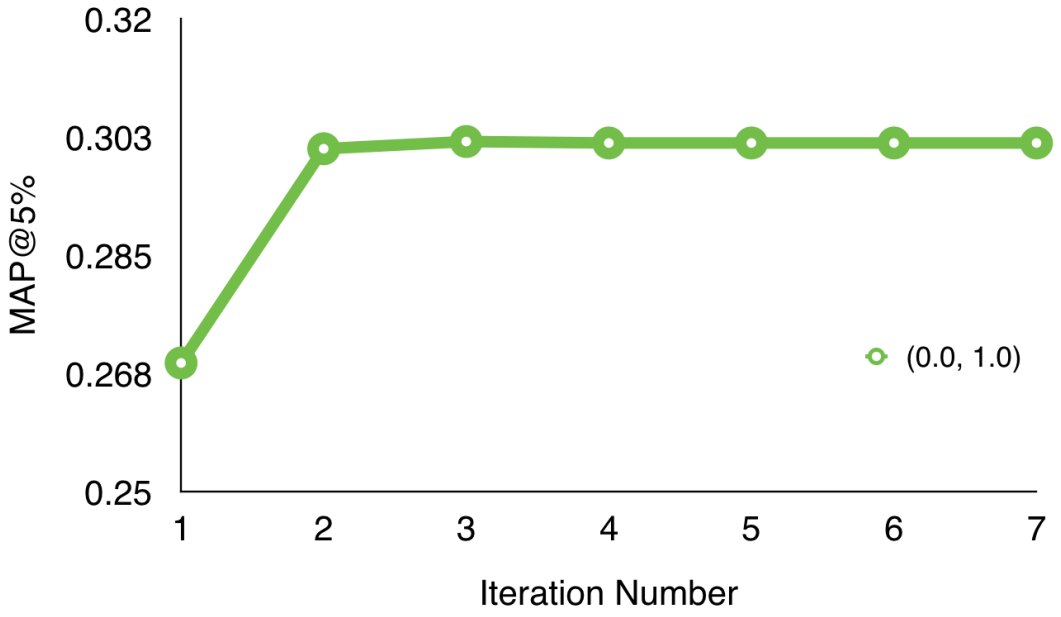
As shown in Figure 3.6, both album-wise event recognition performance and image importance score prediction performance improve over iterations, and converges after a small number of iterations.

Event-specific image importance

In Table 3.2, we show the performance gained by using the two pathway model in Figure 3.4 versus training with only Pathway1. We show the comparison for both AlexNet and ResNet. There is a steady improvement with the use of Pathway2. The performance gain is about 0.5%



(a) Album-wise event recognition accuracy v.s. iteration number for hyper-parameters $(m, \alpha) = (0.0, 1.0)$ of the iterative curation-recognition procedure.



(b) Image importance prediction accuracy (MAP@5%) v.s. iteration number for hyper-parameters $(m, \alpha) = (0.0, 1.0)$.

Figure 3.6: Performance of our joint system with respect to iteration number on ML-CUFED using ResNet features.

Table 3.2: Comparison of the Curation-Siamese with only Pathway1, as used in Chapter 2, and the two pathway model used in this paper. Note that all the results shown here are obtained assuming ground-truth event types are known during the test stage.

	MAP@t%				**P@t%**			
t%	5	10	15	20	5	10	15	20
AlexNet-Pathway1	0.298	0.362	0.417	0.469	0.199	0.300	0.354	0.407
AlexNet-Pathway1&2	0.305	0.368	0.421	0.472	0.211	0.307	0.362	0.412
ResNet-Pathway1	0.305	0.376	0.427	0.477	0.202	0.309	0.368	0.423
ResNet-Pathway1&2	**0.310**	**0.382**	**0.432**	**0.481**	**0.206**	**0.311**	**0.372**	**0.428**

on both MAP and Precision, which is similar to the gain from the use of 2-stage learning in Chapter 2.

For the image importance score prediction task, we compare our methods to several baselines in Table 3.3 using AlexNet features and Table 3.4 using ResNet features. For an upper-bound of our method, we also show the result for CNN-GTEvent, where we assume the ground truth event type is known, and predict the importance score based on that. CNN-GTEvent serves as the best result we can get when the event recognition stage is perfect.

As shown in Table 3.4 and Table 3.3, CNN-Noevent performs better than CNN-Noevent (test). This suggests the divergence of the importance prediction for different event types.

CNN-Noevent performs a little better than CNN-Noevent (test). CNN-LSTM-Iterative greatly outperforms CNN-Noevent, with a steady 3% MAP increment. There is also a steady 3% increment for P at t $<$ 20%. CNN-LSTM-Iterative closely approaches the upper bound (CNN-GTEvent), with a more notable performance gap between CNN-LSTM-Iterative and CNN-Noevent. CNN-LSTM-Iterative greatly outperforms the other two models. With AlexNet, 70% of the gap that exists between CNN-Noevent (test) and the results using the ground truth event type (CNN-GTEvent) is crossed by CNN-LSTM-Iterative, while with ResNet, the 79% of the gap is crossed. With AlexNet, 56% of the gap is crossed, while with ResNet, the 62% of it is crossed. This is because the ResNet features achieve better event type recognition performance, and better event type recognition in turn helps improve the image importance score prediction result.

Table 3.3: Comparison of event-specific image importance predictions using different methods with AlexNet. We also show the score using a random ranking as a lower bound, and a CNN-GTEvent result which uses ground-truth event type information when testing as an upper-bound.

	MAP@t%						**P@t%**					
t%	5	10	15	20	25	30	5	10	15	20	25	30
Random	0.113	0.161	0.211	0.256	0.303	0.350	0.044	0.090	0.142	0.193	0.243	0.298
CNN-Noevent(test)	0.251	0.303	0.358	0.414	0.462	0.508	0.142	0.211	0.284	0.335	0.384	0.436
CNN-Noevent	0.258	0.316	0.369	0.425	0.475	0.519	0.168	0.245	0.307	0.373	0.422	0.468
CNN-LSTM-Iterative	**0.278**	**0.347**	**0.400**	**0.453**	**0.502**	**0.547**	**0.191**	**0.280**	**0.340**	**0.394**	**0.450**	**0.491**
CNN-GTEvent	0.305	0.372	0.424	0.476	0.522	0.565	0.218	0.304	0.361	0.417	0.461	0.504

Table 3.4: Comparison of event-specific image importance predictions with different methods using ResNet features. We also show Random ranking score as a lower bound, and a CNN-GTEvent result which uses ground-truth event type information when testing as an upper-bound.

	MAP@t%						**P@t%**					
t%	5	10	15	20	25	30	5	10	15	20	25	30
Random	0.113	0.161	0.211	0.256	0.303	0.350	0.044	0.090	0.142	0.193	0.243	0.298
CNN-Noevent(test)	0.272	0.330	0.380	0.434	0.483	0.530	0.167	0.256	0.327	0.379	0.432	0.476
CNN-Noevent	0.280	0.352	0.403	0.455	0.504	0.552	0.178	0.281	0.347	0.404	0.454	0.497
CNN-LSTM-Iterative	**0.302**	**0.371**	**0.419**	**0.470**	**0.520**	**0.568**	**0.205**	**0.300**	**0.360**	**0.413**	**0.459**	**0.507**
CNN-GTEvent	0.309	0.383	0.432	0.482	0.529	0.573	0.205	0.311	0.373	0.428	0.472	0.512

Event recognition

Table 3.5 shows the results of different methods for event recognition. For an album with multiple labels, we deem it correctly predicted if the top-1 prediction is among the ground-truth event labels. As shown, ResNet features perform much better than AlexNet features. For both AlexNet and ResNet features, there is a performance gain over all three baselines. We can also observe that both iterative curation-recognition and LSTM method help to improve the final result. This suggests that both these types of information in an event album are helpful in deciding the event type of this album: image importance information, and album sequential information.

We compare our results with another CNN-based model in [124]. Wu *et al.*[124] use a fine-tuned AlexNet to extract image features, and average the image features for album-wise prediction of event type. Here, we reimplement their approach for ML-CUFED, using ResNet. Our model substantially outperforms theirs.

Table 3.5: Comparison of event-recognition models on ML-CUFED and PEC. Note that for the PEC result, our model is trained on ML-CUFED, while Wu *et al*.[124]'s model and SHMM are trained on the PEC training set.

Dataset	**ML-CUFED**				**PEC**	
	Avg. Acc.		**F1-Score**		**Avg. Acc.**	
Method	AlexNet	ResNet	AlexNet	ResNet	AlexNet	ResNet
CNN-recognition	75%	82.9%	0.698	0.772	80.9%	84.5%
CNN-LSTM	76.6%	81.5%	0.713	0.759	82.7%	85.5%
CNN-Iterative	78%	83.7%	0.729	0.781	81.8%	86.4%
CNN-LSTM-Iterative	**79.3%**	**84.5%**	**0.737**	**0.786**	**84.5%**	**87.9%**
Wu *et al*.[124]		83.4%		0.773	*84.5%	*89.1%
SHMM [11]	-		-		*76.3%	

Qualitative Results

In this section, we show more examples of the qualitative results of our algorithm.

In Figure 3.7 and 3.8, several albums in ML-CUFED are shown. Figure 3.7 is an example of test results from AlexNet, and Figure 3.8 is using ResNet. For the examples shown here, the album-wise event type prediction is incorrect with the CNN recognition method, which simply averages the results from the classification of single images, but with the proposed CNN-LSTM-Iterative algorithm, the event type prediction is corrected. Also, we can see the ground-truth and predicted importance ranking of the images in each album. We also show the baseline image importance prediction results in the middle row. This baseline is achieved without event type prediction by just averaging the importance prediction across all event types. Note that there are equal ranks for multiple images in the ground-truth importance ranking. This is because the importance scores from 5 votes of Amazon Mechanical Turk(AMT) workers have a lot of ties. Therefore, the ranks shown here are the median ranking of all the images with the same score, and thus the ranks for the images with same ground-truth score are the same.

Only a fraction of images in the albums are shown here due to limited space. We deliberately choose both images with high ground-truth importance and low ground-truth importance for each album to show the overall quality of the albums.

For example, in the third example of a Cruise Trip album in Figure 3.7, there are many images of the iceberg and the sea similar to the last image shown here. If only CNN-recognition is used, and the prediction is produced by averaging the prediction of every image in the album, the album is recognized as a Beach Trip. However, after we assign different importance scores to images, as shown in the ranking of images, this album is correctly recognized as a Cruise Trip. Also, by comparing the image importance ranking between the second and third rows, corresponding to baseline importance prediction and CNN-LSTM-iterative importance prediction method, we can see that predicting the event type as a cruise increases the importance score of the first three images, which are more relevant to the event type, while decreasing the importance score of the photo of the cat and the selfie-like photo. The image rankings of the proposed method (in the third row) is obviously closer to the ground-truth ranking than the baseline method (in the second row). This demonstrates the advantages of the joint recognition-curation algorithm. In other words, our full method is able to rank important images (which are more indicative of event types) at the top.

In Figure 3.9 and 3.10, more examples with correct event type prediction in ML-CUFED are shown. Figure 3.9 is from the network using AlexNet, and 3.10 is from the network using ResNet. We can also see how the images are ranked with predicted importance by the baseline algorithm and the proposed joint event recognition-curation algorithm. We can see many examples of better importance prediction results with the joint algorithm using the event type prediction, such as the first and second image in the first Birthday album in Figure 3.9; and the first three images in the third Wedding album in Figure 3.10.

In Figure 3.11 and Figure 3.12, we show some examples of incorrect event type prediction in ML-CUFED. Figure 3.11 is from AlexNet, and Figure 3.12 is from ResNet. In Figure 3.11, the ground-truth event type of the three example albums are book signing event (business activity), ball (group activity), and graduation party (graduation) respectively. In Figure 3.12, the ground-truth event type of the three example albums are Korean traditional wedding, Christmas family

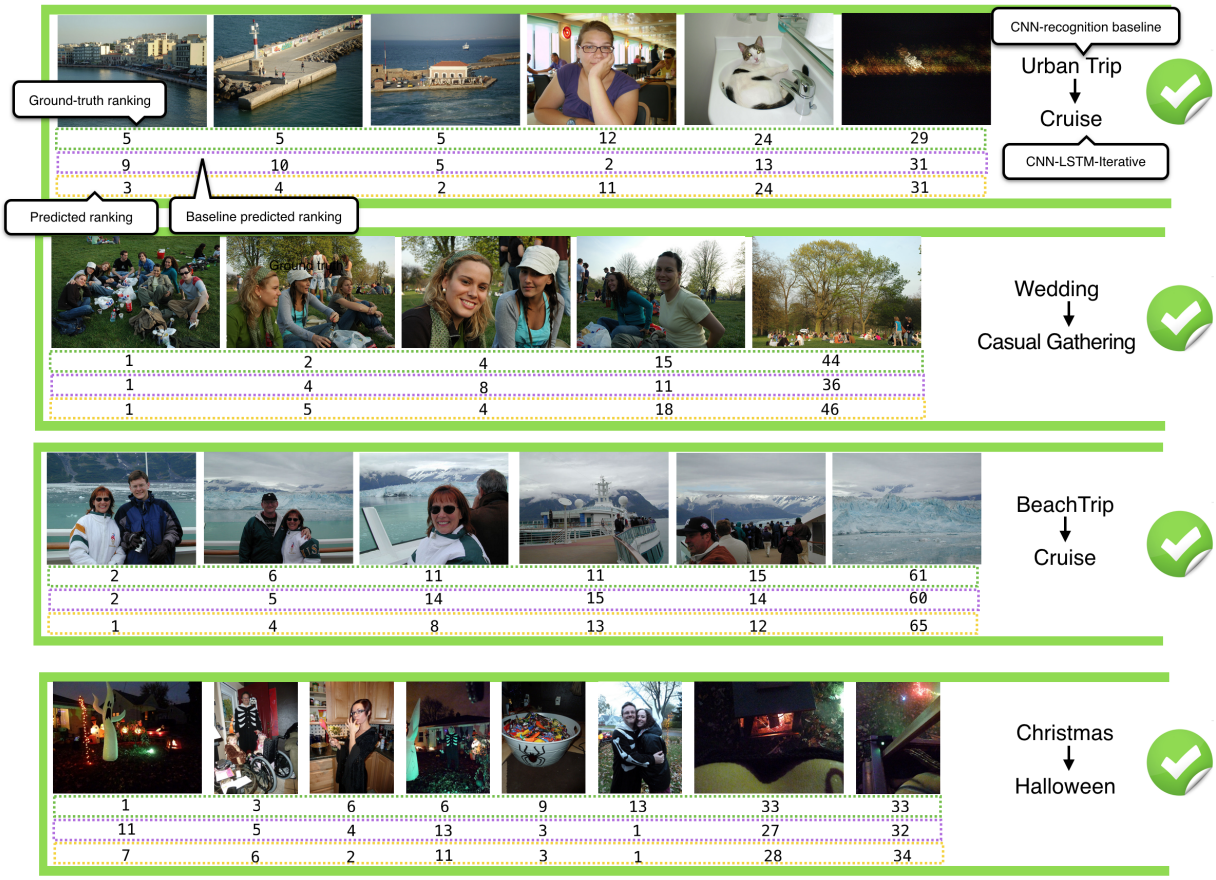


Figure 3.7: Examples of recognition-curation result on ML-CUFED using AlexNet. These examples were incorrectly categorized by CNN-recognition, but correctly categorized by CNN-LSTM-Iterative. We show the ground-truth ranking, the baseline predicted ranking, and the predicted importance ranking.

Figure 3.8: Examples of recognition-curation result on ML-CUFED using ResNet. These examples were incorrectly categorized by the CNN-recognition method, but correctly categorized by the CNN-LSTM-Iterative.

Figure 3.9: More examples of the recognition-curation results from ML-CUFED using AlexNet. The event types of albums are correctly recognized, as shown to the right of each album.

Figure 3.10: More examples of recognition-curation results from ML-CUFED using ResNet. The event types of albums are correctly recognized, as shown in the right of each album.

party, and casual friends gathering respectively.

Figure 3.11: Examples of recognition-curation result from ML-CUFED using AlexNet, whose event types are predicted incorrectly. The predicted event type and the ground-truth event type are shown in the right of each album. The ground-truth event type is shown in parenthesis.

3.5.4 Results on the PEC Dataset

To show the generalizability of our algorithm, we compare our result with [11] and [124] on PEC. The PEC dataset is an 807-album event dataset with 14 social event classes. There is no ground-truth importance score in PEC, thus we cannot train our algorithm on it. Therefore, we use the model we trained on ML-CUFED and test the model on the PEC test set containing 10 albums each class.

PEC has several event types that are not contained in ML-CUFED, such as Saint Patrick's Day, Easter, and Skiing, and there are two event types that can map to single event type in ML-CUFED: Children's Birthday and Birthday can be mapped to single Birthday event in ML-CUFED. Therefore, we provide the mapping from PEC label to ML-CUFED label in Table 3.6 here. There are 9 event types in PEC after merging.Note that the mapping is not perfect, and the

Figure 3.12: Examples of recognition-curation result from ML-CUFED using ResNet, whose event types are predicted incorrectly.

noise in the mapping makes the performance of our method shown here a little poorer than it really is.

Table 3.6: Event type matching from PEC Dataset to ML-CUFED Dataset.

PEC	(Children's) Birthday	Christmas	Concert	Graduation	Exhibition
ML-CUFED	Birthday	Christmas	Show	Graduation	Museum
PEC	Halloween	Hiking, Road Trip	Wedding	Cruise	
ML-CUFED	Halloween	Nature Trip	Wedding	Cruise	

Due to the label changes, we recalculate the performance of Stopwatch HMM (SHMM) [11] on the test data based on the confusion matrix they provided in the paper. For merged labels, the corresponding rows in the confusion matrix are merged. For missing labels, there are many possible approaches, and we follow the most loose one which assumes the best possible predictions: Assume false positive on those labels will be correct predictions if those labels disappear.

The comparison of different methods is shown in Table 3.5. Similar to the result on

ML-CUFED, we observe consistent performance gain from both LSTM network and iterative updates. For the result of our reimplementation of [124], it is worth noticing that this model is trained on PEC, and it achieves current state-of-the-art result on PEC. Although our model is trained on ML-CUFED, it achieves very close performance with [124].

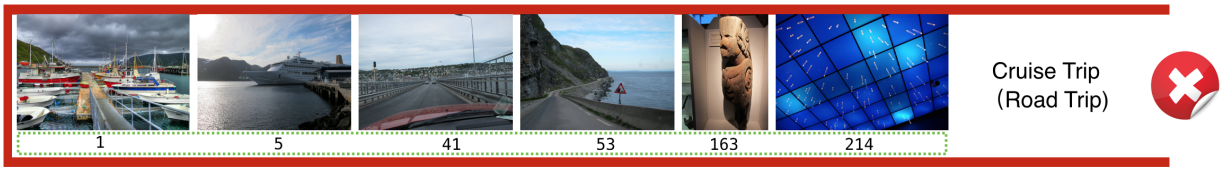
In Figure 3.13, We show some examples of event recognition result on PEC dataset of our CNN-LSTM-Iterative system using ResNet. As in Section 3.5.3, we show two examples which are incorrectly categorized by the CNN-recognition method, but correctly categorized by the CNN-LSTM-Iterative in Figure 3.13(a); four examples which are correctly recognized in Figure 3.13(b); one example which is wrongly recognized by CNN-LSTM-Iterative in Figure 3.13(c).

There is no ground-truth image importance score in PEC, therefore in Figure 3.13 we only show the image importance rank predicted by CNN-LSTM-Iterative. For each album, we only show a fraction of images in it, but we deliberately choose both images with high predicted importance and low predicted importance.

We also show the PEC results using AlexNet in Figure 3.14.

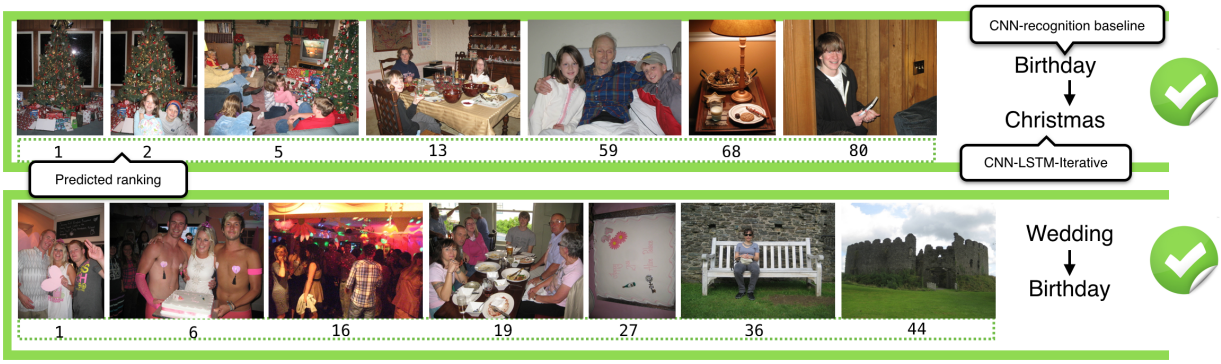
(a) Examples of two albums that are incorrectly categorized by CNN-recognition method, but correctly categorized by CNN-LSTM-Iterative.

(b) Examples of four albums that are correctly recognized.

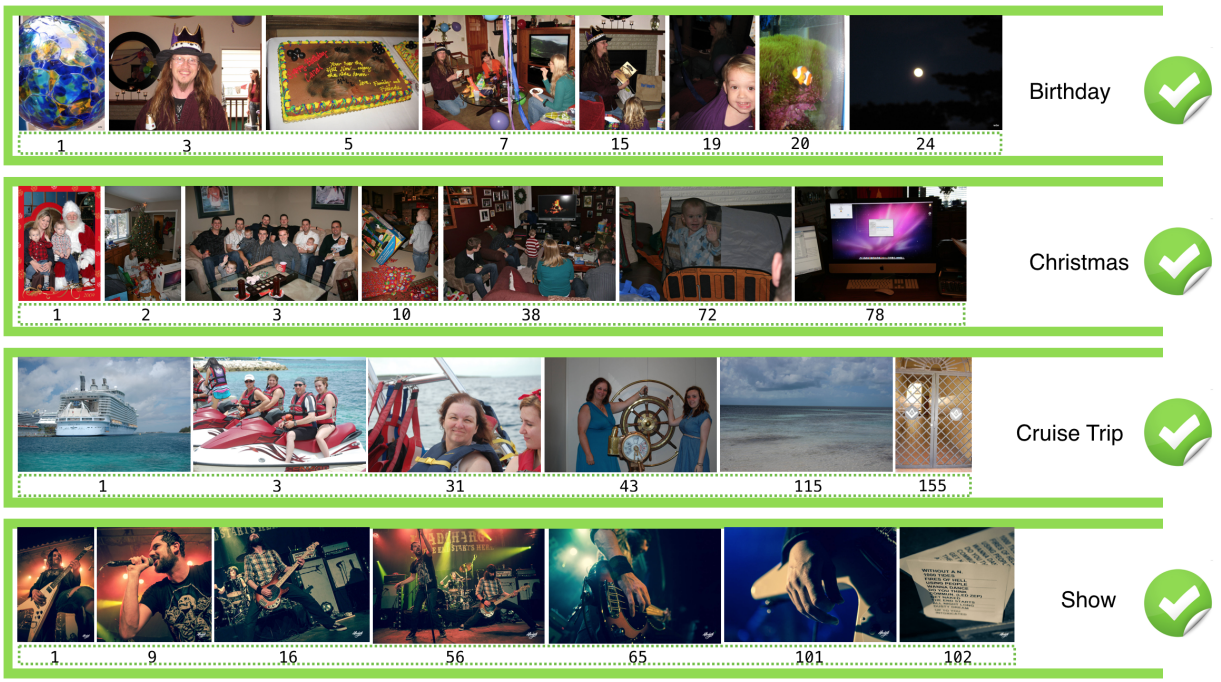


(c) An album whose event type is predicted incorrectly.

Figure 3.13: Examples of album recognition result on PEC dataset using ResNet. Rank of the predicted image importance is also shown for each image.



(a) Examples of two albums that are incorrectly categorized by CNN-recognition method, but correctly categorized by CNN-LSTM-Iterative.



(b) Examples of four albums that are correctly recognized.

(c) An album whose event type is predicted incorrectly.

Figure 3.14: Examples of album recognition result on PEC dataset using AlexNet.

3.6 Conclusion

In this work, we explore the problem of automatically recognizing and curating personal event albums. It is the first attempt to solve the following two tasks jointly: recognizing the event type of an album, and finding the important images in this album. Specifically, the result from a CNN for image-wise event recognition, an LSTM Network for album-wise event recognition, and a Siamese Network for image importance prediction are integrated by a unified, iterated updating algorithm. We show that the joint algorithm significantly improves both image importance prediction and event recognition.

3.7 Acknowledgements

Chapter 3, in full, is an edited reprint of the material as it appears in the following publication: Wang, Y., Zhe, L., Shen, X., Měch, R., Miller, G., Cottrell, G. W. (2017), "Recognizing and Curating Photo Albums via Event-Specific Image Importance", In *British Machine Vision Conference (BMVC)*, 2017. The dissertation author was a primary researcher and an author of the cited material.

This work was supported in part by NSF grants SMA-1041755, IIS-1219252 to G. W. Cottrell, and gift money from Adobe Research to Garrison. W. Cottrell.

Chapter 4

Skeleton Key: Image Captioning by Skeleton-Attribute Decomposition

In this chapter, we extend the problem of image understanding by extending the interpreting an image using beyond tags and classes. We aim to use sentence to describe an image.

Recently, there has been a lot of interest in automatically generating descriptions for an image. Most existing language-model based approaches for this task learn to generate an image description word by word in its original word order. However, for humans, it is more natural to locate the objects and their relationships first, and then elaborate on each object, describing notable attributes. We present a coarse-to-fine method that decomposes the original image description into a skeleton sentence and its attributes, and generates the skeleton sentence and attribute phrases separately. By this decomposition, our method can generate more accurate and novel descriptions than the previous state-of-the-art. Experimental results on the MS-COCO and a larger scale Stock3M datasets show that our algorithm yields consistent improvements across different evaluation metrics, especially on the SPICE metric, which has much higher correlation with human ratings than the conventional metrics. Furthermore, our algorithm can generate descriptions with varied length benefiting from the separate control of the skeleton and attributes. This enables image description generation that better accommodates user preferences.

4.1 Introduction

The task of automatically generating image descriptions, or image captioning, has drawn great attention in computer vision community. The problem is challenging in that the description generation process requires the understanding of high level image semantics beyond simple object or scene recognition, and the ability to generate a semantically and syntactically correct sentence to describe the important objects, their attributes and relationships.

The image captioning approaches generally fall into three categories. The first category tackles this problem based on retrieval: given a query image, the system searches for visually similar images in a database, finds and transfers the best descriptions from the nearest neighbor

captions for the description of the query image [27, 51, 67, 87]. The second category typically uses template-based methods to generate descriptions that follow predefined syntactic rules[35, 66, 74, 32, 128, 83]. Most recent work falls into the third category: language model-based methods [34, 117, 126, 30, 81, 62]. Inspired by the machine translation task [105, 5, 19], an image to be described is viewed as a "sentence" in a source language, and an Encoder-Decoder network is used to translate the input to the target sentence. Unlike machine translation, the source "sentence" is an image in the captioning task, therefore a natural encoder is a Convolutional Neural Network (CNN) instead of a Recurrent Neural Network (RNN).

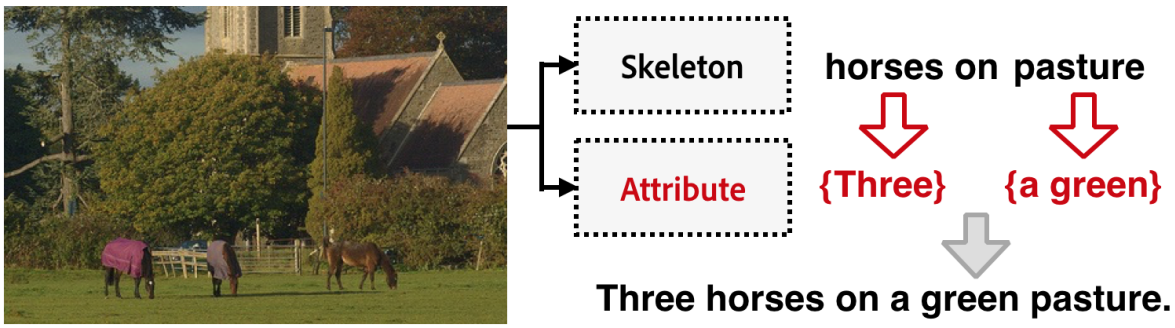


Figure 4.1: Illustration of the inference stage of our coarse-to-fine captioning algorithm with skeleton-attribute decomposition. First, the skeleton sentence is generated, describing the objects and relationships; Then, the objects are revisited and the attributes for each object are generated.

Starting from the basic form of a CNN encoder-RNN decoder, there have been many attempts to improve the system. Inspired by the success in machine translation, Long-short Term Memory (LSTM) is used as the decoder in [117, 30]. Xu *et al.*[126] add an attention mechanism to the system that learns to attend to parts of the image for word prediction. It is also found that feeding high level attributes instead of CNN features yields improvements [131, 123].

Despite the variation in approaches, most of the existing LSTM based methods suffer from two problems: 1) they tend to parrot back sentences from the training corpus, and lack variations in the generated captions [26]; (2) due to the inherent structures that predict captions word by word, LSTM predicts attributes before predicting the subject they are referring to. Mixtures of attributes, subjects, and relations in a complete sentence create large variations across training

samples, which can affect training effectiveness.

In order to overcome these problems, in this paper, we propose a coarse-to-fine algorithm to generate the image description in a two stage manner: First, the skeleton sentence of the image description is generated, containing the main objects involved in the image, and their relationships. Then, the objects are revisited in a second stage using attention, and the attributes for each object are generated if they are worth mentioning. The flow is illustrated in Figure 4.1. By dealing with the skeleton and attributes separately, the system is able to generate more accurate image captions.

Our work is also inspired by a series of Cognitive Neuroscience studies. During visual processing such as object recognition, two types of mechanisms play important roles: first, a fast subcortical pathway that projects to the frontal lobe does a coarse analysis of the image, categorizing the objects [12, 33, 39], and this provides top-down feedback to a slower, cortical pathway in the ventral temporal lobe [110, 13] that proceeds from low level to high level regions to recognize an object. The exact way that the top-down mechanism is involved is not fully understood, but Bar [6] proposed a hypothesis that low spatial frequency features trigger the quick "initial guesses" of the objects, and then the "initial guesses" are back-projected to low level visual cortex to integrate with the bottom-up process.

Analogous to this object recognition procedure, our image captioning process also comprises two stages: 1) a quick global prediction of the main objects and their relationship in the image, and 2) an object-wise attribute description. The objects predicted by the first stage are fed back to help the bottom-up attribute generation process. Meanwhile, this idea is also supported by object-based attention theory. Object based attention proposes that the perceptual analysis of the visual input first segments the visual field into separate objects, and then, in a focal attention stage, analyzes a particular object in more detail [86, 31].

The main contributions of this paper are as follows: 1) We are the first to divide the image caption task such that the skeleton and attributes are predicted separately; 2) our model

improves captioning performance consistently against a very strong baseline that outperforms the published state-of-the-art results. The improvement on the recently proposed SPICE evaluation metric is significant. It is worth mentioning that SPICE [1] has much higher correlation with human judgment than all the conventional metrics; 3) we also propose a mechanism to generate image descriptions with variable length using a single model. The coarse-to-fine system naturally benefits from this mechanism, with the ability to vary the skeleton/attribute part of the captions separately. This enables us to adapt image description generation according to user preferences, with descriptions containing a varied amount of object/attribute information.

4.2 Related Work

Existing image captioning methods

Retrieval-based methods search for visually similar images to the input image, and find the best caption from the retrieved image captions. For example, Devlin *et al*.in [27] propose a K-nearest neighbor approach that finds the caption that best represents the set of candidate captions gathered from neighbor images. This method suffers from an obvious problem that the generated captions are always from an existing caption set, and thus it is unable to generate novel captions.

Template-based methods generate image captions from pre-defined templates, and fills the template with detected objects, scenes and attributes. Farhadi *et al*.[35] use single ⟨object, action, scene⟩ triple to represent a caption, and learns the mapping from images and sentences separately to the triplet meaning space. Kulkarni *et al*.[66] detect objects and attributes in an image as well as their prepositional relationship, and use a CRF to predict the best structure containing those objects, modifiers and relationships. In [68], Lebret *et al*.detect phrases in an image, and generate description with a constrained model with the detected phrases. These approaches heavily rely on the templates, and so generate rigid captions.

Language model-based methods typically learn the common embedding space of images and captions, and generate novel captions without many rigid syntactical constraints. Kiros and Zemel [61] propose multimodal log-bilinear models conditioned on image features. Mao *et al*.[81] propose a Multimodal Recurrent Neural Network (MRNN) that uses an RNN to learn the text embedding, and a CNN to learn the image representation. Vinyals *et al*.[117] use LSTM as the decoder to generate sentences, and provide the image feature as input to the LSTM directly. Xu *et al*.[126] further introduce an attention-based model that can learn where to look while generating corresponding words. You *et al*.[131] use pre-generated semantic concept proposals to guide the caption generation, and learn to selectively attend to those concepts at different time-step. Similarly, Wu *et al*.[123] also show that high level semantic features can improve the caption generation performance.

Our work is a language-model based method. Unlike approaches to LSTM-based methods that try to feed a better image representation to the language model, we focus on the caption itself, and show how breaking the original word order in a natural way can yield better performance.

Analyzing the sentences for image captioning

Parsing of a sentence is the process of analyzing the sentence according to a set of grammar rules, and generates a rooted parse tree that represents the syntactic structure of the sentence [63]. There are some language-model based works that parse the captions with language analysis approach such as parsing for better sentence encoding. Socher *et al*.[102] propose the Dependency Tree-RNN, which uses dependency trees to embed sentences into a vector space, and then perform caption retrieval with the embedded vector. The model is unable to generate novel sentences.

Another work that is probably the closest to our paper is by Tan and Chan [109]. They employ a hierarchical LSTM model that views captions as a combination of noun phrases and other words, and try to predict the noun phrases (together with other words) directly with an

LSTM. The noun phrases are encoded into a vector representation with a separate LSTM. In the inference stage, K image relevant phrases are generated first with the lower level LSTM. Then, the upper level LSTM generates the sentence that contains both the "noun phrase" token and other words. When a noun phrase is generated, suitable phrases from the phrase pool are selected, and then used as the input to the next time-step. This work is relevant to ours in that it also tries to break the original word order of the caption. However, it directly replaces the phrases with a single word "phrase token" in the upper level LSTM without distinguishing those tokens, although the phrases can be very different. Also, the phrases in an image are generated ahead of the sentence generation, without knowing the sentence structure or the location to attend to.

Evaluation metrics

Evaluation of image caption generation is as challenging as the task itself. Bleu [89], CIDEr [116], METEOR [25], and ROUGE [75] are common metrics used for evaluating most image captioning benchmarks such as MS-COCO and Flickr30K. However, these metrics are very sensitive to n-gram overlap, which may not necessarily be a good way to measure the quality of an image description. Recently, Anderson *et al*.[2] introduced a new evaluation metric called SPICE that overcomes this problem. SPICE uses a graph-based semantic representation to encode the objects, attributes and relationships in the image. They show that SPICE has a much higher correlation with human judgement than the conventional evaluation metrics.

In our work, we evaluate our results using both conventional metrics and the new SPICE metric, but would like to advocate SPICE due to its closer correlation with human judgement. We also show how unimportant words like "a" impact scores on conventional metrics.

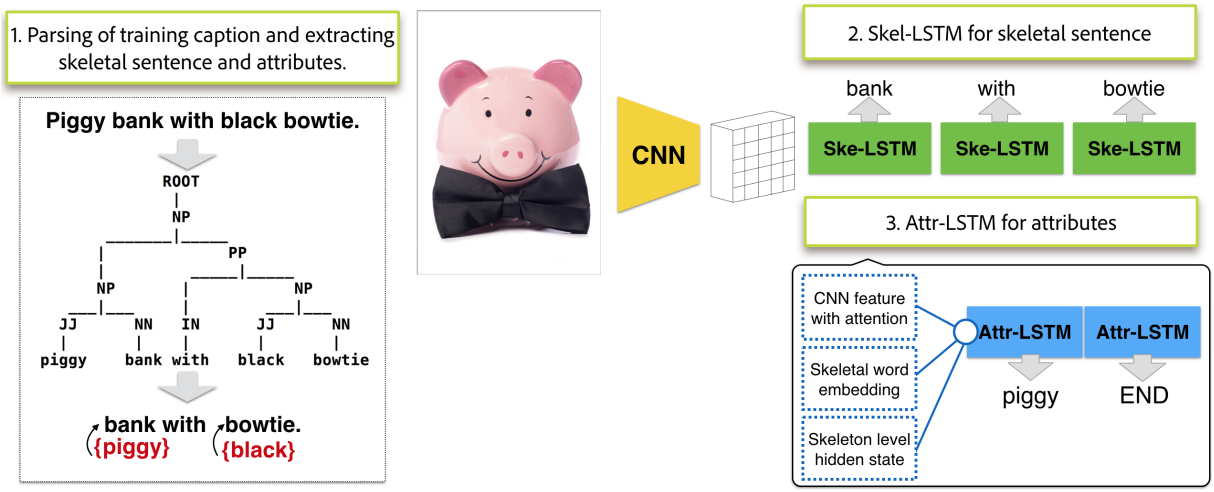


Figure 4.2: The overall framework of the proposed algorithm. In training stage, the training image caption is decomposed into the skeleton sentence and corresponding attributes. A Skel-LSTM is trained to generate the skeleton, and then an Attr-LSTM generates attributes for each skeletal word.

4.3 The Proposed Model

The overall framework of our model is shown in Figure 4.2. In the training stage, the ground-truth captions are decomposed into the skeleton sentences and attributes for the training of two separate networks. In the test stage, the skeleton sentence is generated for a given image, and then attributes conditioned on the skeleton sentence are generated. They are then merged to form the final generated caption.

4.3.1 Skeleton-Attribute decomposition for captions

To extract the skeleton sentence and attributes from a training image caption, we use the Stanford constituency parser [63, 80]. As shown in Figure 4.2, the parser constructs a constituency tree from the original caption, while the nodes hierarchically form phrases of different types. The common phrase types are Noun phrase (NP), Verb phrase (VP), Preposition phrase (PP), and Adjective phrase (AP).

To extract the objects in the skeleton sentence, we find the lowest level NP's, and keep the

last word within the phrase as skeletal object word. The words ahead of it within the same NP are attributes describing this skeletal object. The lowest level phrases of other types are kept in the skeleton sentence.

Sometimes, it is difficult to decide whether all the words except for the last one in a noun phrase are attributes. For example, the phrase "piggy bank" is a noun-noun compound. Should we keep "piggy bank" as a single entity, or use "piggy" as a modifier? In this work, we don't distinguish noun-noun compounds from other attribute-noun word phrases, and treat "piggy" as the attribute of "bank", as shown in Figure 4.2. Our experience is that the coarse-to-fine network can learn the correspondence, although strictly speaking they are not attribute-object pairs.

4.3.2 Coarse-to-fine LSTM

We use the high level image feature extracted from a CNN as the input feature to the language model. For the decoder part, our coarse-to-fine model consists of two LSTM submodels: one for generating skeleton sentences, and the other for generating attributes. We denote the two submodels as Skel-LSTM and Attr-LSTM respectively.

Skel-LSTM

The Skel-LSTM predicts the skeleton sentence given the image features. We adopt the soft attention based LSTM in [126] for the Skel-LSTM. Spatial information is maintained in the CNN image features, and an attention map is learned at every time step to focus attention to predict the current word.

We denote the image features at location $(i, j) \in L \times L$ as $v_{ij} \in \mathbb{R}^D$. The attention map at time step t is represented as normalized weights $\alpha_{ij,t}$, computed by a multilayer perceptron conditioned on the previous hidden state h_{t-1}.

$$\alpha_{ij,t} = \text{Softmax}(\text{MLP}(v_{ij}, h_{t-1})) \tag{4.1}$$

Then, the context vector z_t at time t is computed as:

$$z_t = \sum_{i,j} \alpha_{ij,t} v_{ij} \tag{4.2}$$

The context vector is then fed to the current time step LSTM unit to predict the upcoming word.

Unlike [126], in our model, the attention map $\alpha_{ij,t}$ is not only used to predict current skeletal word, but also to guide the attribute prediction: the attributes corresponding to a skeletal word describe the same skeletal object, and the attention information we get from Skel-LSTM can be reused in the Attr-LSTM to guide where to look.

Attr-LSTM

After the skeleton sentence is generated, the Attr-LSTM predicts the attribute sequence for each skeletal word. Rather than predicting multiple attribute words separately for one object, the Attr-LSTM can predict the attribute sequence as a whole, naturally taking care of the order of attributes. The Attr-LSTM is similar to the model in [117], with several modifications.

The original input sequence of the LSTM in [117] is:

$$x_{-1} = \text{CNN}(I) \tag{4.3}$$

$$x_t = W_e y_t, t = 0, 1, ..., N-1 \tag{4.4}$$

where I is the image, $\text{CNN}(I)$ is the CNN image features as a vector without spatial information, W_e is the learned word embedding, and y_t is the ground-truth word encoded as a one-hot vector. y_0 is a special start-word token.

In our coarse-to-fine framework, attribute generation is conditioned on the skeletal word it is describing. Therefore, apart from the image feature, the Attr-LSTM should be informed by the current skeletal word. On the other hand, the context of the skeleton sentence is also important

to give the Attr-LSTM a global understanding of the caption, rather than just focusing on the single current skeletal word. We experimented with feeding the skeletal hidden activations from different time steps into the Attr-LSTM, including the previous time step, the current time step, and the final time step, and found that the current time step hidden activations yield the best result. Moreover, as mentioned in Skel-LSTM, rather than using global image features as the input, we use attention-based image features to encourage the attribute predictor to focus on the current skeletal word.

We formulate the input of Attr-LSTM at the first time step as a multilayer network that fuses different sources of information into the embedding space:

$$x_{-1} = \text{MLP}(W_I z_T + W_t s_T^{skel} + W_h h_T^{skel}) \tag{4.5}$$

where T is the time step of the current skeletal word, $z_T \in \mathbb{R}^D$ is the attention weighted average of the image features, $s_T^{skel} \in \mathbb{R}^{m_s}$ is the embedding of the skeletal word at time T, $h_T^{skel} \in \mathbb{R}^{n_s}$ is the hidden state in the Skel-LSTM of dimension n_s. m_s and n_s are dimensionality of Skel-LSTM word embedding and LSTM unit respectively. W_l, W_t, W_h are learned parameters. The remaining input to Attr-LSTM is the same as Equation 4.4. The Attr-LSTM framework is illustrated in Figure 4.2.

In training stage, the ground truth skeleton sentence is fed into the Skel-LSTM, and s_T^{skel} is the ground truth skeleton word embedding. In test stage, s_T^{skel} is the embedding of predicted skeleton word.

Attention refinement for attribute prediction

Optionally, we can refine the attention map acquired in the Skel-LSTM for better localization of the skeletal word, thus improve the attribute prediction. The attention map α is a pre-word α that is generated before the word is predicted. It can cover multiple objects, or can even be in a

different location from the predicted word. Therefore, a refinement of the attention map after the prediction of the current word can provide more accurate guidance for the attribute prediction.

The LSTM unit at time step T outputs the word probability prediction $P_{attend} = (p_1, ..., p_Q)$, where Q is the vocabulary size in Skel-LSTM. In addition to the single weighted sum feature vector z_T, we can also use the feature vector v_{ij} in each location as input to the Skel-LSTM. Thus, for each of the L^2 locations, we can get the probability of word prediction P_{ij}. We can use the spatial word probability to refine the attention map α:

$$\alpha_{post(ij)} = \frac{1}{Z} P_{attend}^T \cdot P_{ij} \tag{4.6}$$

where Z is the normalization factor so that $\alpha_{post(ij)}$ sums to one. The refined post-word α is proportional to the similarity between P_{attend} and P_{ij}. In Figure 4.3, we illustrate the attention refinement process. For each word predicted by the Skel-LSTM, the attention map, predicted words for each location, and refined attention map are shown. When the word is an object, we can see how the refined attention map gets more accurate attention focusing on objects of interest (e.g. First image last word "bottle"). Since the object words are those that need Attr-LSTM to predict attributes for, the attention map improvement is helpful for those words by providing Attr-LSTM with more accurate attended area.

Fusion of Skeleton-Attribute

After attributes are predicted for all the skeletal words, attributes are merged into the skeleton sentence just before the corresponding skeletal word, and the final caption is formed.

4.3.3 Variable-length caption generation

Due to the imperfections in the current parser approach that we use, there are some cases where the parsing result is noisy. Most of the time, the noise is from incorrect noun

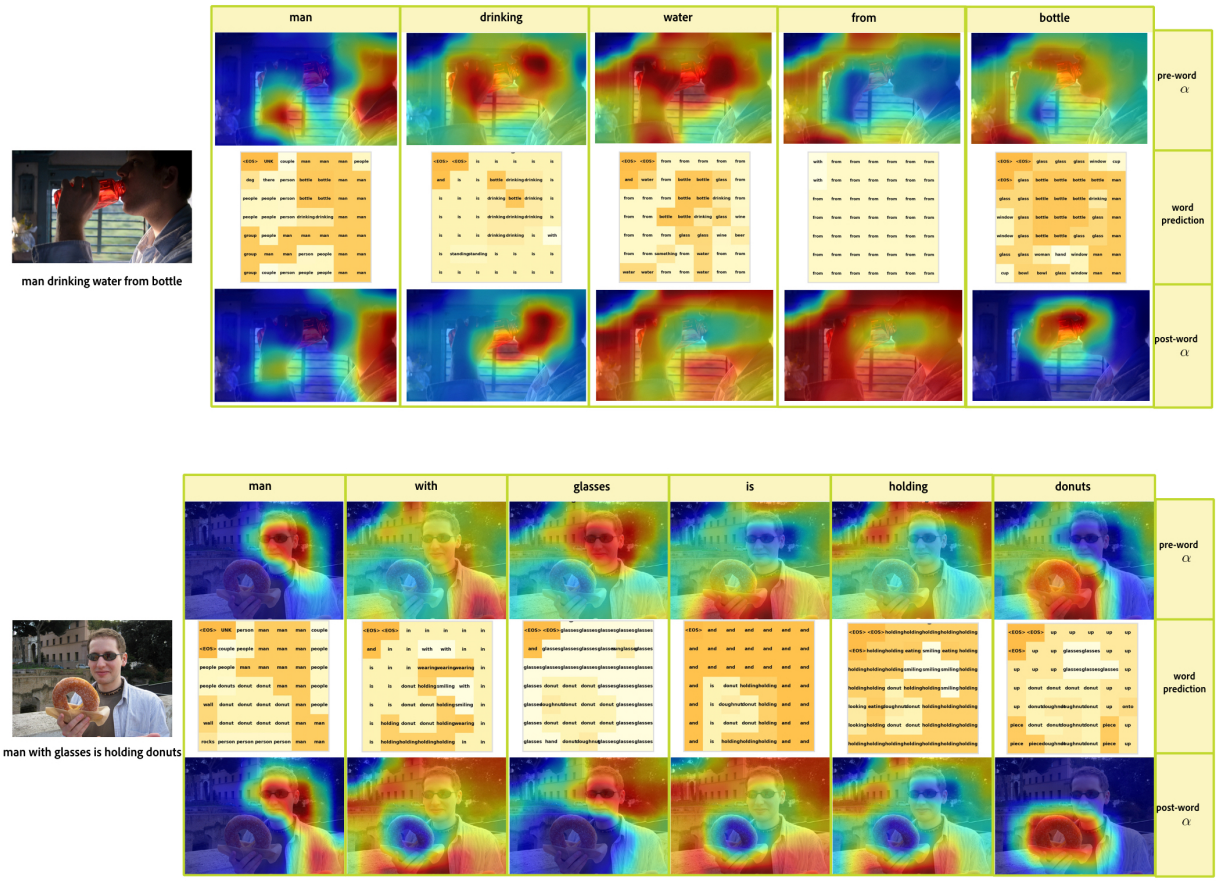


Figure 4.3: Illustration of attention refinement process during inference stage. All the skeleton words in generated skeleton sentence are shown. For each word, the attention map, predicted words for each location, and refined attention map are shown.

phrase recognition, and short skeleton sentences with one or several missing objects. This leads to a shorter skeleton prediction in the Skel-LSTM on average, thus eventually causes shorter predictions for the full sentence.

To overcome this problem, we designed a simple yet effective trick to vary the length of the generated sentence. Without modifying the trained network, In the inference stage of either Skel-LSTM or Attr-LSTM, we modify the sentence probability with a length factor:

$$\log(\hat{P}) = \log(P) + \gamma \cdot l \tag{4.7}$$

Where P is the probability of a generated sentence, and $\hat{P}$ is the modified sentence probability. l

is the length of the generated sentence. γ is the length factor to encourage or discourage longer sentences. Note that the modification is performed during generation of the each word rather than performed after the whole sentence is generated. It is equivalent to adding γ to each word log probability except for the end-of-sentence token $\langle \text{EOS} \rangle$ when sampling the next word from the word probability distribution. This trick of sentence probability modification works well together with beam search.

Our coarse-to-fine algorithm especially benefits from this mechanism, since it can be applied to either Skel-LSTM or Attr-LSTM, resulting in varied information in either objects, or the description of those objects. This allows us to generate captions according to user preference on the complexity of captions and amount of information in captions.

4.3.4 Tag enhancement

Works have shown that tags can be used to improve the quality of caption generation ([131] [34]). One intuitive way to use tag to enhance caption generation is similar to Section 4.3.3. In the inference stage, for a set of pre-detected tags, for either Skel-LSTM or Attr-LSTM, we modify the sentence probability:

$$\log(\hat{p}(w_i)) = \log(p(w_i)) + \alpha \cdot c(w_i) \tag{4.8}$$

where w_i is the ith word, and $c(w_i)$ is the confidence of the detected tag w_i. If w_i is not within the set of tags detected, $c(w_i) = 0$. α is the encourage factor that controls how much we emphasize on the tags.

4.4 Experiments

In this section, we describe our experiments on two datasets to test our proposed approach.

4.4.1 Datasets

We perform experiments on two datasets: the popular benchmark MS-COCO, and Stock3M, a new dataset with much larger scale and more natural captions.

MS-COCO has 123,287 images. Each image is annotated with 5 human generated captions, with an average length of 10.36 words. We use the standard training/test/validation split that is commonly used by other work [131, 123], and use 5000 images for testing, and 5000 images for validation.

MS-COCO is a commonly used benchmark for image captioning tasks. However, there are some issues with the dataset: the images are limited and biased to certain content categories, and the image set is relatively small. Moreover, the captions generated by AMT workers are not particularly natural. Therefore, we collected a new dataset: Stock3M. Storck3M contains 3,217,654 user uploaded images with a large variety of content. Each image is associated with one caption that is provided by the photo uploader on a stock website. The caption given by the photo uploader is more natural than those found in MS-COCO, and the dataset is 26 times larger in terms of number of images. The captions are much shorter than MS-COCO, with an average length of 5.25 words, but they are more challenging, due to a larger vocabulary and image content variety. We use 2000 images for validation and 8000 images for testing.

4.4.2 Experimental details

Preprocessing of captions

We follow the preprocessing procedure in [59] for the captions, removing the punctuations and converting all characters into lower case. For MS-COCO, we discard words that occur less than 5 times in skeleton sentences, and less than 3 times in attributes. This results in 7896 skeleton, and 5199 attribute words. In total, there are 9535 unique words. For the baseline method that processes the full sentences, a similar preprocessing procedure is applied to the full sentences.

Words that occur less than 5 times are discarded, resulting in 9567 unique words.

For Stock3M, due to the larger vocabulary size, we set the word occurrence thresholds to 30 for skeleton and 5 for attributes respectively. This results in 11047 skeleton and 12385 attribute words, with a total of 14290 unique words. In the baseline method that processes full sentences, the occurrence threshold is 30, resulting in 13788 unique words.

Image features and training details for MS-COCO

It has been argued that high level features such as attributes are better as input to caption-generating LSTMs [131, 123]. Our empirical finding is that by simply adopting a better network architecture that provides better image features, and fine-tuning the CNN within the caption dataset, the features extracted are already excellent inputs to the LSTM. We use ResNet-200 [48] as the encoder model. Images are resized to 256×256 and randomly cropped to 224×224. The layer before the average pooling layer and classification layer is used for the image features. and it outputs features with size $2048 \times 7 \times 7$, maintaining the spatial information.

Our system is implemented in Torch [20]. We fine-tune the CNN features as follows: first, the CNN features are fixed, and an LSTM is trained for full sentence generation. After the LSTM achieves reasonable results, we start fine-tuning the CNN with learning rate 1e-5. The fine-tuned CNN is then used for both Skel-LSTM and Attr-LSTM. The parameters for the Decoder network are as follows: word embedding is trained from scratch, with a dimension of 512. For Skel-LSTM, we set the learning rate 0.0001, and the hidden layer dimension 1800. For Attr-LSTM, the learning rate is 0.0004, and the hidden layer is 1024-dimensional. Adagrad is used for training. The learning rate is cut in half once after the validation loss stops dropping.

For tag enhancement, we follow [34] and use the same test result of tags.

Image features and training details for Stock3M

We use GoogleNet [107] fine-tuned on Stock3M as the CNN encoder, and add an embedding module after the 1024-dimensional output of GoogleNet $pool5/7 \times 7s1$ layer.

Stock3M is different from MS-COCO in that the images mostly contain single objects, and the captions are more concise than MS-COCO. The average length of Stock3M captions is about half that of MS-COCO. Hence, we did not observe improvement with the attention mechanism, because there are fewer things to focus on. For simplicity, we use the LSTM in [117] for Skel-LSTM. Consequently, for Attr-LSTM, there is no attention input in the -1 time step. We will show that even without attention, the coarse-to-fine algorithm improves substantially over baseline.

For tag enhancement, we fine-tune the GoogleNet on Stock and learn the Stock images and tags to the same embedding space, and use the cosine similarity between the image and tags as confidence score for detected tags.

Parameters in the testing stage

For both Skel-LSTM and Attr-LSTM, we use a beam search strategy, and adopt length factor γ as explained in Section 4.3.3. In Table 4.1, we list the hyper-parameters of the models we use for all the results we show in the main paper and here. The beam size and length factor γ are chosen on a validation set for both Stock3M and MS-COCO.

Table 4.1: Choice of beam size and length factor γ for both baseline model and our proposed coarse-to-fine model. The values are decided on validation set.

	Model	Beam size	length factor γ	Attribute beam size	attribute length factor γ
MS-COCO	Baseline	3	0.1	-	-
	Coarse-to-fine	5	0.1	2	0.9
Stock3M	Baseline	2	1.0	-	-
	Coarse-to-fine	2	1.2	2	0.3

4.4.3 Results

Evaluation metrics

Apart from the conventional evaluation metrics that are commonly used: Bleu [89], CIDEr [116], METEOR[25], and ROUGE [75], we use the recently proposed SPICE metric [2] which is not sensitive to n-grams and builds a scene graph from captions to encode the objects, attributes and relationships in the image. We emphasize our performance on this metric, because it has much higher correlation with human ratings than the other conventional metrics, and it shows the performance specific to different types of information, such as different types of attributes, objects, and relationships between objects.

Baseline

In order to demonstrate the effectiveness of our method, we also present a baseline result. The baseline method is trained and tested on full caption sentences, without skeleton-attribute decomposition. For each dataset, we use the same network architecture as in the Skel-LSTM architecture, and use the same hyper-parameters and the same CNN encoder as in our proposed coarse-to-fine method.

Quantitative results

We report both SPICE in Table 4.2 and conventional evaluation metrics in Table 4.3, and our results compared with baseline results in Table 4.4.

Table 4.2: Performance of our proposed method and the baseline method on SPICE measurement, for the two datasets. We also include the results on different semantic concept subcategories.

	Model	SPICE	Precision	Recall	Object	Relation	Attribute	Size	Color	Cardinality
MS-COCO	Baseline	0.188	0.508	0.117	0.350	0.048	0.098	0.045	0.132	0.039
	Ours	**0.196**	**0.529**	**0.123**	**0.363**	**0.050**	**0.110**	**0.073**	**0.170**	**0.064**
Stock3M	Baseline	0.157	0.173	0.166	0.250	0.049	0.077	0.129	0.135	-
	Ours	**0.172**	**0.190**	**0.185**	**0.276**	**0.061**	**0.081**	**0.144**	**0.151**	-

Table 4.3: Performance of our proposed methods and other state-of-the-art methods on MS-COCO and Stock3M. Only scores that were reported in the papers are shown here.

Datasets	Models	B-1	B-2	B-3	B-4	METEOR	ROUGE-L	CIDEr
MS-COCO	NIC [117]	-	-	-	0.277	0.237	-	0.855
	LRCN [30]	0.669	0.489	0.349	0.249	-	-	-
	Toronto [126]	0.718	0.504	0.357	0.250	0.230	-	-
	ATT [131]	0.709	0.537	0.402	0.304	0.243	-	-
	ACVT [123]	0.74	0.56	0.42	0.31	0.26	-	0.94
	Baseline	0.742	0.577	0.442	0.340	0.268	0.552	1.069
	Ours	0.742	0.577	0.440	0.336	0.268	0.552	1.073
	Baseline (w/o a)	0.664	0.481	0.351	0.258	0.245	0.485	0.949
	Ours (w/o a)	**0.673**	**0.489**	**0.355**	**0.259**	**0.247**	**0.489**	**0.966**
Stock3M	Baseline	0.236	0.133	0.079	0.050	0.108	0.233	0.720
	Ours	**0.245**	**0.138**	**0.083**	**0.052**	**0.110**	**0.239**	**0.724**
	Baseline (w/o a)	0.233	0.134	0.082	0.053	0.108	0.235	0.737
	Ours (w/o a)	**0.246**	**0.140**	**0.086**	**0.055**	**0.111**	**0.241**	**0.738**

Table 4.4: Performance of our proposed methods and baseline method on MS-COCO and Stock3M, using tags to enhance performance.

Datasets	Models	B-1	B-2	B-3	B-4	METEOR	ROUGE-L	CIDEr	SPICE
MS-COCO	Baseline (Tag)	0.746	0.582	0.447	0.343	0.272	0.557	1.091	0.193
	Ours (Tag)	**0.754**	**0.590**	**0.451**	**0.344**	**0.275**	**0.560**	**1.097**	**0.198**
Stock3M	Baseline (Tag)	0.240	0.136	0.082	**0.052**	0.110	0.236	0.735	0.159
	Ours (Tag)	**0.248**	**0.139**	**0.083**	**0.052**	**0.113**	**0.243**	**0.739**	**0.175**

First, it is worth mentioning that our baseline method is a very strong baseline. In Table 4.3, we compare our method with published state-of-the-art methods. Our baseline method already outperforms the state-of-the-art by a considerable margin, indicating the importance of a powerful image feature extractor. By just fine-tuning the CNN with the simple baseline algorithm, we outperform the approaches with augmentation of high level attributes [131, 123]. The baseline already ranks 3rd - 4th place on the MS-COCO CodaLab leaderboard [1].

In Table 4.5, we show our submission to MS-COCO online testing server. The server evaluates models with 40,775 test images with ground-truth captions that competitors do not have access to. We also show the other published state-of-the-art results in Table 4.5. Note that we do not use any commonly used augmentation tricks such as ensembling, or scheduled sampling [118], which can improve the performance further.

Table 4.5: Performance of our method on online MS-COCO testing server). We also show the results of other published state-of-the-art results.

Models	Bleu-1		Bleu-2		Bleu-3		Bleu-4		METEOR		ROUGE-L		CIDEr	
	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40
Ours	**0.734**	**0.912**	**0.564**	**0.829**	**0.425**	**0.724**	**0.320**	**0.612**	**0.262**	**0.356**	**0.542**	**0.698**	**1.011**	**1.026**
ATT_VC [131]	0.731	0.900	0.565	0.815	0.424	0.709	0.316	0.599	0.250	0.335	0.535	0.682	0.943	0.958
OriolVinyals [118]	0.713	0.895	0.542	0.802	0.407	0.694	0.309	0.587	0.254	0.346	0.530	0.682	0.943	0.946

SPICE is an F-score of the matching tuples in predicted and reference scene graphs. It can be divided into meaningful subcategories. In Table 4.2 we report the SPICE score as well as the subclass scores of objects, relations and attributes. In particular, size, color and count attributes are reported. As in Table 4.2, consistent improvement over the baseline across the two datasets is achieved, and the improvement is also consistent on the subcategories. The cardinality F-score for Stock3M is not reported here because there are too few images with this type of attribute to have a meaningful evaluation: there are only 78 cardinality attributes out of 8000 test images.

In Table 4.3, we also show the comparison between the proposed method and baseline method on conventional evaluation metrics. As shown, there is no significant improvement over

[1] `https://competitions.codalab.org/competitions/3221`

baseline on most of the conventional metrics on MS-COCO. This is due to the intrinsic problem of the conventional metrics: they highly rely on n-gram matching. The proposed coarse-to-fine algorithm breaks the original word order of the training captions, and thus weakens the objective of predicting exact n-grams as in the training captions. There is even a small drop on BLEU-3 and BLEU-4 on MS-COCO against the baseline. To investigate if the two methods indeed have similar performance as reflected in those conventional metrics, we conducted further analysis on the results.

We preprocess the ground-truth and predicted captions to remove all the *a*'s in the captions. This is motivated by the observation that 15% of words in the MS-COCO captions are *a*. This functional word affects the n-gram match greatly, though it conveys very little information in the MS-COCO like captions. Therefore, by removing the *a*'s in the captions, we obtain a measure that is not influenced by the n-grams using *a*, and hence is more focused on content words. The performance evaluation on the same datasets with *a* removed is shown in Table 4.3 as "Baseline/Ours (w/o *a*)". It can be seen that consistent improvement is achieved with our coarse-to-fine method.

In Table 4.3, we also present the performance of our coarse-to-fine method as well as the baseline method on Stock3M evaluated on conventional metrics. In Stock3M, the frequency of the word *a* is only 2.5%, therefore it has no big impact on the relative performance of the two methods. We can see consistent improvement on all the metrics.

In Table 4.4, we show the effect of using tag as enhancement for both our coarse-to-fine method and the baseline method on MS-COCO and Stock3M. The methods are evaluated on both conventional metrics and SPICE. We can see that for MS-COCO, using tags helps for both baseline and our coarse-to-fine method, but the gain for coarse-to-fine model is greater. This is because our coarse-to-fine model is inherently more suitable for using detected tag as cues: tags are naturally divided into two types: skeleton tags and attribute tags, and the skeleton sentence and attributes are enhanced by different types of tags. For stock3M, tags also help for both baseline

method and coarse-to-fine method. However, the improvement tag enhancement brings is not as obvious, and this is because the tags in Stock3M are not directly taken from the vocabulary of captions, and the distribution of tags are not the same with captions, which makes the predicted tags less helpful for caption generation.

4.4.4 Analysis of generated descriptions

Comparison of our coarse-to-fine method and baseline method

In Figure 4.4 and Figure 4.5, we show some qualitative results of generated captions from our proposed method and the baseline method on MS-COCO and Stock3M datasets respectively. Captions in green text boxes are generated by the proposed coarse-to-fine method, and captions in red text boxes are generated by the baseline method. We can see that for most images, our proposed method outperforms the baseline method in two different aspects: 1) more accurate number/color attributes (e.g. Figure 4.4 1st column 2nd and 6th image; Figure 4.4 2nd column 5th image; Figure 4.5 1st column 2nd and 5th image); 2) more accurate skeleton captions with better objects (e.g. Figure 4.4 1st column 3rd image; Figure 4.4 2nd column 1st and 2nd image; Figure 4.5 2nd column 1st image).

We also show 5 examples in which coarse-to-fine method performs no better than or worse than baseline method in Figure 4.4 and Figure 4.5. The examples are shown in the third column, on the right of Figure 4.4 and Figure 4.5. We can see that wrong recognition of objects or missing main objects in the image is still the dominant cause of error.

Generating variable-length captions.

In the coarse-to-fine algorithm, a length factor is applied to the Skel-LSTM and Attr-LSTM separately to encourage longer skeleton/attribute generation in order to generate captions that have similar length to the training captions. However, we can further manually tune the length

Figure 4.4: Qualitative comparison of our proposed algorithm (green text box to the right of each image) and baseline (red text box) on MS-COCO. On the left, we can see our method outperforms baseline method. On the right, we can see some examples on which either methods generates captions with clear flaws.

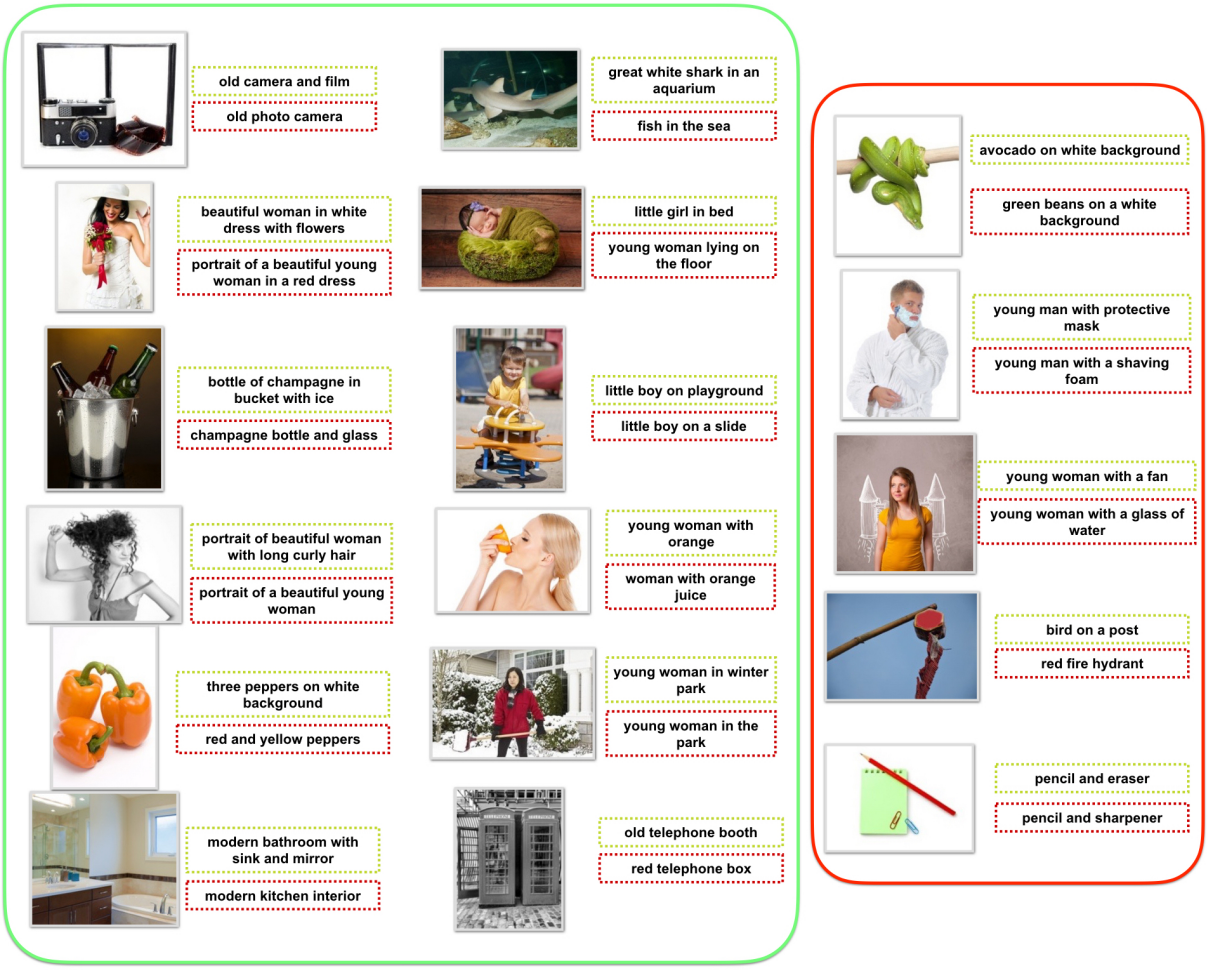


Figure 4.5: Qualitative comparison of our proposed algorithm (green text box to the right of each image) and baseline (red text box) on Stock3M. On the left, we can see our method outperforms baseline method. On the right, we can see some examples on which either methods generates captions with clear flaws.

factor value to control the length of skeleton/attribute of the generated captions. In Figure 4.6, we show some test examples from Stock3M and MS-COCO . For each of the images, four captions are generated with four pairs of (skeleton, attribute) length factor values: (-1, -1), (1.5, -1), (-1, 1.5), (1.5, 1.5). The four value pairs represent all combinations of encouraging less/more information in skeleton/attributes. Attributes are marked in red in the generated caption. We can see how the length factor works together with beam search to get syntactically and semantically correct captions. The amount of object/attribute information naturally varies with the length of the skeleton/attributes.

We can certainly apply the same trick on the baseline method using different length factor values. For comparison, in Figure 4.6 (red box), we show the four captions generated from baseline method also using four different length factor values: $\gamma \in \{-1, -0.5, 0.5, 1.5\}$. As illustrated, although the captions generated by the baseline model can also have different lengths, they are much less flexible and useful than the ones generated by our coarse-to-fine model. This is because the coarse-to-fine model can decompose the caption into skeletons and attributes, and have separate requirements for objects and attributes according to user preference: the user may prefer descriptions that only describe main objects but in more details; or he/she may prefer descriptions that contain all the objects in the images, but cares less about the object attributes.

In Figure 4.7 and Figure 4.8, we show more examples from MS-COCO and Stock3M. We can see that the captions generated by our coarse-to-fine model are much more flexible and useful than the ones generated by the baseline method. And we can also observe the effect of separate control of attribute/skeleton in our coarse-to-fine model.

Post-word α helps with attribute prediction

The results we show in Table 4.2 and 4.3 for proposed coarse-to-fine model adopts attention refinement for attribute prediction in the Attr-LSTM on MS-COCO. Here, we further validate the effectiveness of the post-word α refinement approach in Table 4.6, by comparing

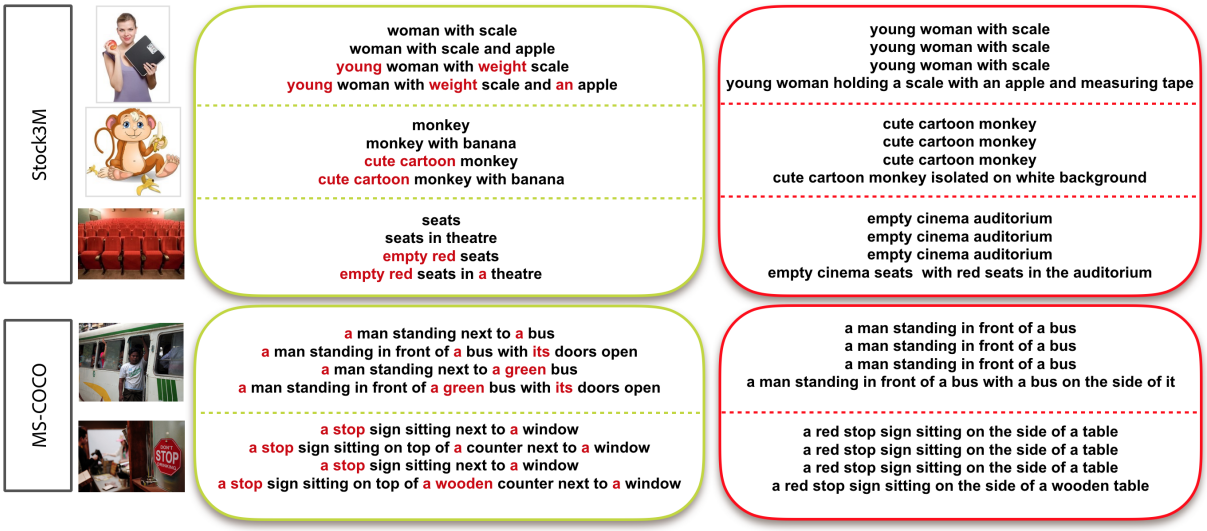


Figure 4.6: Examples of predicted titles for image examples from Stock3M and MS-COCO. Four titles are generated from our coarse-to-fine model (middle, in green box) and baseline model (right, in red box) respectively, using four pairs of length factor γ.

Figure 4.7: More examples of predicted titles for image examples from MS-COCO. Four titles are generated from our coarse-to-fine model (middle, in green box) and baseline model (right, in red box) respectively, using four pairs of length factor γ.

Figure 4.8: More examples of predicted titles for image examples from Stock3M. Four titles are generated from our coarse-to-fine model (middle, in green box) and baseline model (right, in red box) respectively, using four pairs of length factor γ.

the result without attention refinement (Pre-word α) with the result with attention refinement (Post-word α). The post-word α only refines the attended area for attribute prediction, therefore we only show the improvement of SPICE score on attribute subcategories. The performance on other categories is unchanged. We see consistent improvement across different types of attributes, especially on color and size. This proves that a good attention map helps effectively on attribute prediction.

Table 4.6: Comparison of our proposed method with and without post-word α attention on MS-COCO.

Model	Attribute	Color	Size	Cardinality
Pre-word α	0.107	0.167	0.069	0.063
Post-word α	**0.110**	**0.170**	**0.073**	**0.064**

The ability of generating unique and novel captions

It has been pointed out that the current LSTM based method has a problem generating sentences that have not been seen in the training set, and generates the same sentences for different test images [26]. This means that the LSTM dynamics are caught in a rut of repeating the sequences it was trained on for visually similar test images, and is less capable of generating unique sentences for a new image with an object/attribute composition that is not seen in the training set. With the skeleton-attribute decomposition, we claim that our algorithm can generate more unique captions, and can give more accurate attributes even when the attribute-object pattern is new to the system. As shown in Table 4.7, our coarse-to-fine model increases the percentage of generated unique captions by 3%, and increases the percentage of novel captions by 8%.

Table 4.7: Percentage of generated unique sentences and captions seen in training captions for the baseline method and our coarse-to-fine method. The statistics are gathered from the test set of MS-COCO containing 5000 images.

Model	Unique captions	Seen in Training
Baseline	63.96%	56.06%
coarse-to-fine	66.96%	47.76%

Difference between SPICE measurement and METEOR measurement

Figure 4.9: Examples of some generated captions that get high score on SPICE but get low score on METEOR.

In Figure 4.9, we demonstrate that different evaluation metrics can yield opposite judgement results for a pair of generated captions. In the main paper, we emphasize our results on SPICE metrics rather than the conventional metrics. Here, in order to validate our claim, we show the comparison between SPICE and METEOR metrics. For each image in Figure 4.9,

there are three text boxes below it: the caption in green text box is generated by our proposed coarse-to-fine model; the caption in red box is generated by baseline model; the caption(s) in black text box is(are) ground-truth caption(s). For Stock3M, there is one ground-truth caption per image; for MS-COCO, there are 5 ground-truth captions per image. For the predicted captions by two models, we also show in parentheses the SPICE score (S) and METEOR score (M) for each of the predictions. We can see that for the first two rows of eight images, coarse-to-fine model yields higher SPICE score but lower METEOR score, and human judgement is closer to SPICE score than METEOR score.

For example, the first image in the first row is about the tea leaves, and the main object "tea" occurs in the coarse-to-fine model prediction, but not in the baseline prediction. Baseline prediction recognized the main object in the image wrongly to coffee beans. However, baseline prediction has higher METEOR, because of the mention of "pile of". However, the correct mention of objects is obviously more important in human judgement.

In the third row of Figure 4.9, we also show three examples from MS-COCO for which METEOR scores are closer to human judgement. This shows that the SPICE measurement is still not perfect.

4.5 Conclusion

In this paper, we propose a coarse-to-fine model for image caption generation. The proposed model decomposes the original image caption into a skeleton sentence and corresponding attributes, and formulates the captioning process in a natural way in which the skeleton sentence is generated first, and then the objects in the skeleton are revisited for attribute generation. We show with experiments on two challenging datasets that the coarse-to-fine model can generate better and more unique captions over a strong baseline method. Our proposed model can also generate descriptive captions with variable lengths separately for skeleton sentence and attributes,

and this allows for caption generation according to user preference.

In the future work, we plan to investigate more complicated skeleton/attribute decomposition approaches, and allow for attributes that appear after the skeletal object. It is also of interest to design a model that automatically decides on the length of generated caption based on the visual complexity of the image.

4.6 Acknowledgements

Chapter 4, in full, is an edited reprint of the material as it appears in the following publication: Wang, Y., Zhe, L., Shen, X., Cohen, S., Cottrell, G. W., "Skeleton Key: Image Captioning by Skeleton-Attribute Decomposition", In *Computer Vision and Pattern Recognition (CVPR)*, 2017. The dissertation author was a primary researcher and an author of the cited material.

This work was supported in part by NSF grants SMA-1041755, IIS-1219252 to G. W. Cottrell, and gift money from Adobe Research to Garrison. W. Cottrell.

Chapter 5

Conclusion

In this thesis, I focused on image understanding with deep learning. In Chapter 2 and 3, I focused on album-wise image understanding. The understanding of an album is divided into two folds: learning the event type of the album, and finding the important images in the album given the specific event type of the album. With a dataset we collected for this problem, we demonstrated that the event-specific image importance property is learnable despite its highly subjectiveness. I proposed a siamese network to predict the image importance given the album event type it is from, and an iterative procedure which involves a CNN based model and an LSTM based model to learn the event type of the given album, and the siamese model that learns the importance score of each image in the album. I demonstrated that the two problems in understand event albums can be solved together, and they can in turn help the performance of each other. In Chapter 4, I focused on another image understanding problem: image captioning. I proposed a coarse-to-fine model which decomposes the original image captions into skeleton sentence and its attributes, and by generating the skeleton and attributes in a two stage manner, we managed to improve the image captioning performance, and moreover, our model was able to adjust the amount of information a generated caption contains according to user preference. With these two sub-problems, the thesis provides the way for better image understanding in two aspects.

Despite the effort presented in the thesis, there is a lot more to be explored. To achieve the final goal of event album curation, we not only need image importance score independently for each image, but also need to consider other aspects of the sub-album property, for example, the coverage of the album, and the uniqueness of each image curated. On the other hand, the album types we learnt are pre-defined event types, but in reality, there are much more event types than the selected event types we study, and many personal events are mixture of multiple typical event types. Therefore, a model that deals with an arbitrary event type is desirable. For image captioning, although our model generates more unique captions with more variance, the captions generated in general suffer from the problem of parroting back the training corpus, and the quality of machine generated descriptions are still much less satisfactory than human generated captions.

The simple attempt of using tags to guide the captions in the thesis shows one possibility of improving the quality of captions, which can be further investigated.

On the other hand, the two sub-problems I studied are the two components of a more challenging task for album understanding: generating narrative paragraph for a personal album [76]. It is interesting to study how the image importance interacts with paragraph generation, and how captions generated for each image can in turn help decide the importance of the images.

Bibliography

[1] P. Anderson, B. Fernando, M. Johnson, and S. Gould. Spice: Semantic propositional image caption evaluation. In *ECCV*, 2016.

[2] P. Anderson, B. Fernando, M. Johnson, and S. Gould. Spice: Semantic propositional image caption evaluation. In *ECCV*, 2016.

[3] S. Bacha, M. S. Allili, and N. Benblidia. Event recognition in photo albums using probabilistic graphical model and feature relevance. In *International Conference on Pattern Recognition (ICPR)*, 2016.

[4] R. A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., 1999.

[5] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014.

[6] M. Bar. A cortical mechanism for triggering top-down facilitation in visual object recognition. *Journal of Cognitive Neuroscience*, 2003.

[7] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio. Theano: new features and speed improvements. Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012.

[8] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008.

[9] Y. Benjamini and D. Yekutieli. The control of the false discovery rate in multiple testing under dependency. *Annals of Statistics*, 29:1165–1188, 2001.

[10] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio. Theano: a CPU and GPU math expression compiler. In *Proceedings of the Python for Scientific Computing Conference (SciPy)*, June 2010. Oral Presentation.

[11] L. Bossard, M. Guillaumin, and L. Van. Event recognition in photo collections with a stopwatch hmm. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, 2013.

[12] J. Bullier. Integrated model of visual processing. *Brain Research Review*, 2001.

[13] J. Bulliera and L. G. Nowakb. Parallel versus serial processing: new vistas on the distributed organization of the visual system. *Current Opinion in Neurobiology*, 1995.

[14] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, June 1986.

[15] A. Ceroni, V. Solachidis, C. Niederée, O. Papadopoulou, N. Kanhabua, and V. Mezaris. To keep or not to keep: An expectation-oriented photo selection method for personal photo collections. In *Proceedings of the 5th International Conference on Multimedia Retrieval (ICMR)*, 2015.

[16] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2014.

[17] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *CoRR*, abs/1606.00915, 2016.

[18] W. Chen, T. yan Liu, Y. Lan, Z. Ma, and H. Li. Ranking measures and loss functions in learning to rank. In *Advances in Neural Information Processing Systems 22 (NIPS)*, 2009.

[19] K. Cho, B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078, 2014.

[20] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, 2011.

[21] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):603–619, May 2002.

[22] J. Dai, K. He, and J. Sun. Instance-aware semantic segmentation via multi-task network cascades. *CoRR*, abs/1512.04412, 2015.

[23] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Studying aesthetics in photographic images using a computational approach. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part III*, ECCV'06, pages 288–301, Berlin, Heidelberg, 2006. Springer-Verlag.

[24] M. Del Fabro, A. Sobe, and L. Böszörmenyi. Summarization of real-life events based on community-contributed content. In *Proceedings of the Fourth International Conferences on Advances in Multimedia (MMEDIA)*, 2012.

[25] M. Denkowski and A. Lavie. Meteor universal: Language specific translation evaluation for any target language. In *In Proceedings of the Ninth Workshop on Statistical Machine Translation*, 2014.

[26] J. Devlin, H. Cheng, H. Fang, S. Gupta, L. Deng, X. He, G. Zweig, and M. Mitchell. Language models for image captioning: The quirks and what works. *CoRR*, abs/1505.01809, 2015.

[27] J. Devlin, S. Gupta, R. B. Girshick, M. Mitchell, and C. L. Zitnick. Exploring nearest neighbor approaches for image captioning. *CoRR*, 2015.

[28] S. Dhar, V. Ordonez, and T. L. Berg. High level describable attributes for predicting aesthetics and interestingness. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[29] M. Dixit, S. Chen, D. Gao, N. Rasiwasia, and N. Vasconcelos. Scene classification with semantic fisher vectors. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

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

[31] J. Duncan. Selective attention and the organization of visual information. *Journal of Experimental Psychology*, 1984.

[32] D. Elliott and F. Keller. Image description using visual dependency representations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1292–1302, 2013.

[33] A. K. Engel, P. Fries, and W. Singer. Dynamic predictions: Oscillations and synchrony in top-down processing. *Nature Reviews Neuroscience*, 2001.

[34] H. Fang, S. Gupta, F. N. Iandola, R. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig. From captions to visual concepts and back. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[35] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In *Proceedings of the 11th European Conference on Computer Vision: Part IV*, ECCV'10, pages 15–29, Berlin, Heidelberg, 2010. Springer-Verlag.

[36] R. Fergus, P. Perona, and A. Zisserman. Weakly supervised scale-invariant learning of models for visual recognition. *International Journal of Computer Vision*, 71(3):273–303, 2007.

[37] 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 *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2568–2577, June 2015.

[38] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *CoRR*, abs/1508.06576, 2015.

[39] C. D. Gilbert, M. Sigman, and R. E. Crist. The neural basis of perceptual learning. *Neuron*, 31(5), 2001.

[40] R. Girshick. Fast r-cnn. In *International Conference on Computer Vision (ICCV)*, 2015.

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

[42] Y. Gong, Y. Jia, T. Leung, A. Toshev, and S. Ioffe. Deep convolutional ranking for multilabel image annotation. *CoRR*, abs/1312.4894, 2013.

[43] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[44] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2009.

[45] M. Gygli, H. Grabner, H. Riemenschneider, F. Nater, and L. J. V. Gool. The interestingness of images. In *IEEE International Conference on Computer Vision, ICCV*, 2013.

[46] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *In Proc. Computer Vision and Pattern Recognition Conference (CVPR)*, 2006.

[47] C. Harris and M. Stephens. A combined corner and edge detector. In *In Proc. of Fourth Alvey Vision Conference*, pages 147–151, 1988.

[48] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[49] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *CoRR*, abs/1502.01852, 2015.

[50] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 1997.

[51] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Int. Res.*, 47(1):853–899, May 2013.

[52] B. K. Horn and B. G. Schunck. Determining optical flow. Technical report, Cambridge, MA, USA, 1980.

[53] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[54] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

[55] P. Isola, D. Parikh, A. Torralba, and A. Oliva. Understanding the intrinsic memorability of images. In *Advances in Neural Information Processing Systems 24*. Curran Associates, Inc., 2011.

[56] P. Isola, J. Xiao, A. Torralba, and A. Oliva. What makes an image memorable? In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[57] M. Jas and D. Parikh. Image specificity. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[58] 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.

[59] A. Karpathy and F. Li. Deep visual-semantic alignments for generating image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 3128–3137, 2015.

[60] A. Khosla, A. D. Sarma, and R. Hamid. What makes an image popular? In *International World Wide Web Conference (WWW)*, 2014.

[61] R. Kiros, R. Salakhutdinov, and R. Zemel. Multimodal neural language models. In T. Jebara and E. P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 595–603. JMLR Workshop and Conference Proceedings, 2014.

[62] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *CoRR*, abs/1411.2539, 2014.

[63] D. Klein and C. D. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 423–430, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[64] A. Kolesnikov and C. H. Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. *CoRR*, abs/1603.06098, 2016.

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

[66] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Babytalk: Understanding and generating simple image descriptions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12):2891–2903, Dec 2013.

[67] P. Kuznetsova, V. Ordonez, T. L. Berg, and Y. Choi. Treetalk: Composition and compression of trees for image descriptions.

[68] R. Lebret, P. H. O. Pinheiro, and R. Collobert. Phrase-based image captioning. *CoRR*, abs/1502.03671, 2015.

[69] Y. Lecun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 5 2015.

[70] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Comput.*, 1(4):541–551, Dec. 1989.

[71] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller. Efficient backprop. In *Neural Networks: Tricks of the Trade, This Book is an Outgrowth of a 1996 NIPS Workshop*, pages 9–50, London, UK, UK, 1998. Springer-Verlag.

[72] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua. A convolutional neural network cascade for face detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[73] 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*, 2007.

[74] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, CoNLL '11, pages 220–228, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[75] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Proc. ACL workshop on Text Summarization Branches Out*, page 10, 2004.

[76] Y. Liu, J. Fu, T. Mei, and C. W. Chen. Storytelling of photo stream with bidirectional multi-thread recurrent neural network. *CoRR*, abs/1606.00625, 2016.

[77] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *CVPR (to appear)*, Nov. 2015.

[78] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004.

[79] X. Lu, Z. Lin, H. Jin, J. Yang, and J. Z. Wang. Rapid: Rating pictorial aesthetics using deep learning. In *Proceedings of the ACM International Conference on Multimedia*, 2014.

[80] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.

[81] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). *ICLR*, 2015.

[82] R. Mattivi, J. Uijlings, F. deNatale, and N. Sebe. Exploitation of time constraints for (sub-) event recognition. In *ACM Workshop on Modeling and Representing Events (J-MRE 11)*, 2011.

[83] M. Mitchell, X. Han, J. Dodge, A. Mensch, A. Goyal, A. Berg, K. Yamaguchi, T. Berg, K. Stratos, and H. Daumé, III. Midge: Generating image descriptions from computer vision detections. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12, pages 747–756, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[84] M. C. Mozer. A focused backpropagation algorithm for temporal pattern recognition. *Complex Systems*, 3(4), 1989.

[85] M. Nagel, T. E. J. Mensink, and C. G. M. Snoek. Event fisher vectors: Robust encoding visual diversity of visual streams. In *British Machine Vision Conference*, 2015.

[86] U. Neisser. *Cognitive Psychology*. 1967.

[87] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *Neural Information Processing Systems (NIPS)*, 2011.

[88] P. Over, G. Awad, M. Michel, J. Fiscus, W. Kraaij, A. F. Smeaton, G. Quéenot, and R. Ordelman. Trecvid 2015 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2015*. NIST, USA, 2015.

[89] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[90] S. Park and N. Kwak. Cultural event recognition by subregion classification with convolutional neural network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2015.

[91] J. Redmon and A. Farhadi. YOLO9000: better, faster, stronger. *CoRR*, abs/1612.08242, 2016.

[92] S. Reiter, B. W. Schuller, and G. Rigoll. A combined LSTM-RNN - HMM - approach for meeting event segmentation and recognition. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing, ICASSP 2006, Toulouse, France, May 14-19, 2006*, pages 393–396, 2006.

[93] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[94] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[95] F. Sadeghi, J. Tena, A. Farhadi, and L. Sigal. Learning to select and order vacation photographs. In *Winter Conference on Applications of Computer Vision (WACV)*, 2015.

[96] H. Sak, A. W. Senior, and F. Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*.

[97] A. Salvador, M. Zeppelzauer, D. Manchon-Vizuete, A. Calafell-Orós, and X. Giró-i Nieto. Cultural event recognition with visual convnets and temporal models. In *CVPR ChaLearn Looking at People Workshop 2015*, 06/2015 2015.

[98] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1605.06211, 2016.

[99] I. Simon, N. Snavely, and S. M. Seitz. Scene summarization for online image collections. In *IEEE International Conference on Computer Vision, ICCV*, 2007.

[100] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[101] P. Sinha, S. Mehrotra, and R. Jain. Summarization of personal photologs using multidimensional content and context. In *ICMR*, 2011.

[102] R. Socher, A. Karpathy, Q. V. Le, C. D. Manning, and A. Y. Ng. Grounded compositional semantics for finding and describing images with sentences. *TACL*, 2:207–218, 2014.

[103] R. H. Sumit Chopra and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proc. of Computer Vision and Pattern Recognition Conference, IEEE Press*, 2005.

[104] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[105] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112, 2014.

[106] C. Szegedy, S. Ioffe, and V. Vanhoucke. Inception-v4, inception-resnet and the impact of residual connections on learning. *CoRR*, abs/1602.07261, 2016.

[107] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[108] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '14, pages 1701–1708, Washington, DC, USA, 2014. IEEE Computer Society.

[109] Y. H. Tan and C. S. Chan. phi-lstm: A phrase-based hierarchical LSTM model for image captioning. *CoRR*, abs/1608.05813, 2016.

[110] K. Tanaka. Inferotemporal cortex and object vision. *Annual Review of Neuroscience*, 1996.

[111] K. Tang, L. Fei-Fei, and D. Koller. Learning latent temporal structure for complex event detection. In *CVPR*, 2012.

[112] Y. Tang. Deep learning using linear support vector machines. In *Workshop on Representational Learning, ICML*, 2013.

[113] 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.

[114] S. Tsai, L. Cao, F. Tang, and T. S. Huang. Compositional object pattern: a new model for album event recognition. In *Proceedings of the 19th International Conference on Multimedia 2011*, 2011.

[115] S. Tschiatschek, R. Iyer, H. Wei, and J. Bilmes. Learning mixtures of submodular functions for image collection summarization. In *Neural Information Processing Systems (NIPS)*, 2014.

[116] R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 4566–4575, 2015.

[117] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. *CoRR*, abs/1411.4555, 2014.

[118] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge. *CoRR*, abs/1609.06647, 2016.

[119] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. pages 511–518, 2001.

[120] T. C. Walber, A. Scherp, and S. Staab. Smart photo selection: Interpret gaze as personal interest. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2014.

[121] L. Wang, Z. Wang, Y. Qiao, and L. V. Gool. Transferring object-scene convolutional neural networks for event recognition in still images. *CoRR*, abs/1609.00162, 2016.

[122] X. Wang and Q. Ji. Hierarchical context modeling for video event recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2016.

[123] Q. Wu, C. Shen, L. Liu, A. Dick, and A. van den Hengel. What value do explicit high level concepts have in vision to language problems. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*, 2016.

[124] Z. Wu, Y. Huang, and L. Wang. Learning representative deep features for image set analysis. *IEEE Trans. Multimedia*, 17(11):1960–1968, 2015.

[125] Y. Xiong, K. Zhu, D. Lin, and X. Tang. Recognize complex events from static images by fusing deep channels. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[126] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In D. Blei and F. Bach, editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 2048–2057. JMLR Workshop and Conference Proceedings, 2015.

[127] Z. Xu, Y. Yang, and A. G. Hauptmann. A discriminative cnn video representation for event detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1798–1807, 2015.

[128] Y. Yang, C. L. Teo, H. Daumé, III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 444–454, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[129] C.-H. Yeh, Y.-C. Ho, B. A. Barsky, and M. Ouhyoung. Personalized photograph ranking and selection system. In *Proceedings of the International Conference on Multimedia*, 2010.

[130] J. D. X. J. Yi Li, Haozhi Qi and Y. Wei. Fully convolutional instance-aware semantic segmentation. 2017.

[131] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo. Image captioning with semantic attention. *CoRR*, abs/1603.03925, 2016.

[132] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[133] M. D. Zeiler. Adadelta: An adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.

[134] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems 27 (NIPS)*, 2014.