January 2016

Content-based Image Understanding with Applications to Affective Computing and Person Recognition in Natural Settings

Ming Chen
Purdue University

Graduate School Form
30 Updated 12/26/2015

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Ming Chen

Entitled
CONTENT-BASED IMAGE UNDERSTANDING WITH APPLICATIONS TO AFFECTIVE COMPUTING AND PERSON RECOGNITION IN NATURAL SETTINGS

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

JAN P. ALLEBACH Chair	QIAN LIN
MIREILLE BOUTIN	
MARY L. COMER	
EDWARD J. DELP	

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy of Integrity in Research" and the use of copyright material.

Approved by Major Professor(s): JAN P. ALLEBACH

Approved by: V. Balakrishnan 12/06/2016

Head of the Departmental Graduate Program Date

CONTENT-BASED IMAGE UNDERSTANDING WITH APPLICATIONS TO AFFECTIVE COMPUTING AND PERSON RECOGNITION IN NATURAL SETTINGS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Ming Chen

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2016

Purdue University

West Lafayette, Indiana

ACKNOWLEDGMENTS

This dissertation is the result of many experiences I have encountered at Purdue from dozens of remarkable individuals who I wish to acknowledge. First and foremost, I would like to gratefully and sincerely thank my major advisor, Prof. Jan P. Allebach, who has been a tremendous mentor for me. I would like to thank you for encouraging my research, giving me the moral support, providing the research assistantship during my study, and most importantly, always supporting my life choices during these years. I would like to express my sincere appreciation. Second, I would like to thank Dr. Qian Lin, who is my mentor at HP Labs. It is with your great effort and brainstorming that we finish the last chapter of this dissertation. I would also like to thank my committee members, Prof. Mireille Boutin, Prof. Mary L. Comer, and Prof. Edward J. Delp for their valuable comments on the research and dissertation.

I would also like to thank Prof. Shuicheng Yan and Yunchao from the National University of Singapore who have had many valuable discussion with me when I was there and developed some of the work. I am thankful to my EISL colleagues and my friends throughout the years. Thanks Lu for caring and supporting.

My highest gratitude goes to my father Kefei Chen, my mother Ling Zhou, and my brother Yueyang Chen for their love and support. I thank my parents for taking me to a lot places when I was young, for providing me with the best environment growing up. Thank you to my brother, who is the fun guy at home, for bringing so much joy to our life. I know I can always count on my family when times are tough.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

ABSTRACT

Chen Ming PhD, Purdue University, December 2016. Content-based Image Understanding with Applications to Affective Computing and Person Recognition in Natural Settings. Major Professor: Jan P. Allebach.

Understanding the visual content of images is one of the most important topics in computer vision. Many researchers have tried to teach the machine to see and perceive like human. In this dissertation, we develop several new approaches for image understanding with applications to affective computing, and person detection and recognition. Our proposed method applied to fashion photo analysis can understand the aesthetic quality of photos. Further, a bilinear model that takes into account the relative confidence of region proposals and the mutual relationship between multiple labels is developed to boost multi-label classification. It is evaluated both on object recognition and aesthetic attributes learning. We also develop a person detection and recognition system in natural settings that can robustly handle various pose, viewpoints, and lighting conditions. The system is then put into several real scenarios that has different amount of labelled data. Our algorithm that utilizes unlabelled data reduces the effort needed for data annotation while achieving similar results as with labelled data.

1. INTRODUCTION

The volume of visual data being generated is growing exponentially. With that large amount of data, being able to understand their visual content would be of great value. However, image content understanding is intrinsically a challenging problem for two reasons. First, visual data is very rich and ambiguous, thus making it very hard to find a good image representation. Second, some tasks are rather subjective and are even hard for human to reach a consensus. The goal of this dissertation is to investigate and develop algorithms towards extracting better visual features and understanding the content of images with emphasis on affective computing and person recognition.

In Chapter 2, we investigate a sub-problem of affective computing: aesthetic quality evaluation for fashion photos. We design global features, generic features and features that consider the location of the salient object to form the image representation. A dataset that contains 500 images from an online fashion shopping website is constructed. We conduct psychophysical experiments to collect ground truth aesthetic quality evaluation from human subjects. Then, a model is learned to predict the aesthetic quality.

In Chapter 3, we first introduce a general method called Confidence Ordered Proposals (COP) that can boost the performance of multi-label object classification and then apply it to aesthetic attributes learning. Different from single-label image classification, images with multiple labels usually have a large variety and interaction among the different objects or concepts in the images. So we propose to make use of multiple region proposals and learn a model that combines the raw classification result of each proposal. This method takes into account both the relative confidence of region proposals and the mutual relationship between multiple labels. Later, this approach is applied to classify images into different aesthetic categories, which is a more intuitive way to describe the aesthetic quality than simply giving a score as in

Chapter 2. This is formulated as a multi-label classification problem and we apply COP to it .

In Chapter 4, we investigate another sub-problem of affective computing: image emotion classification. This is similar to aesthetic quality in the sense that both of them require very subjective evaluation. We utilize the recent development in deep convolutional neural network(CNN) to learn rich features for emotion classification. A multi-scale pooling method using CNN features is proposed to improve the previous best results.

In Chapter 5, we introduce a pipeline for person detection and recognition in natural settings and test it in several use case with different amount of labeled data. Instead of doing face detection or human body detection, which have been widely explored before, we propose to use head region instead that can not only been reliably detected. We also utilize large scale external datasets to train a Faster R-CNN for head detection and another deep CNN that does head recognition. Experimental results show that our person detection and recognition system achieves promising performance for a challenging TV series dataset. With the semi-supervised learning approach, we can achieve comparable or even better results compared to using the fully labelled dataset with minimum only effort of annotation

Lastly,in Chapter 6, we conclude the dissertation and summarize our contributions.

2. AESTHETIC QUALITY EVALUATION FOR FASHION PHOTOS

2.1 Introduction

Rating images based on their aesthetic quality is a popular topic in computer vision and image understanding in recent years. It has not only attracted the attention of the research community, but has potential to be used in many real applications, especially in those online communities where a large number of photos are being posted and shared.

A particular type of on-line community in which aesthetic of photos is important is that which focuses on fashion items. Since aesthetic appearance is a critical element determining the attractiveness of such items to potential buyers, providing high-quality photos of the items is essential. However, these items are often photographed by the owners of the items. These individuals are generally amateur photographers with no knowledge of the basics of good photography; and the equipment that they use to capture their images are often mobile cameras, for which special care is needed to capture high-quality images. All the above potentials for creating an enjoyable online community highly depend upon the ability to differentiate high-quality photos from low-quality photos, or more specifically, to autonomously assign an aesthetic score to the photos. However, aesthetic inference is not an easy task. There is no technical definition for aesthetic quality, so we can not use a simple expression to describe that.

In this chapter, we use a machine learning-based approach to investigate the problem of aesthetic quality evaluation for photos of fashion products. In Fig. 2.1, we show the pipeline of our system. Following some basic guidelines that professional photographers use, we design a set of features to represent the images. We then con-

duct psychophysical experiments to collect ground truth evaluation of the aesthetic quality specifically for photos of fashion products. The rest of this chapter is organized as follows. In Sec. 2.2, we review some of the prior work. In Sec. 2.3, the features are introduced. In Sec. 2.4, the ground truth data collection procedure is described. The results are shown in Sec. 2.5, followed by our conclusions in Sec. 2.6.

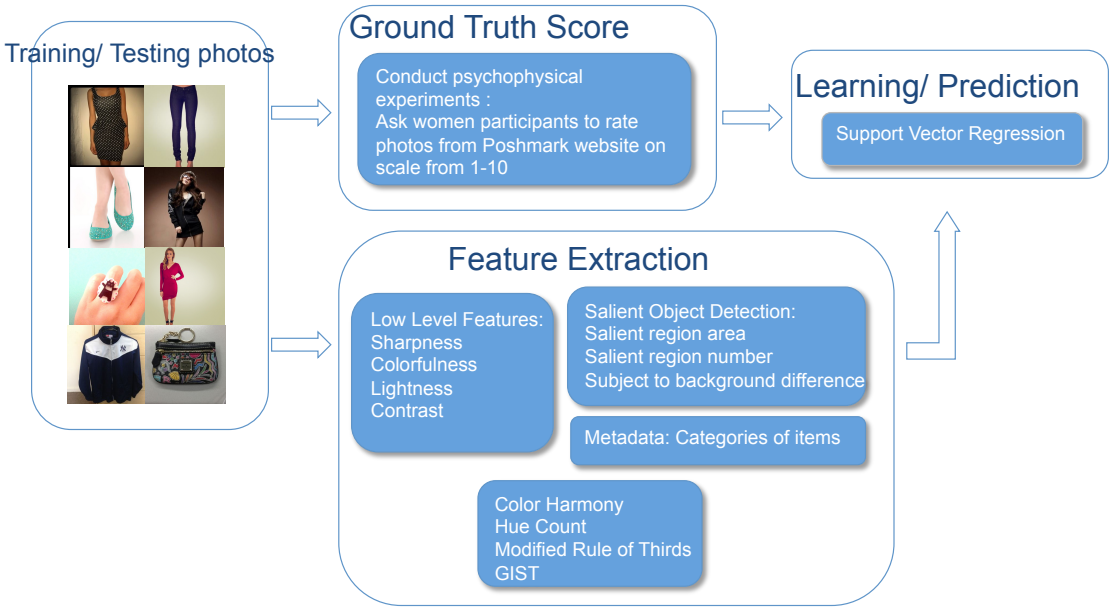


Fig. 2.1. Pipeline of the aesthetic quality evaluation system

2.2 Related Work

There are many previous works on learning the aesthetic quality of photos. Some work follow guidelines from professional photographers and designed features that are relevant to what photographers think good-looking photos should have. Datta et al. [1] is one of the first teams of researchers that formulated a computational approach for aesthetic inference. They used a total of 56 global features to train a support vector machine to classify photos into 'High quality' and 'Low quality'. Ke et al. [2] also designed features based on guidelines for photographers to mimic human aesthetic perception. They did a more in-depth investigation in understanding how high-level semantic features help aesthetic quality evaluation. Luo et al. [3] and Wong et al. [4] incorporated a salient region detection method, derived a number of high-level semantic features based on the subject and background division, and trained a system to predict the aesthetic scores of photos. By incorporating object-level information, they showed that the performance of aesthetic evaluation can be further improved.

Marchesotti et al. [5] departed from using features that are specific to aesthetic inference, and used generic image descriptors such as SIFT, GIST, and bag-of-visual words instead for feature extraction. They showed that although the features that are used are not specifically designed for aesthetic quality evaluation, they achieve very good results. The author also released a large-scale public dataset, AVA, that contains images taken by photographers and rated online for a photographic competition. AVA had since become the benchmark dataset for many works on aesthetic quality evaluation.

These works made no assumptions on the types of photos and achieved reasonable result for general photos. Later works such as Li et al. [6] and Xue et al. [7] focused on images with faces and added features related to face detection to improve the prediction accuracy. Luo et al. [8] divided photos into seven categories based on their content and designed different predictors for different types of photos. With prior

knowledge of the content in the photos, different classifiers are trained accordingly to boost the performance.

2.3 Feature Extraction

It has been shown in previous research that designing relevant features is the key issue for aesthetic inference. In this chapter, we incorporate global and generic features, compositional rules, salient object detection, and metadata together to form the feature vector. The name of each feature below is followed by a reference number F_n, where n is an integer. These reference numbers are used in Table 2.2 at the end of the chapter.

2.3.1 Global Features

- **Sharpness (F_1).** The sharpness of an image is an important measure of how well the photo is taken. For photos of a fashion product that is intended to be sold, whether the textures and details are captured clearly is extremely vital. As most photos for this particular application are taken by mobile phones, they are prone to blurriness due to movement and limited capability of the camera. So measuring the sharpness of an image can be a discriminative feature for aesthetic quality. Calculating a sharpness metric has been coined as the reciprocal of computing the blurriness in Reference [9]. However, this method relies on estimating the blurring kernel, which is computationally expensive. So we use Narvekar et al.'s no-reference sharpness metric [10] to measure the sharpness of a photo.

- **Lightness (F_1).** Under-exposure and over-exposure are two common issues for photos of low aesthetic quality. In order to measure the overall lighting condition of the image, the image is first converted to CIELAB color space.

Then the average value of the image in the L^* channel is taken as the lightness score, given as

$$q_l = \frac{1}{|I|} \sum_{x,y} L^*(x, y), \tag{2.1}$$

where $|I|$ is the size of the image and $L^*(x, y)$ is the L^* value at each pixel. CIELAB is a perceptually uniform color space and the L^* channel approximately matches the human perception of lightness. This makes it ideal for the lightness metric for images.

- **Contrast (F_1).** High-quality photos usually have higher global contrast than those of low-quality. We use a method similar to that described in Reference [8] to measure the contrast of an image as follows. The image is converted into CIELAB color space; and the histogram of L^* is computed. Then, we take the span of the histogram that contains the central 98% of the image pixels as the contrast score.

- **Colorfulness (F_1).** Viewers usually prefer photos that are colorful. Here, we used the method proposed in Reference [11] to measure the overall colorfulness of an image. They first define an opponent color space as

$$rg = R - G, \tag{2.2}$$

$$yb = \frac{1}{2}(R + G) - B, \tag{2.3}$$

and calculate the first and second order statistics in that color space. Then, they set up a psychophysical experiment, and ask people to rate images according to colorfulness. Finally, a metric q_{co} is fit to the ground truth data. q_{co} is given as

$$q_{co} = \sigma_{rgyb} + 0.3 \cdot \mu_{rgyb}, \tag{2.4}$$

where $\sigma_{rgyb} = \sqrt{\sigma_{rg}^2 + \sigma_{yb}^2}$, and $\mu_{rgyb} = \sqrt{\mu_{rg}^2 + \mu_{yb}^2}$. The metric has obtained a correlation of over 90% with the experimental data.

2.3.2 Salient Object Extraction

The clarity of the subject is one of the most important factors that distinguishes high-quality photos from low-quality photos. Professional photographers try hard to focus on the subject, so that it attracts the visual attention of the viewer. For a fashion product that is supposed to be sold, whether it stands out from the background is especially important. Taking the subject into consideration has been proposed before to help perform aesthetic inference in References [3, 4] . Luo et al. [3] assumed that the subject region is clear, while the background is blurred. They extract the subject region by detecting blurriness, and then subtract it from the image. This assumption can only be applied to photos taken by professional cameras, where the depth of field can be very small. For photos taken by mobile phones, as in our case, this is no longer true. Wong et al. [4] assumed that the salient region of the image contains the subject. They compute the saliency map using Itti et al.'s [12] visual saliency model, and use it as a seed for segmentation. The segments that correspond to high value in the saliency map are used as the salient object. However, Wong et al.'s method tends to result in a lot of small salient regions that are spread all over the image. This is because Itti et al.'s visual saliency model is designed to detect the location of the significant stimuli, and does not take into consideration any high-level information related to the object.

We follow Wong et al.'s approach and incorporate the saliency map with the segmentation result to form a salient object mask. The saliency map SM for the image I is computed using the method described in Reference [13] , where the image is represented as a low-rank matrix plus sparse noise. The background is modeled as a low-rank matrix, as it contains redundant information, while the salient region is modeled as sparse noise, as it is quite different from the background region.

We then perform over-segmentation using the mean-shift algorithm [14], and determine whether the segment belongs to the salient object O or the background B.

If the mean saliency value for a segment S_i exceeds a predefined threshold, then that segment is considered to be part of the salient object and vice versa.

$$O = \{(x,y)|(x,y) \in S_i, avg(S_i) > \alpha \cdot avg(SM)\}, \tag{2.5}$$

$$B = \{(x,y)|(x,y) \in S_i, avg(S_i) <= \alpha \cdot avg(SM)\}, \tag{2.6}$$

Empirically, α is chosen to be 1.5. Figure 2.2 shows an example image with its saliency map, segmentation result, and salient object mask.

Using the salient object mask, the following features are defined.

- **Subject-Background Difference (F_2).** The subject-to-background difference indicates whether the subject stands out from the background. The image is converted to HSV color space; and for each channel, the squared difference between the mean value of the subject and the mean value of the background is calculated as

$$q_{sbd,i} = \left(\frac{1}{|O|} \sum_{(x,y) \in O} I_i(x,y) - \frac{1}{|B|} \sum_{(x,y) \in B} I_i(x,y) \right)^2, \tag{2.7}$$

  where I_i denotes a specific channel i of the HSV color space i.e. Hue, Saturation, or Value.

- **Number of Salient Regions (F_2).** The number of salient regions is calculated as the number of distinct connected components in the salient object mask. It is a measure of how complicated the image is. Usually, a simple subject is depicted in a high-quality photo, while there may be multiple subjects that are distractive to the viewers in a low-quality photo.

- **Aggregate Size of All Salient Regions (F_2).** The aggregate size of all salient regions is calculated as the total number of pixels that belong to the salient object. For a well-taken photo, the area that the salient object occupies should be neither too large nor too small.

(a) Original image

(b) Saliency map

(c) Segmentation result

(d) Salient object mask

Fig. 2.2. Illustration of steps for generating salient object mask.

2.3.3 Compositional Rules

- **Color Harmony (F_3).** Harmonic colors are sets of colors that are aesthetically pleasing in terms of visual perception. Cohen et al. [15] applied the Matsuda color harmony model [16] for color image enhancement. Luo et al. [8] incorporated color harmony into their aesthetic quality predictor, and achieved very reliable result. In this work, a harmonic template fitting method [15] is used to measure the color harmony of a photo.

- **Modified Rule of Thirds (F_3).** Placing the subject at a location such that the image looks visually balanced is always a requirement for good photos. A widely adopted principle for that is the rule of thirds. This rule suggests that if we divide the image into nine identical cells by two equally spaced horizontal lines and two equally spaced vertical lines, we should place the center of the main subject at one of the four intersections of two lines. However, most photos of fashion products tend to have the subject horizontally centered in the image, which violates the rule of thirds. So we propose a modified rule of thirds. Instead of dividing the image horizontally into three parts, we divide it into two parts. Then the normalized minimum distance between the centroid of the salient region and each of the two intersections of the dividing lines is calculated as

$$q_{rot} = \min_{i=1,2} \sqrt{\left(\frac{x_c - x_i}{l}\right)^2 + \left(\frac{y_c - y_i}{w}\right)^2}. \tag{2.8}$$

Here rot denotes "rule of thirds", and (x_c, y_c) and (x_i, y_i) denote the coordinates of the centroid of the salient object and the two intersections, respectively. The parameters l and w denote the length and width of the image, respectively.

2.3.4 Generic Features

- **GIST Descriptor (F_4).** The GIST descriptor was originally used for scene categorization [17]. A 128-dimensional feature vector that describes the global structure of a scene is estimated using spectral information and coarse localization. In practice, principal components analysis is applied to the feature vector; and only the first five components are used.

2.3.5 Product-Type-Based Group ID

Utilizing metadata is an effective way to gain prior knowledge of the characteristics of photos, thus improving the prediction accuracy. At the website from which we drew our images, there are 13 types of products that are being sold, i.e. dresses and skirts, sweaters, tops, outwear, jackets and blazers, denim, pants, boots, shoes, accessories, jewelry, handbags, and clutches and wallets. We divide these 13 categories into 5 groups based on the visual similarity of their typical images as shown in Table 2.1. Figure 2.3 shows typical images from each group.

Table 2.1.
Five major groups of fashion merchandise sold at a fashion website

Group 1	dresses and skirts, sweaters, tops, outwear, jackets and blazers
Group 2	denim, pants
Group 3	boots, shoes
Group 4	accessories, jewelry
Group 5	clutches and wallets, handbags

2.4 Data Collection

To train and test the aesthetic score predictor, a database of a large number of manually rated photos is necessary. References [1] and [3] used the photos from

(a) Group 1

(b) Group 2 (c) Group 3

(d) Group 4 (e) Group 5

Fig. 2.3. Typical photos of each of the five groups defined in Table 2.1

online photo sharing websites such as Photo.net and flickr.com, which are already rated by the online community, while in Reference [18] the researchers conducted experiments themselves to collect ground truth aesthetic scores. Although most of the recent research was conducted using photos from the online community, there are two drawbacks to this approach. First, most photos on Photo.net and flickr.com are posted by enthusiast photographers, which means the overall photo quality is biased. Second, most photos on those websites focus on a general set of topics, such as landscapes, portraits, and artistic ideas. Since different types of photos may have different characteristics, an aesthetic score predictor trained on photos of landscapes may not work well for photos of fashion products. For the above reasons, we decide to construct a database of manually rated photos, specifically for photos of fashion products.

We downloaded over two thousand photos from a fashion website and carefully selected 500 of them as training and testing samples. Each of the product-type-based groups defined in Table 2.1 has approximately the same number of photos to ensure that there is little bias between each group. The ground truth aesthetic scores of the 500 images were obtained through psychophysical experiments conducted with 18 human subjects. All the human subjects were women who regularly buy fashion products online. The experiment took an hour for each participant, during which time each participant was asked to rate 150 images with a Matlab graphical user interface (GUI). At the beginning of the experiment, we collect some basic information of the subjects to get an idea of their shopping and photo taking behaviors by asking the following six simple questions:

- Have you ever purchased any fashion products online?
- Do you purchase second-hand clothes?
- How much do you spend on clothing every month?
- What is your favorite website for clothing merchandise?

- How many photos do you take on average per month?
- What do you usually take photos of?

The questions are displayed in the startup page of the Matlab GUI. Figure 2.4 shows the startup page of the experiment. Each subject is asked to rate the photos on a 1 to 10-point scale, where 1 denotes worst quality and 10 denotes best quality. In Fig. 2.5, we show the screenshot of the experiment. Each photo was rated by at least 5 subjects. For each photo, the average value among all subjects who rated that photo is used as the ground truth aesthetic score. In Fig. 2.6, we show the histogram of the average scores. It can been seen that the ground truth aesthetic score is close to a Gaussian distribution, where most photos are rated as medium aesthetic quality fewer photos are rated as either very high or very low quality. A plot of the standard deviation for photos of different average score is shown in Fig. 2.7. From the data, we find that people tend to reach a consensus for photos at the two extremes, while having different opinions for those in the middle.

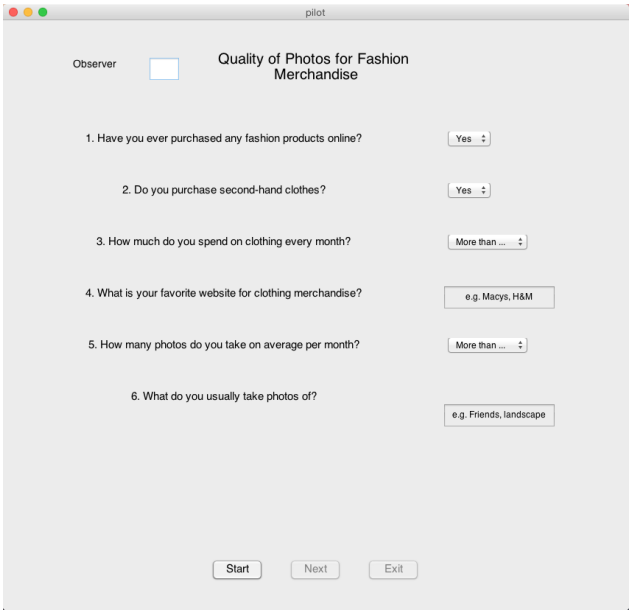


Fig. 2.4. Startup page of the GUI for the data collection experiment

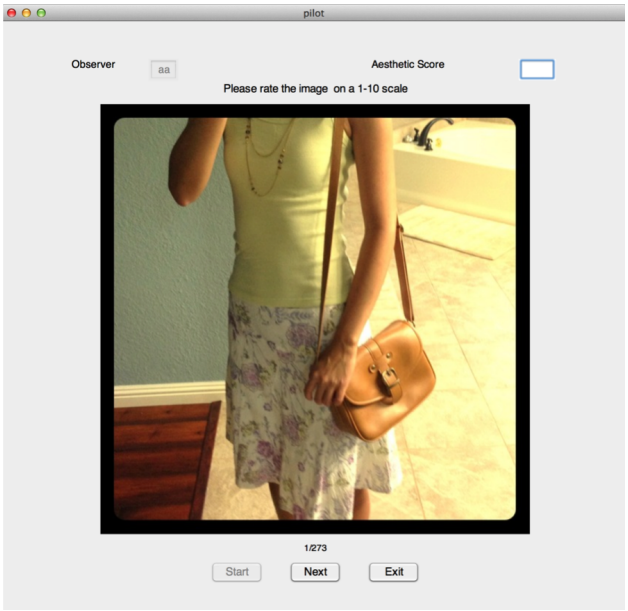


Fig. 2.5. Screenshot of the experiment GUI

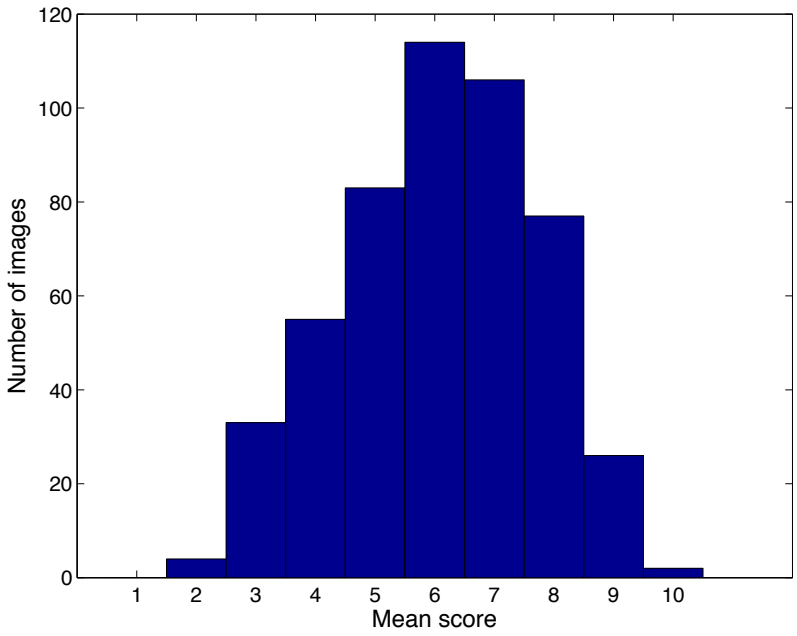


Fig. 2.6. Histogram of the average scores

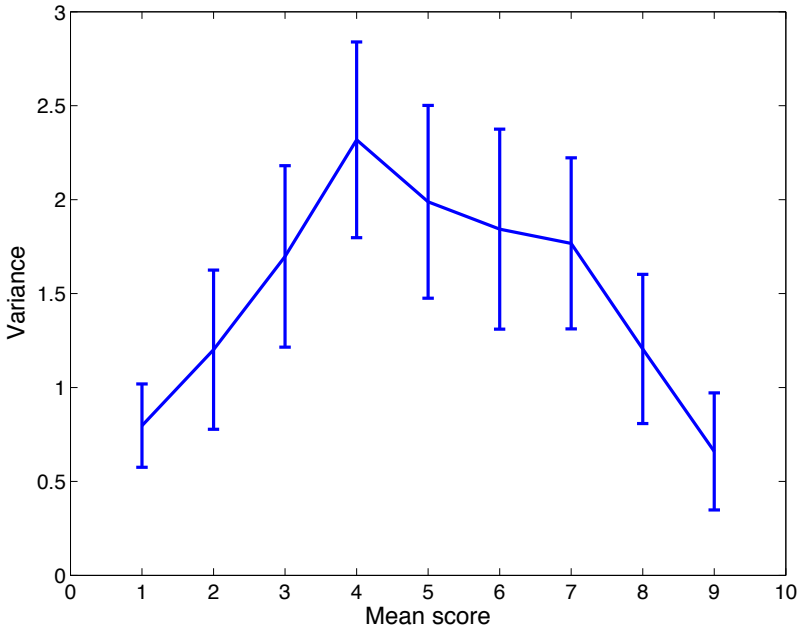


Fig. 2.7. Variance of the scores versus the average scores

2.5 Predicting Aesthetic Quality

We model and evaluate aesthetic quality evaluation in two tasks, as a classification problem (Sec. 2.5.1), and as a regression problem (Sec. 2.5.2).

2.5.1 Classification

We set up the classification problem similar to some previous work **??**, where one tries to discriminate between images of high aesthetic quality and low aesthetic quality. For classification, we define the binary labels

$$y_i = \begin{cases} 1 & \text{if } rank(q_i) \text{ in the top } \delta \\ -1 & \text{if } rank(q_i) \text{ in the bottom } \delta \end{cases} \tag{2.9}$$

where q_i is the average aesthetic scores from the experiment. The classification problem is parameterized by the variable δ. As we change the value of δ, the difficulties

of the classification problem changes. When $\delta = 0.5$, we use all the data available for training and testing. In this case, the classification problem becomes more difficult, as positive and negative iamges are more ambiguous. The classification problem becomes easier as δ gets smaller, but in this case we have access to less data.

We learn the model for aesthetic quality to predict y_i from the image representation x_i using an l_2-regularized with a hinge-loss radial-basis-function (RBF) support vector machine (SVM) classifier. The loss function to be minimized is:

$$\underset{w}{\text{minimize}} \quad \sum_{i=1}^{n} \max\big(1 - y_i w^T \phi(x_i), 0\big) + \frac{\lambda}{2}||w||^2, \tag{2.10}$$

where λ is the regularization paramter, and $\phi(x_i)$ is the radial basis kernal. The label of the sample is give by the folowing equation:

$$y_i = sgn(w^T \phi(x_i)) \tag{2.11}$$

In Fig. 2.8, we show the average accuracy using different feature sets with respect to different value of δ. A five-fold cross-validation is run ten times and the average accuracy is calculated over each cross-validation. The baseline accuracy is 50% as the positive set and negative set have the same number of samples. We compare the results of different feature sets: global features, salient object related features, compositional rules, and generic features respectively. We also test the result of the combined features, where we train a separate classifier for each feature set and the average decision values are used as the final confidence scores. We can see that for all methods, the average accuracy is decreasing as δ increases. From this, we can infer that the features that we extracted are indeed correlated with aesthetic quality as it shows that easier classification problems leads to higher accuracy. However, when $\delta = 0.05$, the accuracy is relatively lower. This is probably due to the fact that when δ is close to zero, the number of samples are very small, which leads to unstable performance. This can also been seen from the large variance of average accuracy at $\delta = 0.05$.

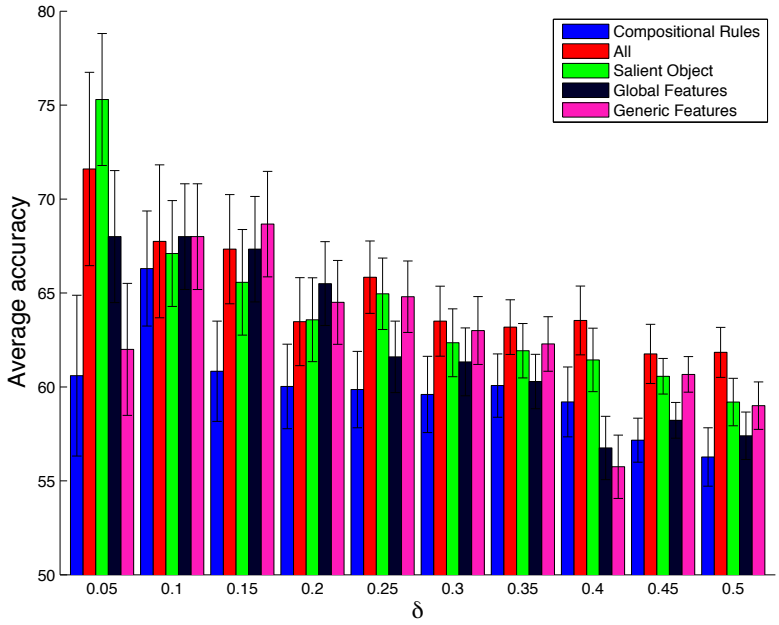


Fig. 2.8. Average accuracy for different feature sets

2.5.2 Regression

We also tried to formulate aesthetic quality evaluation as a regression problem, where we try predict the aesthetic score of a given image. We trained a support vector regression model using the 500 photos that we collected. 5-fold cross validation is carried out to train the model parameters, and to ensure reliability. To evaluate the performance of the predictor, we calculate the pearson correlation coefficient ρ between the ground truth score and the predicted score.

We have tested the system using 4 sets of features. The correlation coefficient values ρ are shown in Table 2.2 with their corresponding set of features. As we increase the number of features, the correlation coefficient decreases. The performance is significantly improved when salient object detection is added to the feature set. This is consistent with our expectation, since the human subjects put a lot of attention on the subject, i.e. the product being sold, when they rated the images. Finally, the product-type-based feature is considered; and an individual predictor is trained for

each of the groups. With the Group ID incorporated in the system, the performance is further improved. The correlation coefficients for each group, using all the features are shown in Table 2.3. Some examples of high and low quality photos as well as their ground truth and predicted aesthetic scores are shown in Fig. 2.9.

Table 2.2.
Correlation coefficients for aesthetic score prediction (scale from 1 to 10) for all groups of photos, using different subsets of the features

	Global Features (F_1)	Salient Object Detection (F_2)	Compositional Rules (F_3)	GIST (F_4)	All combined
ρ	0.52	0.59	0.45	0.55	0.62

Table 2.3.
Correlation coefficients for aesthetic score prediction for each group of photos, using all features

	Group 1	Group 2	Group 3	Group 4	Group 5	Without Group ID
ρ	0.59	0.62	0.61	0.63	0.58	0.62

2.6 Conclusion

In this chapter, we have developed a system that can predict the aesthetic quality for photos of fashion products. Global and generic features, salient object detection, compositional rules, and metadata together are used as the feature vector to represent the images. We have also constructed a database of manually rated photos specifically for photos of fashion products. The testing results show that we can achieve good prediction accuracy using the designed feature sets. We would expect that if a larger amount of ground truth data is collected and more features are added, the prediction accuracy could be further improved.

(a) Y=8.6, $\hat{Y}$=7.9

(b) Y=9, $\hat{Y}$=9.6

(c) Y=1.5, $\hat{Y}$=3.2

(d) Y=4.1, $\hat{Y}$=2.6

Fig. 2.9. Ground truth (Y) and predicted ($\hat{Y}$) aesthetic scores for some example photos.

3. CONFIDENCE ORDERED PROPOSALS: A GENERAL METHOD FOR MULTI-LABEL CLASSIFICATION WITH APPLICATIONS IN AESTHETIC ATTRIBUTES LEARNING

3.1 Introduction

Large-scale object classification has received significant attention in the past few years. With the help of the large amount of training data such as ImageNet [19] and the recent progress of deep convolutional neural network, the state-of-the-art performance of object classification have been improved by a large margin [20–23].

Single-label [19] and multi-label [24] object classification are two sub-problems of object classification. Most existing systems treat single-label object classification the same way as multi-label object classification. They take the whole image as input and output a vector that contains the score of every object class. However, there are two reasons why this approach might not work well for multi-label object classification. First, for multi-label images, the objects in the images are located at different positions with a large variety of poses and orientations. Second, multiple objects in the same image can occlude each other, thus generating different appearances of the objects among images. Some examples of single-label and multi-label images are shown in Fig. 3.1.

Seeing the drawback of using whole images as input, object proposal-based methods are introduced [25] to solve multi-label object classification. This approach starts with generating a number of proposals where the objects are likely to occur. Then for each proposal, a vector that contains the scores of all the labels is assigned. In this way, the problem of multi-label classification is divided into several single-label classification problems, which supposably do not suffer from the drawbacks mentioned

Fig. 3.1. Examples of single-label images from ImageNet and multi-label images from VOC

above. Finally, a decision of prediction is made based on the scores of the proposals. The common way of combining the scores is max-pooling, which takes the maximum score of the proposals as the representative and evaluate the performance based on that.

The idea of proposal-based object classification is closely related to Multiple Instance Learning (MIL), which was a generalization of supervised learning first introduce in [26] in the context of drug activity prediction. For each sample, also called bag, the labels are associated with a set of instances instead of individual instances. MIL has been applied to many visual recognition tasks including image classification [27, 28] and image retrieval [29–31]. In MIL, a basic assumption is that a bag is positive if at least one of the instance is positive. A standard MIL algorithm [32] learns a classifier and use the maximum classification score of all the instances to determine whether the bag is positive or not. The previously introduced proposal-based multi-label object classification can also be considered as MIL. The proposals correspond to the instances and the image that contains all the proposals is the bag.

Recent research in MIL has suggested that max-pooling does not always give the best result. Hu et al. [29] formulate image ranking as MIL and compared using the maximum score of the instances, average scores of the instances, and softmax scores of the instances. They found that taking the softmax scores of the instances outperforms other methods as it can weight more important regions heavier.

Similar to [29], we argue that simply applying max-pooling to the scores of the proposals is not enough to produce the best result. The validity of max-pooling is based on a very strong assumption that both the proposal extraction method and the classifier that assigns scores to the proposals are very reliable. When performing max-pooling, only one out of the many proposals gets to affect the final result, which means the relationship across proposals and their relative importance are not explored. Also, using a single proposal ignores the global context. When several object classes share similar parts, the proposals corresponding to those parts can fire up easily and disregard other parts of the object that are more discriminative. All the above factors also easily lead to miss-classification using max-pooling. Furthermore, the confidence scores are computed on a per class basis, ignoring the correlation and exclusion among object classes.

In this chapter, we propose a method called Confidence-Ordered Proposals (COP) to utilize the ensemble of proposals while alleviating the vulnerability of max-pooling. We aim to learn a bilinear classifier that jointly considers the cross-proposal relationship and cross-label correlation and exclusion. We start by extracting a set of object proposals from the image and feed them into a CNN to generate a preliminary confidence score matrix that contains the C-dimensional score of all the proposals from one image. Then the score matrix is re-arranged according to their order and used as features to train a bilinear classifier for each class. The constraints of the optimization problem are set based on two criteria: high scoring proposals are emphasized with higher weights; classes whose scores are close to the class corresponding to the current bilinear classifier being trained are given lower weight. We will show that the

bilinear classifier trained on COP is very effective and produce improved results on the benchmark datasets VOC 2007 and VOC 2012.

Having proved the effectiveness of COP, we apply it to aesthetic attributes learning and show that the classification accuracy is indeed improved with the help of COP.

The rest of the chapter is organized as follows. In Sec. 5.2, we discuss some related work in object classification. In Sec. 4.3, we describe the details of our proposed Confidence-Ordered Proposals method. The experimental results are presented in Sec. 5.7. The application of the proposed approach to another problem aesthetics attributes learning is presented in Sec. 3.5. Finally, we draw a conclusion in Sec. 5.8.

3.2 Related Work

Before the rise of deep learning, traditional framework for object classification follows the popular feature extraction, feature coding, and feature pooling pipeline. In the first step, hand-crafted features such as Scale Invariant Feature Transform (SIFT) [33], Histogram of Oriented Gradients (HOG) [34] and Local Binary Patterns (LBP) [35] are extracted either around the interest points or densely over the entire image. Then the features are encoded using vector quantization (VQ), locally constrained linear coding (LLC) [36], or the Improved Fisher Vector (IFV) [37]. Finally, feature pooling based on hierarchical matching [38,39] is performed on these encoded features to form the image representation. With the pooled features, a Support Vector Machine (SVM) is trained for the classification. Later work think beyond the traditional pipeline and explore the context information [39–42]. This has proved to be very effective and improved the previous results.

Deep convolutional neural network based systems have shown its promising performance on various visual recognition tasks in recent years. Ranzato et al. [43] applied CNN to handwritten digit recognition. Motivated by the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [19], a lot of work specifically focused on tackling the large-scale object classification. [20–22, 44, 45] are some of these work

that have shown many insightful ideas in designing the network and achieved great results on ImageNet.

The large amount of data of ImageNet is one of the key factors to CNN's success on object classification. As CNN is usually very large and has many parameters, it makes it very hard to train an effective network for small-scale datasets. Razavian et al. [46] proposed a CNN features + SVM pipeline that alleviates the lack of training data for a variety of visual recognition tasks. They showed that the CNN activations of images extracted directly from a pre-trained network can be used as off-the-shelf features for classification. These features are then fed into an SVM for either scene classification, image retrieval, and object classification. Later, [23, 45, 47–50] demonstrated that CNN models that are pre-trained on large datasets such as ImageNet can be fine-tuned with the target datasets that do not have enough training images. This makes it feasible to access a specific CNN for every dataset and task.

Most recently, Oquab et al. [48], Girshick et al. [49] and Wei et al. [25] presented object proposal-based methods for multi-label object classification and detection. This approach has proved to be more effective for multi-label object classification than using full image for classification as it can better target the multiple objects that are located around the entire image. All methods above use the score of a single proposal when making the final classification prediction. Unlike previous methods, we take advantage of the ensemble proposals and explore the relative importance to improve the classification performance.

Bilinear classifiers are proposed to capture the dependence of data on multiple factors. This is particularly useful for visual data that is better represented as a matrix. Tenenbaum et al. [51] separate style from content, such as handwriting words across letters and faces or objects in different viewing consitions using a bilinear model. Pirsiavash et al. [52] applied bilinear SVM to model the spatiotemporal relationship in video sequences and showed state-of-the-art results in people detection and action classification. Tan et al. [53] used sparse low rank bilinear logistic regression to detect face liveness from a single image. In this chapter, the confidence scores of the pro-

posals of every class is formed as a matrix and used as the image representation. We formulate the confidence of proposals and their object class correlation and exclusion in a bilinear model to improve the preliminary classification score.

3.3 Multi-label Image classification with Confidence-ordered Proposals

In this section, we introduce the architecture of our proposed method, Confidence-Ordered Proposals (COP), and how to apply it to multi-label object classification. We first apply a proposal extraction method Edge Box [54] that is both computationally efficient and accurate to generate a number of object proposals for each image. Then we feed the proposals into a CNN to generate the preliminary classification scores. The CNN can be any model that outputs a classification score for each label on the target dataset. After that, the scores of the proposals of each image is re-arranged and a bilinear model using COP as features is learned for each class, aiming to account for the cross-proposal and cross-label relationship. With the COP model, the classification scores are refined to produce improved results.

3.3.1 Proposal Extraction

The first step of the pipeline is to generate a number of object proposals. For better efficiency and accuracy of the overall system, we require the proposal extraction method to be both computationally efficient and have high object detection recall rate. Many methods [54–59]have emerged for object proposal extraction in recent years. Objectness based methods [55–57] rank proposals by learning a classifier and assigning an objectness score to each proposal. Among these methods, Cheng et al. [57] proposed a simple yet powerful feature called binarized normed gradients (BING) that achieve very high speed. However, BING only generates very loosely fitting proposals and does not perform very well at high Intersection over Union (IoU).

Another paradigm of proposal extraction methods is based on superpixel merging [58,59]. Selective Search [58] computes hierarchical segmentations based on superpixel

and place bounding boxes around the segments. Because of its high recall rate, Selective Search was used in the top detection system [49]. However, since it needs to generate superpixels, the computational time is low.

Zitnick et al. [54] proposed Edge Box which is an object proposal extraction method based on edges. Edge Box ranks the proposals by measuring the number of edges that exist in the box minus those that are members of contours that overlap the box's boundary. Their experimental results show that Edge Box generates object proposals at comparable speed with BING while having recall rate as high as Selective Search. Due to its computational efficiency and high accuracy, we decide to use Edge Box to extract object proposals.

3.3.2 Model Formulation

Our proposed method Confidence-Ordered Proposals aims to learn a bilinear classifier that jointly accounts for the cross-proposal and cross-label relationship for every object class. Each proposal is assigned a preliminary confidence score using an object classification method first. Then the relative importance of high scoring proposals and the correlation and exclusion between labels are explored when we re-arrange the preliminary scores. In principle, any method that outputs a confidence score for a proposal can be used. Here, our work is based on CNN, of which the output of the last fully-connected layer are the confidence scores.

For each image, K proposals are extracted and their preliminary confidence scores are computed using the CNN. Let's define a matrix $P \in \mathbb{R}^{K \times C}$ for a particular image, where K denotes the number of proposals extracted from the image and C denotes the number of classes. The ith row denoted by $P_{i,:}$ is the confidence scores of a particular proposal for all object classes. The jth column denoted by $P_{:,j}$ is the confidence score for all the proposals of a particular class.

One bilinear classifier needs to be trained for each object class, C classifiers in total. Consider a particular object class l for now. Given an image and its cor-

responding score matrix P that contains the scores of the proposals, we form the re-arranged score matrix M following the procedures below:

1. For every column j of P, it is sorted separately in descending order and assigned to the jth column of a temporary matrix M' such that

$$M'_{1,j} \geq M'_{2,j} \geq \ldots \geq M'_{K,j}$$

2. Let $M = M'$. Swap the first column of M, $M_{:,1}$ with the lth column of M, $M_{:,l}$. Then sort each row of M independently.

Then we learn a bilinear model according to the following cost functions and constraints using the re-arranged matrix M as features:

$$\begin{aligned} \underset{x_1,x_2,b}{\text{minimize}} \quad & \sum_{i=1}^{n} \max\big(1 - y_i(x_1^T M^i x_2 + b), 0\big) \\ \text{subject to} \quad & x_{1,1} \geq x_{1,2} \geq \ldots \geq x_{1,K} \geq 0, \text{and} \\ & x_{2,2} \leq \ldots \leq x_{2,C} \leq 0, \end{aligned} \tag{3.1}$$

where x_1, x_2 and b are the bilinear model parameters for the current class we are training for, M^i is the re-arranged score matrix, and y_i is the class label that denotes whether the image belongs to the current class.

Since the columns of the matrix M denote the sorted scores of all the proposals, x_1 aims to model the relationship across proposals. By enforcing positive values and descending order on the weight, a larger weight is assigned to the proposals with a higher score. This is based on the assumption that proposals with higher scores are more likely to contain the object, thus, are of more importance. Similarly, since the rows of the matrix M denote the re-ranked scores among classes, x_2 aims to model the relationship across different classes. Recall that except for the elements in the first column of M, which is re-arranged so that it corresponds to the current class we are training for, all the other rows are sorted in descending order. By enforcing negative values and ascending order on x_2, a more negative weight is assigned to the classes with higher scores other than the current class, so that the more confusing

classes are penalized. In this way, the classifier increases its ability to discriminate among other classes.

For the optimization problem, we try to solve for the cross-proposal weight x_1, the cross-label weight x_2, and the bias term b that yield the minimum summation of hinge loss. Notice that the constraints are linear, but the loss function is not convex with respect to x_1 and x_2. The non-convexity makes the problem hard to solve. However, if we fix one of the variables and solve for the other variable and the bias term b, the problem becomes a linear programming. In practice, we fix x_2 and solve for x_1 and b first. Then, we take the optimal x_2 from the previous step and solve for x_1 and b. Here, we refer to the procedure of optimizing x_1, x_2 and b once as one iteration. Usually, it takes five iterations for the variables to converge. With the optimal parameter x_1, x_2, and b, the bilinear classifier is defined as

$$f_\beta(M) = x_1^T M x_2 + b, \tag{3.2}$$

where the model parameter is denoted by $\beta = \{x_1, x_2, b\}$. For the rest of the C object classes, we follow the same procedure and learn a bilinear classifier.

3.3.3 Hard Sample Mining

In the previous subsection, we introduce the formulation of confidence-ordered proposals and how to learn the parameters in general. However, for large-scale visual recognition problem such as object classification, the number of training samples are in the order of 10,000. As a result, the number of constraints and number of variables of the linear programming problem are also very huge, which can lead to inaccurate solutions or can even be impractical to optimize using standard linear programming solvers. In order for the bilinear model to be feasible on larger datasets, we adopt a hard samples mining approach that was used in [60].

Hard sample mining is motivated by the bootstrapping idea, where an initial subset of samples are used to train a model, and then the samples that are miss-classified or very close to the decision boundary are collected to update the model.

For an optimization problem that uses hinge loss as in our case, only the samples that are near the decision boundary affect the total loss. Samples that are far away from the decision boundary yields zero loss, thus having no affect on the classifier. As a result, we only need to consider those samples that are within a margin, namely the hard samples, and discard samples that are outside the margin, namely the easy samples, during the training phase. By solving a sequence of training problems using a small number of hard samples, we reduce the complexity of the linear programming in each training iteration. Since our motivation of applying hard sample mining is to reduce the total number of samples, we also perform hard positive mining in addition to hard negative mining. We modify the algorithm in [60] and perform hard sample mining for the positive samples until the number of positive sampels is less than a pre-set threshold. This effectively The detailed procedure of hard sample mining is described below:

As in [60], we define hard and easy samples of a training set D relative to the model parameter β as follows,

$$H(\beta, D) = \{\langle M, y\rangle \in D | y f_\beta(M) < 1\}. \tag{3.3}$$

$$E(\beta, D) = \{\langle M, y\rangle \in D | y f_\beta(M) > 1\}. \tag{3.4}$$

$H(\beta, D)$ represents both positive and negative samples in D that are miss-classified or inside the margin of the classifier defined by β, while $E(\beta, D)$ represents both positive and negative samples in D that are correctly classified or outside the margin.

We start with an initial cache of samples $C_1 \subseteq D$. The cache contains at most N_P positive samples and N_N negative samples. The rest of the data, which is a pool of samples not being used for training, is denoted by $S_1 = D \setminus C_1$. The hard mining algorithms iteratively solve Eq. 3.1 and updates the cache as follows:

1. Train a model β_t using the samples in the current cache C_t. The parameters of the bilinear classifier $\beta_t = \{x_1, x_2, b\}$ are solved iteratively until convergence as described in Sec. 3.3.2.

2. If $H(\beta_t, D) \subseteq C_t$, all the hard samples are from the current cache C_t and there are no hard samples in S_t. In this case, we can not update the cache anymore and should stop and return the current classifier β_t.

3. For positive samples, if there are fewer samples than a pre-set threshold N_{min}, do nothing. Otherwise, for both positive and negative samples, remove the easy samples from the current cache.

4. Add hard samples from the pool, $H(\beta_t, S_t)$, to the cache C_{t+1} for the next round of training. After adding hard samples, the number of positive samples and negative samples in C_{t+1} should not exceed N_P and N_N respectively.

5. Go back to step 1.

3.4 Experimental Results

In this section, we show the image classification results achieved by our proposed method and compare it with several state-of-the-art approaches.

3.4.1 Datasets and Configurations

We evaluate our proposed approach on the well-known public datasets, PASCAL Visual Object Classes Challenge (VOC) [24], which are widely used as the benchmark to assess algorithms for object classification and provide a standardized evaluation platform. These datasets are very challenging as it contains objects that vary significantly in size, position, orientation and pose among images. In this chapter, PASCAL VOC 2007 and VOC 2012 are used for the experiments. These two datasets consist of 9,963 and 22,531 images respectively. Both of the datasets are divided into three subsets, "train", "val" and "test", i.e. 25% for training, 25% for validation and 50% for testing. In our experiments, we combine the "train" and "val" subset as training images, referred to as "trainval" and take the "test" set as testing images (trainval/test: 5,011/4,952 for VOC 2007 and trainval/test: 11,540/10,991 for VOC 2012).

To measure the performance of our approach, *Average Precision* (AP) for each class is employed as the evaluation metric, complying with the standard protocol of PASCAL VOC.

Our CNN implementation is based on the open source library Caffe toolbox [61]. In all the experiments, we used one Nvidia GTX Titan GPU with 6GB memory for CNN training and testing. Network-in-Network developed by Lin et al. [21] is used as the default network pre-trained on ImageNet. When fine-tuning the pre-trained network, the

3.4.2 Proposal-based Multi-label Object Classification

In [25] and [48], proposal-based multi-label object classification are investigated. The general procedure is to extract a number of object proposals first. The proposals are then fed into a shared CNN from which each proposal is assigned a confidence score for each object class. After that, max-pooling is performed across all the proposals for each image to get the final confidence score for every label.

In this work, we first fine-tune the Network In Network (NIN) [21] pre-trained on ImageNet with our target dataset, i.e. VOC 2007 and VOC 2012. Then we extract 500 object proposals from each image. For the proposal extraction method, we apply Edge Box [54] because of its computational efficiency and high accuracy. We use the default configuration of $\alpha = 0.65$ and $\beta = 0.75$ as described in the chapter. In order to capture context information, we enlarge the proposal by extending the short side in both directions such that the short side is as long as the long side of the original proposal. Each proposal is fed into the fine-tuned CNN and the preliminary classification scores from the last fully-connected layer are computed. During the training phase, we take the preliminary scores and train the COP models for each class. During the testing phase, we apply the COP models to the preliminary scores of the proposals and calculate the refined confidence score.

Comparison with max-pooling

Table 3.1 and Tab. 3.2 show the detailed comparison between simply applying max-pooling and applying COP to the preliminary scores of the proposals on VOC 2007 and VOC 2012. The results reported here is based on the NIN ImageNet model fined tuned with the target datasets. We extracted 500 object proposals using Edge Box as it gives the best result. Our choice of the number of proposals is justified in Sec. 3.4.2. From the experimental results, we can see that by replacing max-pooling with COP, the performance has been boosted. The mAP is improved by 3.5% on VOC 2007 and 2.3% on VOC 2012. On both datasets, COP outperforms max-pooling for every object class.

Optimal number of proposals

One of the drawbacks of proposal-based object classification is its complexity. It requires CNN feature extraction for a large number of object proposals. In order to take full advantage of proposal-based method, while not introducing too much computational burden, we evaluate the performance of COP using different number of proposals and compare them with max-pooling in Fig. 3.4.2. We can see that, for both max-pooling and COP, the performance increases as the number of proposal increases up to a certain point and then start to decrease. For max-pooling, mAP starts to drop at 300 proposals, while for COP it starts to drop at 500 proposals. As reported in [54], it requires 800 proposals to achieve 75% recall rate at Intersection of Union (IoU) threshold of 0.7 on VOC 2007 using Edge Box. So we can expect that the best performance is achieved with a relatively large number of proposals. We decide to use 500 proposals in all the remaining experiments as it achieves the optimal result. Once mAP reaches the highest point, it starts to decrease. This can be explained by the fact that as the number of proposal gets very large, many noisy proposals are generated. These noisy proposals do not contain the object but could have a high confidence score by accident. Since max-pooling simply takes the maximum

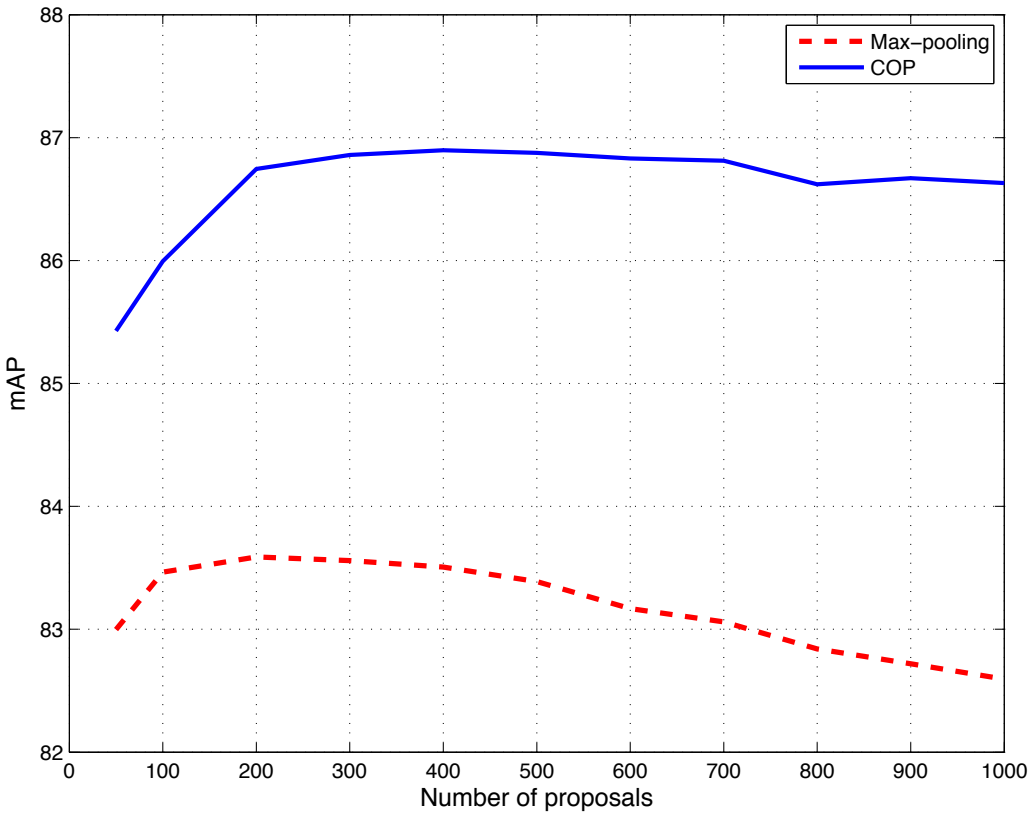


Fig. 3.2. Comparison of mAP versus different number of proposals between COP and max-pooling on VOC 2007

confidence score among all proposals, it is more vulnerable to noisy proposals. The mAP using max-pooling at 1000 proposals is 1% lower than the maximum mAP. Our method, on the other hand, takes advantage of all the proposals and learn a weight to adjust their relative importance. The mAP using COP at 1000 proposals is only 0.3% lower than the maximum mAP. The lower drop at 1000 proposals shows that COP is more robust to noisy proposals.

3.4.3 Combining COP with HCP

As stated in the previous subsection, COP can boost the performance of proposal-based object classification. To see the limit of COP, we apply COP to the best proposal-based object classification method Hypotheses-CNN-Pooling (HCP) [25]. We follow the same procedure described in [25] to generate the preliminary classification score of every proposal.

The training phase of HCP starts with fine-tuning a pre-trained ImageNet model with the target dataset. In this step,a C-dimensional classification score, where C is the number of classes of the target dataset, is generated. This step is referred to as image fine-tuning (IFT) as whole images are used for fine-tuning. Then 500 object proposals (referred to as hypotheses in [25]) are extracted. Instead of using BING [57] for proposal extraction, we use Edge Box [54]. The proposals are clustered into ten clusters based on the normalized cut algorithm [62] and only the highest scoring proposals from the each of the ten clusters are selected as the representative and used for the training stage. The ten representative proposals are then used for another round of fine-tuning called hypotheses fine-tuning (HFT). The output of the HFT model is also a C-dimensional vector that represents the classification scores of every class of the target dataset. The hypotheses fine-tuned model is more data-specific to the hypotheses and is shown to perform better than the image fine-tuned model in [25].

We then apply this HFT model to generate a preliminary classification score for each proposal. The output of the last fully-connected layer of the HFT model are rearranged as described in Sec. 4.3 and used for the training and testing of our COP model. The images that have the highest and lowest five scores are shown in Fig. 3.3 and Fig. 3.4. In the next section, we will show the detailed classification result of COP-HCP and compare it with the state-of-the-art.

Fig. 3.3. The images from the first ten classes that have the highest (green) and lowest scores (red)

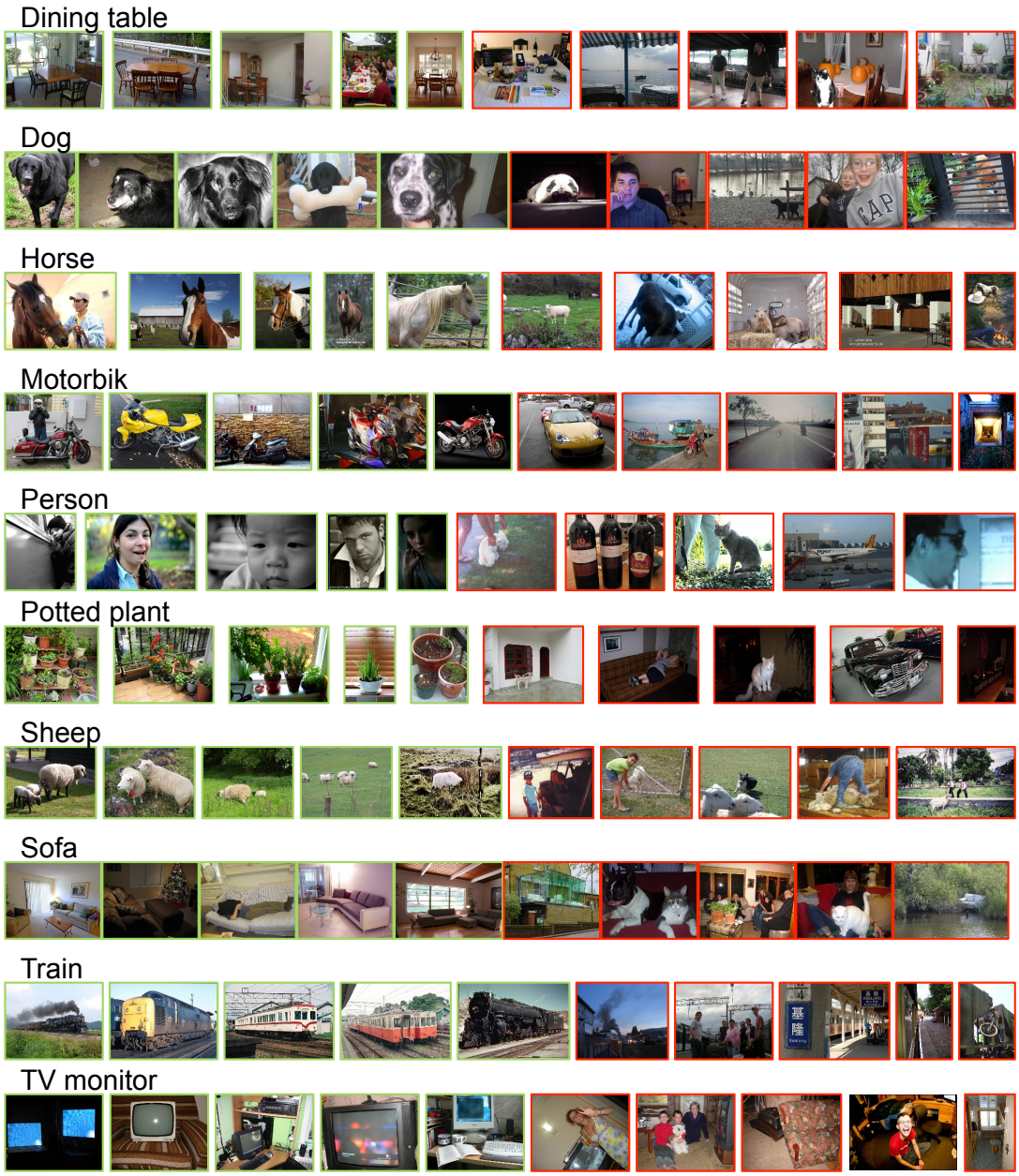


Fig. 3.4. The images from the second ten classes that have the highest (green) and lowest scores (red)

3.4.4 Comparison with State-of-the-art Performance

Image classification on VOC 2007. In Tab. 3.3, we compare the experimental results between several existing solutions with our proposed method on VOC 2007. The results in the upper part [36, 37, 40, 41, 63] of Tab. 3.3 are based on hand-crafted features and are trained without any extra data. The results in the middle part [25, 44–46, 64] are based on CNN, which all utilize additional training images from ImageNet during the pre-training process. In the lower part of the table, we show the results of HCP boosted with COP. We tested our approach with HCP based on different pre-trained ImageNet models, e.g. NIN [21], NIN2000 (NIN trained with an additional 1000 ImageNet classes), and VGG [45].

From the experimental results, we can see that methods that are based on CNN and trained with extra data consistently perform better than methods based on hand-crafted features. The method introduced by Razavian et al. [46] simply applies the features extracted by a CNN pre-trained on ImageNet and trained a linear SVM for classification. Since it directly use a pre-trained CNN and do not fine-tune the network, it is considered as the baseline of CNN-based method. The mAP achieved by the baseline method is 73.9%, which improves the best methods that are based on hand-crafted feature by 2.6%.

The current state-of-the-art result is reported by Wei et al. in [25] (HCP-VGG). They make use of the publicly available 16-layer VGG model [45] as their pre-trained model and applied their Hypotheses-CNN-Pooling pipeline using 500 object proposals for testing. As can been seen from the table, if we apply Confidence-Ordered Proposals on top of HCP-VGG (COP-HCP-VGG), we surpass the current state-of-the-art by 1.9% and achieve mAP of 92.0%. COP-HCP-VGG outperforms other methods, including previous methods and variants of our methods in 15 out of 20 object classes. Significant improvement is achieved on some of the poorly performing class, i.e. “chair”, “table”, and “sofa” compared with previous methods.

In the lower part of the table, we report the COP-HCP performance based on different pre-trained models. We should note that with a better model, mAP can be improved by up to 2.6%.

Image classification on VOC 2012. In Tab. 3.4, we compare the experimental results between several existing solutions with our proposed method on VOC 2012. Among all existing methods shown in Tab. 3.4, only NUS [65] is based on hand-crafted features. The other methods shown in the middle part [25, 45, 48, 64, 66] are all based on CNN that use ImageNet as extra data when pre-training the model. The results that we report are given by building COP on top of HCP. As with VOC 2007, we test our approach with HCP based on different pre-trained models, e.g. NIN, NIN2000, and VGG.

Our best result is achieved by COP-HCP-VGG, which builds COP on top of HCP pre-trained on VGG. It surpasses the current state-of-the-art classification result of 90.1% reported in [25] by 1%. Among all 20 object classes, our method outperforms others in 15 of them. We can also see from Tab. 3.4 that the mAP on VOC 2012 is boosted from 87.9% to 91.1% as we replace NIN with VGG, implying that with a better pre-trained model, the classification results can be further improved.

3.5 Aesthetic Attributes Learning

In Chapter 2, we investigate the problem of aesthetic quality evaluation. We showed that our approach is able to differentiate among high-quality and low-quality images. Despite the promising result, when looking at the prediction result, one may always argue that he/she does not like the image aesthetically. This is natural as aesthetic quality is a very subjective measure. Motivated by this drawback, we try to learn a mid-level features that describe the aesthetic attributes of images. The attributes are well defined such that a person can answer a yes/no question to tell whether the image has this attribute or not. The following list shows some example attributes that describe the aesthetic quality of images.

- high depth-of-field, low depth-of-field, over-exposure, under-exposure, vibrant color, good lighting, rule-of-thirds, big contrast, out-of-focus, blurry, sharp, good composition, nice perspective

The problem of assigning aesthetic attribues to an image can be formulated as a multi-label image classification problem, as one image may have several different characteristics. The most common way for multi-label classification is to train a separate binary classifier for each class and evaluate the performance on for each class. We argue that high-level concept such as aesthetic quality may be determined by a combination of information throughout the entire image, rather than a single patch in the image. So we propose to extract features from local patches throughout the whole image. Then it is very natural to apply COP to the problem of attributes learning.

3.5.1 Image Representation

For aesthetic attributes, some attributes such 'blurry', 'high depth-of-field' are encoded in the edge distribution, while other attributes such as 'vibrant color' and 'over exposure' are embedded in the color information. So we need to consider both structure and color of images when designing the image representation. As in [37], we propose to use SIFT for structural feature and color statistics for color feature. The image descriptors are densely computed for every patch followed by spatial pooling using Fisher Vector.

3.5.2 Applying COP to Aesthetic Attributes Learning

Before training a COP model, we need a base classifier that gives a preliminary classification result. We use SIFT + FV combined with colors statistics + FV [5] as the image representation. Both SIFT and color statistics are extracted from the entire image. We then train a binary linear SVM for every attribute using the combined features.

In the next step, instead of generating a set of region proposals as in the original COP. We simply extract overlapping windows throughout the entire image. This is because the aesthetic attributes are not necessarily related to object proposals as in the case of object classification, but are rather likely to be encoded in the information throughout the entire image.

3.5.3 Dataset

The dataset we use for aesthetic attributes learning is called AVA [67]. It contains mostly artistic photos taken by photography enthusiasts. Following Marchesotti et al. [68], we pick the top five beautiful attributes and top five ugly attributes that are learned from the comments on the photos. The 10 attributes for classification are then:

- **Beautiful attributes**: nice colors, beautiful scene, nice perspective, big congrats, so cute
- **Ugly attributes**: too small, distracting background, snap shot, snap shot, bad focus

3.5.4 Experimental Results

For the base classifier, we train a binary SVM for each aesthetic attribute using SIFT + FV combined with color statistics. Dense SIFT and color statistics are computed for local patches of size 32×32 regularly every 8 pixels. To apply confidence ordered proposal to this problem, we predict the preliminary score for all 10 attributes for all 64×64 windows that overlap by 16 pixels through the entire image.

In Table 3.5, we show the Area under Curve (AUC) for both the base classifier and the enhaced classifier with COP. It can be seen that COP is able to learn the relative confidence of different regions and the mutual relationship between different

aesthetic attribute classes. As is the case for object classification, applying COP helps improve over the preliminary classification results for whole images.

3.6 Conclusion

In this chapter, we introduced a general method to improve the performance of proposal-based multi-label object classification. The raw confidence score of each proposal is rearranged to emphasis the relative importance of proposals with higher scores. By learning the cross-proposal and cross-label relationship from the confidence-ordered proposals, . This method can be applied to any proposal-based object classification framework. From the experimental results on the two benchmark datasets VOC 2007 and VOC 2012, we proved that our proposed method consistently outperforms existing proposal-based method that simply uses max-pooling. By utilizing the best proposal-based multi-label object classification framework Hypothesis-CNN-Pooling, our method achieves the state-of-the-art classification results on both datasets. We also gave a very brief introduction of out current project, where we try to learn a set of aesthetic attributes for a more objective way of aesthetic quality evaluation.

Table 3.1.
Comparison between max-pooling and COP on VOC 2007. The pre-trained model used here is NIN. 500 proposals are extracted for both methods.

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbik	pers	plant	sheep	sofa	train	tv	mAP
Max-pooling	96.8	89.6	95.1	90.7	51.5	79.2	92.7	92.9	64	84	72.6	89.2	86.6	89.9	94.4	68.6	91.2	65.8	95.5	77.4	83.4
COP	97.3	92.9	95.9	92.4	56.9	84.1	94	94	71.3	85.6	80.3	93.8	93	92.2	96.9	70.5	92.8	75	96.2	82.4	86.9

Table 3.2.
Comparison between max-pooling and COP on VOC 2012. The pre-trained model used here is NIN. 500 proposals are extracted for both methods.

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbik	pers	plant	sheep	sofa	train	tv	mAP
Max-pooling	96.4	89.5	93.1	90.2	51.2	87.3	82	93	68.5	80.9	68.3	89.9	81.9	88.2	92.8	68.9	85.4	65.4	93.6	78	82.3
COP	97.1	92.1	94.2	92.9	56.7	89	92.5	94.6	72.6	84.6	74.8	94.7	88.5	89.4	96.9	70.6	88.1	72.2	94.2	80.5	85.8

Table 3.3.

Classification results (AP in %) on VOC 2007. The upper part of the table shows the result using hand-crafted features. The middle part shows the recent results of methods that are based on CNN features. The lower part shows the result of our method built on top of the best proposal-based multi-label object classification pipeline. For our methods, NIN is used as the pre-trained model. 500 proposals are extracted using Edge Box.

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbik	pers	planet	sheep	sofa	train	tv	mAP
FK	75.7	64.8	52.8	70.6	30	64.1	77.5	55.5	55.6	41.8	56.3	41.7	76.3	64.4	82.7	28.3	39.7	56.6	79.7	51.5	58.3
LLC	74.8	65.2	50.7	70.9	28.7	68.8	78.5	61.7	54.3	48.6	51.8	44.1	76.6	66.9	83.5	30.8	44.6	53.4	78.2	53.5	59.3
INRIA	77.2	69.3	56.2	66.6	45.5	68.1	83.4	53.6	58.3	51.1	62.2	45.2	78.4	69.7	86.1	52.4	54.4	54.3	75.8	62.1	63.5
AGS	82.2	83	58.4	76.1	56.4	77.5	88.8	69.1	62.2	61.8	64.2	51.3	85.4	80.2	91.1	48.1	61.7	67.7	86.3	70.9	71.1
AMM	84.5	81.5	65	71.4	52.2	76.2	87.2	68.5	63.8	55.8	65.8	55.6	84.8	77	91.1	55.2	60	69.7	83.6	77	71.3
Razavian	88.5	81	83.5	82	42	72.5	85.3	81.6	59.9	58.5	66.5	77.8	81.8	78.8	90.2	54.8	71.1	62.6	87.4	71.8	73.9
Chatfield	95.3	90.4	92.5	89.6	54.4	81.9	91.5	91.9	64.1	76.3	74.9	89.7	92.2	86.9	95.2	60.7	82.9	68	95.5	74.4	82.4
SPP	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	82.4
VGG-16	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	89.3
VGG-19	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	89.3
VGG-16-19	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	89.7
HCP-NIN	98	93.4	96.8	94.3	66.3	86.9	94.2	96.4	65.5	86.7	74.1	94.2	93.7	90.6	95.3	73.8	93.3	66.8	96.9	82.8	87
COP-HCP-NIN	98.5	95.2	96.9	94.4	69.2	88	95.4	96.5	73.1	88.3	82	94.8	95.5	92.4	97.7	76.5	94.1	76.1	96.9	87.5	89.4
HCP-NIN2K	98.4	96	96.7	94.6	67.4	91	95.1	96.1	70.7	95.4	77.9	95.1	97.4	90.2	95.3	75.3	**96.2**	75.2	97.6	84.6	89.3
COP-HCP-NIN2K	98.1	96.5	96.7	95.2	69.5	91.6	96	96.4	77.4	**96.4**	85	93.5	**98**	92.3	97.7	77	95.9	**83.1**	97.5	87.9	91.1
HCP-VGG	**98.9**	96.8	97.4	96.1	72.4	93.1	95.4	97.2	69	90.6	78.9	96.6	96.6	93.5	96.1	79.2	94.3	74.3	**98.1**	88.8	90.1
COP-HCP-VGG	98.8	**97.2**	**97.5**	**96.6**	**75.5**	**93.5**	**96.4**	**97.5**	**77.5**	90.8	**86.3**	**97.3**	96.4	**94.3**	**98.3**	**81.3**	93.5	82.1	**98.1**	**91.9**	**92**

Table 3.4.

Classification results (AP in %) on VOC 2012. The upper part of the table shows the result using hand-crafted features. The middle part shows the recent results of methods that are based on CNN features. The lower part shows the result of our method built on top of the best proposal-based multi-label object classification pipeline. For our methods, NIN is used as the pre-trained model. 500 proposals are extracted using Edge Box.

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbik	pers	plant	sheep	sofa	train	tv	mAP
NUS	97.3	84.2	80.8	85.3	60.8	89.9	86.8	89.3	75.4	77.8	75.1	83	87.5	90.1	95	57.8	79.2	73.4	94.5	80.7	82.2
INRIA-1000	93.5	78.4	87.7	80.9	57.3	85	81.6	89.4	66.9	73.8	62	89.5	83.2	87.6	95.8	61.4	79	54.3	88	78.3	78.7
INRIA-1512	94.6	82.9	88.2	84.1	60.3	89	84.4	90.7	72.1	86.8	69	92.1	93.4	88.6	96.1	64.3	86.6	62.3	91.1	79.8	82.8
Oquab	96.7	88.8	92	87.4	64.7	91.1	87.4	94.4	74.9	89.2	76.3	93.7	95.2	91.1	97.6	66.2	91.2	70	94.5	83.7	86.3
Charfield	96.8	82.5	91.5	88.1	62.1	88.3	81.9	94.8	70.3	80.2	76.2	92.9	90.3	89.3	95.2	57.4	83.6	66.4	93.5	81.9	83.2
Zeiler	96	77.1	88.4	85.5	55.8	85.8	78.6	91.2	65	74.4	67.7	87.8	86	85.1	90.9	52.2	83.6	61.1	91.8	76.1	79
VGG-16	99	88.8	95.9	93.8	73.1	92.1	85.1	97.8	79.5	91.1	83.3	97.2	96.3	94.5	96.9	63.1	93.4	75	97.1	87.1	89
VGG-19	99.1	88.7	95.7	93.9	73.1	92.1	84.8	97.7	79.1	90.7	83.2	97.3	96.2	94.3	96.9	63.4	93.2	74.6	97.3	87.9	89
VGG-16-19	99.1	89.1	96	94.1	74.1	92.2	85.3	97.9	79.9	92	83.7	97.5	96.5	94.7	97.1	63.7	93.6	75.2	97.4	87.8	89.3
HCP-NIN	98.4	89.5	96.2	91.7	72.5	91.1	87.2	97.1	73	89.5	75.1	96.3	93	90.5	94.8	66.5	90.3	65.8	95.6	82	86.8
COP-HCP-NIN	98.4	89.6	95.7	92.1	74.5	91.4	88.4	97.2	75.5	90.6	77.4	96.7	93.2	91.1	97.2	69.9	91.3	67.6	95.4	85.7	87.9
HCP-NIN2K	98.5	91.4	96.2	93.2	72.5	92.6	88.9	97.4	77	95.9	79.3	96.8	97.5	92.9	95.4	67.8	94.7	70	96.8	83	88.9
COP-HCP-NIN2K	98.5	91.3	95.6	93.6	74.3	93.3	90.6	97.5	79.3	**96.9**	81.2	97.1	**97.9**	93	97.5	70.7	**94.8**	72.4	**96.9**	86.1	89.9
HCP-VGG	**99.2**	92.3	97.5	94.1	79.7	92.8	89.9	98.2	79.1	94.3	79.6	97.8	96.8	94	96.6	71.7	93.3	70.2	**96.9**	88.3	90.1
COP-HCP-VGG	**99.2**	**92.4**	**97.6**	**95**	**81.2**	**93.6**	**91.7**	**98.5**	**81.1**	94.8	**82**	**98.1**	96.9	**95**	**98.1**	**74**	93.4	**73.5**	**96.9**	**89.2**	**91.1**

Table 3.5.
AUC for the base classifier and COP refined classifier for the 10 aesthetic attributes

	nice colors	beautiful scene	nice perspective	big congrats	so cute
Base classifier	0.63	0.62	0.59	0.58	0.58
COP	0.65	0.63	0.59	0.61	0.60
	too small	distracting background	snap shot	very dark	bad focus
Base classifier	0.59	0.59	0.58	0.56	0.57
COP	0.60	0.59	0.60	0.59	0.58

4. LEARNING DEEP FEATURES FOR IMAGE EMOTION CLASSIFICATION

4.1 Introduction

In recent years, with the help of modern social media, millions of images are posted online everyday. People use these images to share life events and express emotions. Having the huge image collections, it is tempting to ask: what can we learn from the images? More specifically, we need to infer both the explicit topic, objects, and the implicit topic, emotions, that are presented in the images. Meaningful solutions to the above problems can help to better understand the people and provide customized services for each individual.

To answer the first question, it involves solving object recognition tasks such as object classification, object detection, and object segmentation, which all have objective criteria and are straightforward to answer by human being. In fact, algorithms in these fields are already very mature now. The best object classification system so far can achieve comparable performance to human being on a benchmark dataset ImageNet [19].

The second question has not drawn much attention yet, while it can actually be harder than object recognition in the sense that it is defined in a more subjective and abstract level. Image emotion understanding aims to classify or retrieve images based on the pre-defined emotion category. Figure 4.1 shows example images of four categories from a public dataset ArtPhoto. For human being, it is probably not too complicated to classify the images into different emotion categories from the content and style of the images. However, there is no straightforward way to describe the emotions conveyed by the images for machine.

In this chapter, we explore a computational approach to recognize images of different emotions by applying some of the latest advancements in deep learning. The rest of this chapter is organized as follows. We describe our deep learning-based image emotion classification approach in Sec. 4.3. In Sec. 4.4, the dataset for evaluation and experimental results are shown. Finally, we draw a conclusion in Sec. 5.8.

4.2 Related Work

A lot of the early work in emotion understanding focused on designing hand-crafted features based on psychology and art theory. Joshi et al. [69] suggested that emotions are highly related to aesthetics, and using features for aesthetic analysis such as compositions, emphasis, and depth of field can help emotion prediction. In [69–73], low-level features such as color and texture, and high level attributes such as human face are extracted for image emotion understanding. Zhao et al. [74] explored the use of principles-of-art based features and showed improved result. The disadvantage of hand-crafted features is that they are designed based on observations and common sense. It is very likely that there are some other factors we are not aware of that are also important. Zhao et al. [75] applied generic image descriptors such as GIST and Histogram of Oriented Gradients (HOG) combined with hand-crafted features finding that they are more robust than hand-crafted features and can generalize well to emotion based image retrieval.

Despite the success of hand-crafted features and generic image descriptors, recent development in convolutional neural network (CNN) has demonstrated great success of automatically learned features. The best object classification system on ImageNet [19], a benchmark dataset with millions of images of 1000 object classes, is based on CNN. Razavian [46] showed that the features directly extracted from a CNN trained on ImageNet can produce superior results compared to some state-of-the-art systems on a variety of visual recognition tasks such as scene recognition and image retrieval. Karayev et al. [76] also applied CNN features to recognize image style without any

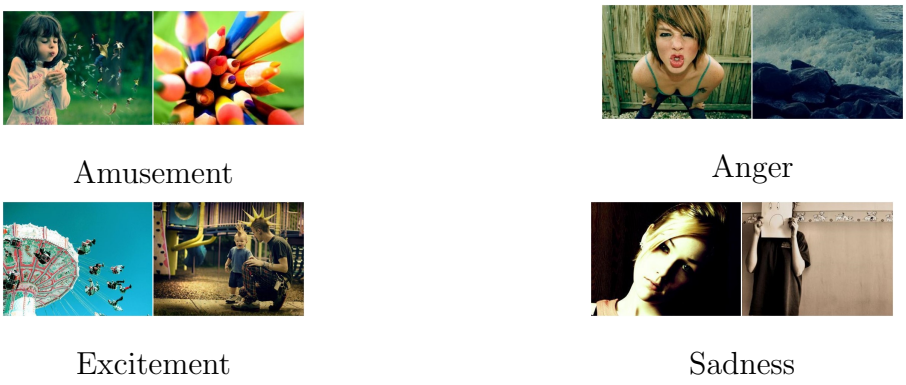


Fig. 4.1. Example of sample images of four emotion categories from ArtPhoto.

knowledge of the data and task and achieved results that are comparable to human performance.

All the above examples show that CNN can be used as an off-the-shelf tool for feature extraction. Motivated by the great potential, we explore a deep learning-based approach for image emotion classification. Instead of designing the features manually as in previous approaches, we use a CNN that automatically learn the image representations through multiple convolutional and fully-connected layers. We extract off-the-shelf CNN features from a pre-trained network [20] to perform image emotion classification. We also show that with fine-tuning and local patch feature pooling using Fisher Vector, we can make the features data-specific and task-specific. The pooled features are then applied to classify image emotions.

4.3 The proposed approach

In this section, we introduce two CNN-based methods for image emotion classification. One method directly uses CNN features from a pre-trained network for classification. The other method aggregates both global and local information by extracting CNN features of the whole image and local patches on multiple scale level.

4.3.1 Off-the-shelf CNN features

As shown in [46], the image features extracted directly from a CNN that was pre-trained on a large dataset such as ImageNet can be used as a powerful image descriptor for visual recognition in general. We take the ImageNet model trained by [20] (AlexNet) without any fine-tuning. Then we extract features from the last fully-connected layer (referred to as fc7 in [20]) using the open-source deep learning library Caffe [61]. The network takes an RGB image of any size, resizes it to 256×256, and outputs a 4096-dimensional feature. The feature vector is $L2$ normalized before training for the classifier. For each emotion category, we train a one-vs-all linear SVM of the form in Equation 5.2, where y_i is the label and $\mathbf{x_i}$ is the 4096-dimensional feature.

$$\min_{\mathbf{w},b} \quad \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \max\big(1 - y_i(\mathbf{w^T}\mathbf{x}_i + b), 0\big) \tag{4.1}$$

Using off-the-shelf CNN features is an easy way for image emotion classification as it doesn't require any knowledge of the task. However, as the original network was trained for single object classification, it is likely to response higher on object type of images. A discussion of the type of images that this method tend to classify as positive samples are presented in Sec. 4.4.4.

4.3.2 CNN features with multi-scale pooling

The previous method uses features that was originally learned for object classification. A normal way to improve on that is to fine-tune the network. We start with the parameters of AlexNet and re-train the network to fit the target dataset. More details of fine-tuning can be found in [46].

Intuitively, we expect that emotions of images are implied from both the global geometric structure and the summary of fine-grained local details over the entire image. So simply applying CNN on the entire image is very likely to miss some discriminating information. In order to account for the fine-grained details, we extract

features for local patches at multiple scales similar to [50]. First, every image is resized to 256×256. We extract the 4096-dimensional features from the resized image using the fine-tuned CNN on four scales, i.e. 256×256, 128×128, 64×64, and 32×32. Each patch is cropped at a step size of 32 pixels over the entire image. The first level has a 4096-dimensional feature corresponding to the full image, while the others have one 4096-dimensional feature for each patch. These patches are supposed to contain more fine-grained local details than the full image.

Except for the first level, which contains only the full image, we need to aggregate the 4096-dimensional features of all the patches to form a single representation. Here, we compute the Improved Fisher Vector (IFV) [37] at each level. A Gaussian Mixture Model (GMM) of K mixtures is first estimated. The first and second order differences between the features and the mean of the Gaussian mixtures are accumulated for each dimension of the feature. IFV has proven to be more powerful than the feature pooling method used in [50] Vectors of Locally Aggregated Descriptors (VLAD) as VLAD only encodes first order differences. The fisher vector has a very high dimensionality of $2K \times 4096$. We perform Principal Component Analysis (PCA) on the fisher vector and reduce the dimensionality to 4096. Now we have a 4096-dimensional vector for each scale. Concatenating them together gives us the final image representation for classification. Figure 4.2 is an illustration of the multi-scale pooling scheme.

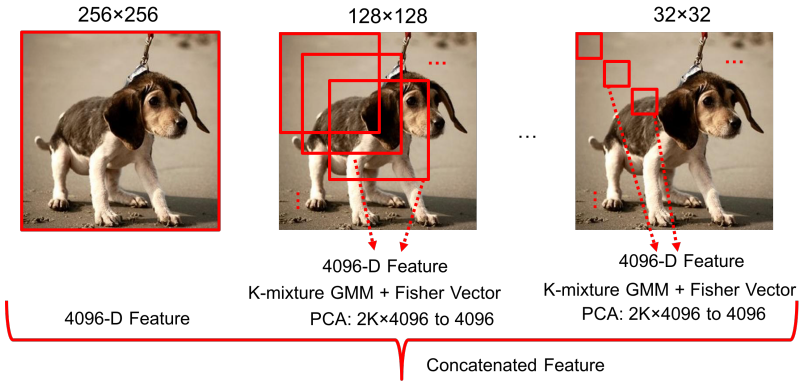


Fig. 4.2. Illustration of the multi-scale pooling scheme

4.4 Experimental Results

4.4.1 Dataset

ArtPhoto. We performed experiments on ArtPhoto [71], a public dataset for image emotion classification. This dataset consists of 806 images downloaded from an art sharing website. Each image is categorized into one of the eight emotion classes: *Anger, Disgust, Fear, Sadness, Amusement, Awe, Contentment, and Excitement*. The distribution of the emotion classes of this dataset is shown in Tab. 4.1. We separated the dataset into training and testing images using 5-fold cross validation as described in [71]. The number of images for each class are approximately the same across all 5-folds.

FlickrEmotion. The size of ArtPhoto is relatively small compared to other datasets that has seen the success of deep learning. So we created a new dataset called FlickrEmotion by ourselves. This dataset contains 11,575 images downloaded from Flickr. The images are acquired by searching the eight emotion classes in ArtPhoto. The number of images for each class is shown in Tab. 4.1. For each class, the images are divided into two halves for training and testing. In total, there are 5,786 training images and 5,789 images, respectively.

Table 4.1.
The numbers of images per emotion class for ArtPhoto and FlickrEmotion.

Dataset	Amusement	Anger	Awe	Contentment	Disgust	Excitement	Fear	Sadness	Total
ArtPhoto	101	77	102	70	70	105	115	166	806
FlickrEmotion	1,429	1,452	1,470	1,490	1,425	1,393	1,462	1,454	11,575

4.4.2 Evaluation Metric

As in [71], we use average true positive rate to evaluate the effectiveness of our proposed approach. This makes it easy for us to compare with state-of-the-art result.

For each class, the true positive rate is averaged over the positive and negative samples. This procedure is independent of the number of positive and negative samples.

4.4.3 Baseline Method

Shallow Features + SVM. We define a baseline method for the comparison of emotion classification. The features are referred to as shallow features as opposed to the features extracted from the deep convolutional neural network. We compute color histogram, GIST and SIFT as in [5, 77]. Suggested by Joshi [69], we also used aesthetic related features [73, 78, 79]. The features are normalized before training a linear SVM with the same procedure as that in Sec. 4.3.1.

4.4.4 Image Emotion Classification

For the experiments, we follow the procedures described in Sec. 4.3. Due to the difference in data size, we applied slightly different parameters for the two datasets. As ArtPhoto is a very small dataset, when fine-tuning the CNN, it requires a smaller learning rate than FlickrEmotion. The number of GMM mixtures is set to 64 for ArtPhoto and 256 for FlickrEmotion. When performing PCA before computing the Fisher Vector, the dimensionality of the 4096-dimensional features is first reduced to 400. For the classifier, the best SVM parameters C for all experiments are determined by 5-fold cross validation on the training set.

Figure 4.3 compares the performance of our baseline method, two proposed methods, and the state-of-the-start result produced by Zhao et al. [74] on ArtPhoto. We can see that, CNN-based methods performed better than previous approach based on hand-crafted and generic features for every emotion category. After fine-tuning and applying multi-scale feature pooling, a consistent gain of 1% is achieved over off-the-shelf-CNN features. The improvement from off-the-shelf CNN to multi-scale CNN is not very significant. This is probably due to the fact that when the data

is far from enough as it is for ArtPhoto, a deep network provides too much degree of freedom. Fine-tuning a CNN can easily over fit the training set even with a low learning rate and regularization techniques.

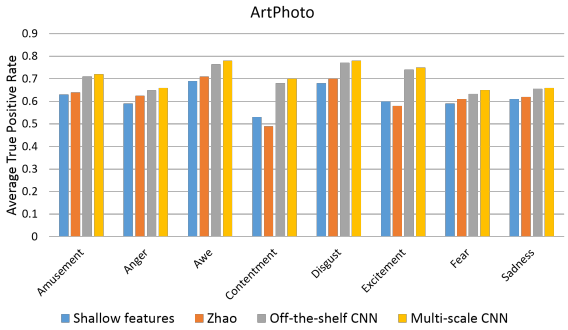


Fig. 4.3. Performance of our proposed methods on ArtPhoto for each image emotion class compared with Zhao [74]

The comparison between the baseline method with our proposed methods on the larger dataset FlickrEmotion is shown in Fig. 4.4. Since the baseline method achieved comparable result to Zhao's method on ArtPhoto, we consider it as a good indicator of how well the proposed methods compare with previous work. From the chart, we can see that CNN-based methods outperform previous approach. Multi-scale CNN surpasses over off-the-shelf CNN by around 5%. In this case, we see that a large dataset really helps to improve the fine-tuned model used in multi-scale CNN.

In Fig. 4.5, we show the top two images classified as positive samples using off-the-shelf CNN features on FlickrEmotion. The results are generally accurate and match our expectation. Interestingly, we notice that a large portion of the top samples contain people. On one hand, many photos in the database indeed contain people and other objects. On the other hand, as mentioned in Sec. 4.3.1, the pre-trained model that we adopt are originally used for object classification, which means it is sensitive to object type of features that characterize the emotion categories, e.g. human with widely opened mouth characterizes anger, a specific type of insects looks disgusting

by nature. This is also consistent with our assumption that taking the full image as input tend to capture global geometric layout such as shape and structures.

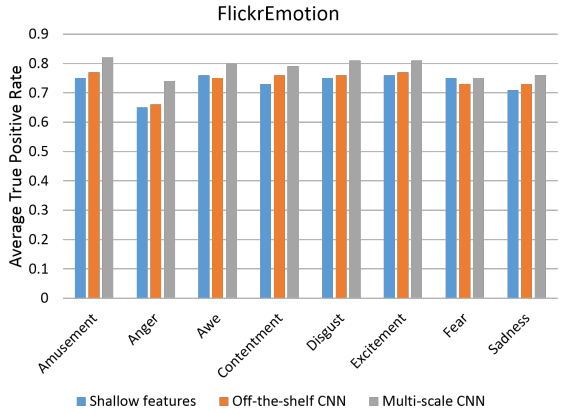


Fig. 4.4. Comparison between the baseline method and our proposed methods on FlickrEmotion

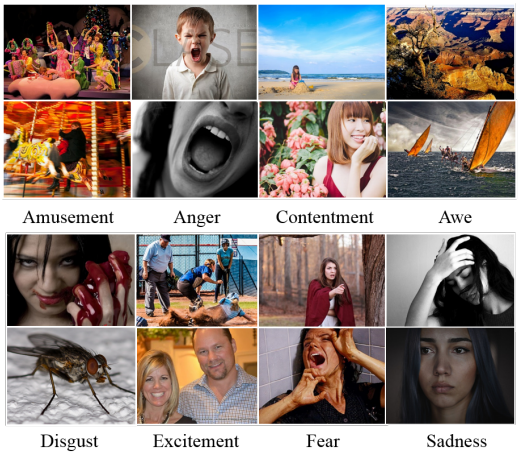


Fig. 4.5. Top two images for each class using off-the-shelf CNN features

To demonstrate the effect that features from local patches has on the types of images recognized, we show the top two images classified as positive samples using multi-scale CNN features. As can be seen in Fig. 4.6, more scene type of images are returned. The results from multi-CNN Image emotions are more aligned with our

definition of emotions. This suggests that abstract concepts such as emotions is the joint effect of concrete attributes such as objects and scenes.

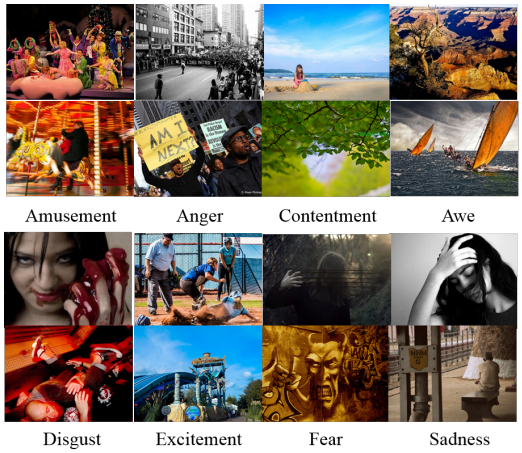


Fig. 4.6. Top two images for each class using multi-scale CNN features

4.5 Conclusion

In this chapter, we addressed the issue of image emotion analysis using a deep learning-based approach. Two methods for image emotion classification were explored. The first method used off-the-shelf CNN features on full image to capture global features. The second method extracted features locally followed by feature pooling using the Fisher Vector. Experiments were conducted to compare our approach with the state-of-the-art. It was shown that our approach outperforms existing methods that are based on hand-crafted features and generic features.

Considering the popularity of social network nowadays, a possible direction for future work is to consider image emotions in the context of social media, where textual information and the interaction between people can be taken into consideration.

5. PERSON DETECTION AND RECOGNITION IN NATURAL SETTINGS

5.1 Introduction

Person detection and recognition is one of the fundamental problems in image understanding. Knowing the location and identity of persons in the image leads to a lot of everyday applications. On social network and online media, detecting persons and tagging the identity has become a convenient way to share memories and organize the photos. In shopping mall and other public area, detecting and recognizing persons from surveillance cameras is an essential approach to automate applications in public safety. In a home/office environment, knowing the presence of persons can help energy harvesting, customized services, and anomaly detection.

However, detecting and recognizing persons in natural settings is a challenging task. Due to the unconstrained environment and the long time span, people can have different pose, wear different clothes, and undergo various lighting conditions and occlusions.

In this chapter, we investigate person detection and recognition as a whole system for home/office environment in a less constrained setting, where images contain person that have different pose, viewpoints from multiple days. We propose the use of head region instead of face or full body for person detection. We show that we can both reliably detect the person and extract powerful features that are mostly invariant to time and help with the recognition stage. To verify our person detection and recognition system, we collect ground truth annotation of head bounding boxes and identities for a TV series Modern Family. With the collected dataset, we investigate several use case of person detection and recognition system in home/office environment: person recognition with fully labeled data, unsupervised person clus-

tering, interactive person recognition with minimum annotation, and semi-supervised interactive person recognition. We propose algorithms to handle different scenarios and show the effectiveness of the overall system.

5.2 Related Work

Person detection and recognition, have been studied for a long time. In the bulk of previous work, person detection and recognition have been mostly treated as two separate problems, each of which has seen great progress in recent years.

Person detection is a vague term as it does not specify what body part is detected. In a lot of previous research, person detection is phrased as face detection or full body detection. Viola-Jones [80] is the textbook face detector that uses Haar feature-based cascade classifiers. This face detector is very fast, but only gives moderate detection performance. Later generic object detector based on the Deformable Parts Model (DPM) [60] has been proved to be effective for both face detection and body detection [81]. DPM models the the object by the appearance and deformation. Here, the appearance for the whole object and each part is represented by the Histogram of Oriented Gradient (HOG) [34]. The deformation calculates the deviation of parts from its ideal location relative the root. The training process will optimize the cost defined by appearance response subtracting the deformation cost at different location and scales. Recent progress in deep learning based approaches such as R-CNN [49], Fast R-CNN [82], and Faster R-CNN [83] have improved the performance of human body detection and face detection by a large margin. While it is required that frontal face, or at least a large portion of the frontal face is visible for face detection, body detection is more robust to different pose and viewpoints.

Similar to person detection, the main effort towards person recognition is on face recognition. This research field has seen great progress in the last few decades from the ones using had-crafted features [84], to more deep learning based system such as [85]. Schroff et al. [86] use large scale proprietary data to train a network with triplet

loss. Parkih et al. [87] also use triplet loss to learn an embedding of face features. These existing work mostly focus on frontal face images with little occlusion. All the above-mentioned face recognition systems more or less require face alignment as a preprocessing step.

Person detection and recognition have also been framed in the context of naming characters in TV series. The majority of this branch of work use multiple cues to recognize the persons. In [88] visual information from face and clothing appearance and textual information from the subtitles are aligned to help recognizing characters. Tapaswi et al. [89] models each episode as a Markov Random Field, integrating face and clothing appearance, speaker recognition and contextual constraints in a probabilistic model. In [89], face descriptors and multiple instance learning is applied and it is demonstrated that only using subtitles can give good results.

A closely related research area to person recognition is person re-identification [90]. In this setting, the same person is captured by cameras at different location and different time of the day. The task is to identify the person captured by one camera given a set of images captured by other cameras. It is expected that people across different time of the day and different location wear the same clothes. Before the rise of deep learning, existing work focus on metric learning [91] and mid-level representation learning [92, 93]. Most recent work [94] have been trying to learn similarity metric through deep network using pairs of images captured from different cameras.

Recently, a dataset Person in Photo Albums (PIPA) [95] is released to help with the research in person recognition in a less constrained environment. Unlike previous research on face detection and face recognition. This work investigated the case where the frontal face is not necessarily visible. The PIPA dataset that they published contains images from every day life from thousands of persons. The author proposed an approach based on combining face recognition model and classifiers for several poselets and reported 83% accuracy for person recognition. Achieving such a high accuracy on a what seems to be a very sceptical. A follow up paper by Oh et al. [96] investigate the flaws in the experiment protocol and found that images from the same

day where people could be wearing the same clothes or even having nearly identical poses are split across the training and testing set, which explains the overly high performance. They propose to split the training and testing set according to albums or time of the day so that person recognition can be evaluated in a more realistic manner.

The datasets for face detection, face recognition, person re-identification and person recognition in general are different. The visible body parts, image quality, clothing type, and pose are all different depending on the specific tasks. A more detailed review of all the differences will be given in 5.5.1.

5.3 Person detection

A person detector is a system that generates a rectangular bounding box surrounding a person whenever a person occurs in the image. It can be applied simply for knowing the location of the person, or for presence detection where we would like to know how many persons there are in the scene. In our scenario, person detection serves as the front-end of a person recognition pipeline. In order for the following recognition engine to perform well, the detector needs to generate as many tight bounding boxes as possible, while avoiding false detections. A good detector can be crucial for the overall performance of the detection and recognition system.

There are two main criteria when developing the person detector. The first criterion is that the body part should be reliably detected in home/office environment. The second criterion is that the detected body part should be effective for the latter recognition stage, which means it should focus on the core features of a person and be invariant to time, pose, and lighting conditions. Intuitively, human being recognize people largely from the facial feature. But in many real case scenarios such as smart home/office applications, the frontal face is not necessarily available and only the side view or even the back view is visible. Face detection will certainly miss a large number of persons. An alternative body part to face for person detection

is human body. For most benchmark body detection datasets [24, 97], the training data contains images of a large variety of body pose, which makes body detection robust in home/office environment. However, since people can change their clothing in home/office environment from time to time, extracting features from the full body may capture too much information that are not invariant to time. This can lead to severe overfitting and very pool generalization performance.

5.3.1 Head Detection using Faster R-CNN

Since neither face detection nor human body detection satisfies the two criteria for our application, a different body part is needed for robust detection and recognition. We propose the use of a less widely explored body part: head for detection and the following recognition task. By definition, head can be either frontal view, side view or even back view. In addition, most part of the head region remains unchanged from day to day as in the case of face. It also captures some contextual information like hairstyle, hair color that can help recognizing different people.

To train the head detector, we apply a state-of-the-art object detection framework Faster R-CNN [83]. Unlike previous detection framework such as R-CNN [49] and Fast R-CNN [82] that are composed of three separate stages, i.e. proposal extraction, proposal classification, and bounding box regression, Faster R-CNN is an end-to-end deep convolutional neural network that combines all three stages into a single network. The network takes image as input and predict the class of region proposals. Unlike region proposals in R-CNN and Fast R-CNN that are generated by traditional methods such as Selective Search [58] and EdgeBox [54], the region proposals in Faster R-CNN are generated by a branch out sub-network called Region Proposal Network (RPN). This sub-network shares the first few convolutional layers of the main detection network as described in Fast R-CNN that mostly look at edge and blob-like low-level features and can thus save a lot of computation. During training, it jointly minimizes the classification loss of the region proposals generated by RPN and

Table 5.1.
Statistics of PIPA dataset

Split	All	Train	Val	Test	Leftover
Photos	37,107	17,000	5,684	7,868	6,555
Albums	1,438	579	342	357	160
Instances	63,188	29,223	9,642	12,886	11,437
Identities	2,356	1,409	366	581	-
Avg/identity	26.82	20.74	26.34	22.18	-
Min/identity	5	10	5	10	-
Max/identity	2,928	99	99	99	-

the distance between the region proposals with the ground truth bounding box. We refer readers to the original Faster R-CNN paper [83] for a more detailed description of the algorithm.

The dataset we use to train the head detector is Person in Photo Albums (PIPA) [95]. All the images are crawled from Flickr and are annotated with head bounding boxes. Here we use the ground truth head bounding box annotation to train the head detector.

Table 5.2 shows the training time, test time, and mean average precision (mAP) of the head detector using two different pre-trained network on PIPA. We can see that with a better pre-trained network, i.e. VGG16, the detection improves by 2.1% over ZF net.

Table 5.2.
Training and testing results of head detection

	ZF	VGG16
Training time (hr)	10	22
Test time (ms)	59	198
mAP (%)	67.6	69.7

In Fig. 5.1, we show some head detection results with the model pre-trained on VGG16. It can be seen that the head detector is able to draw a tight bounding box around the head for frontal view, side view, and back view. The head detector can also handle different lighting conditions.

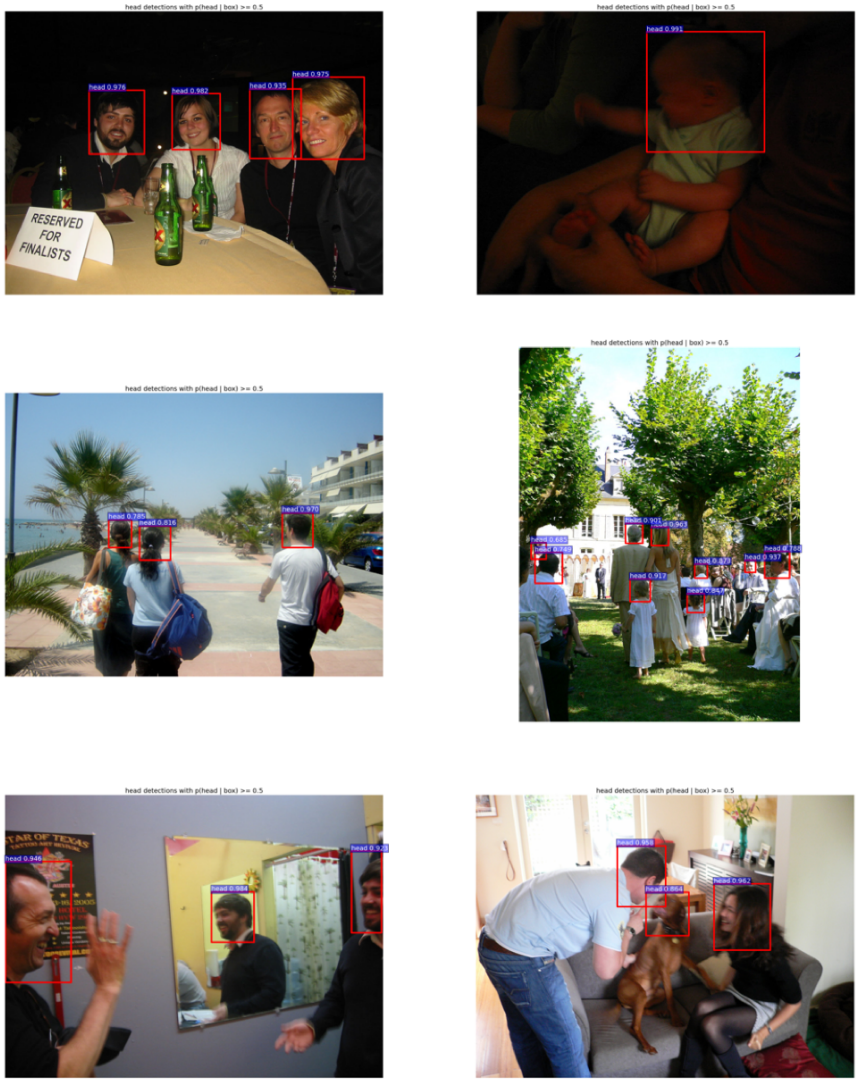


Fig. 5.1. Head detection results with the model pre-trained on VGG16

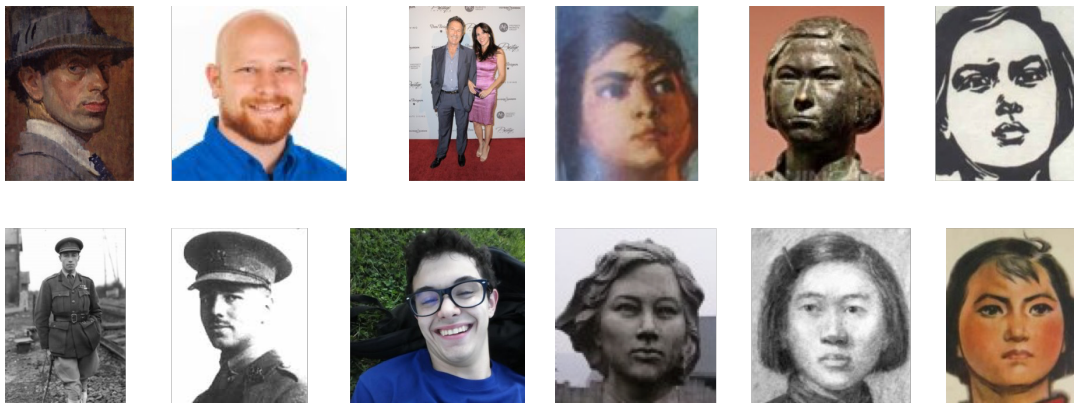
5.4 Person Feature Representation

As proposed in Sec. 5.3, we detect head instead of face or body for the person detection task. Features in the head region will then be extracted as the person representation and fed into the recognition system. Recent progress in deep learning [20–22, 44, 45] has shown that, the features learned from the deep network can be easily transferred to other applications that are different from the original tasks. This means that for person recognition, without the need to collect a large-scale person recognition datasets in home/office environment, we can fully utilize available public datasets. Training on these external datasets can help learn a powerful head features that can discriminate between different identities even in home/office environment that the model has never seen.

5.4.1 Data Cleaning

The dataset we use for training robust head features is called MS-Celeb [98]. This dataset contains images of around 100,000 identities, each containing images ranging from tens to several hundred. The public version of the dataset is essentially raw images crawled from the internet by search queries without any data clean up. The noisiness of the dataset is twofold. On the one hand, unlike other datasets such as CASIA-WebFace [99] and MegaFace [100] that are mostly clean frontal face images, MS-Celeb contains more side views and extreme pose images that adds to the variety of training images. On the other hand, as shown in Fig. 5.2(a) some of the identities are too noisy that they contain images from a number of different people, other identities are confused with fictional characters and are overwhelmed with images that are not real human as shown in Fig. 5.2(b).

In order to not confuse the learning process, we need to filter out the noisy portion of the dataset and only train on identities that are relatively cleaner. We use a proprietary method developed at HP Labs to select the clean identities and perform head detection using the previously trained detector.



(a) Identities that contain multiple person (b) Identities that are not real human

Fig. 5.2. Sample images from the MS-Celeb dataset.

5.4.2 Training Robust Head Features

Now that the detected head regions for the chosen 10,000 identities are available, we can start training the head model. We adopt the AlexNet architecture proposed in [20]. AlexNet contains five convolution layers and three fully connected layers. Each convolution layer is followed by a max-pooling layer and ReLu layer. The first two fully-connected layers (fc6 and fc7) are of dimension 4096. Both fc6 and fc7 are followed by a dropout layer where each neuron is randomly disabled at a fixed probability in each iteration. The last fully-connected layer along with its following softmax layer has dimension of $10,000$ which is the same as the number of identities. For the cost function, we use the cross-entropy loss, which has the form:

$$L(\hat{y}, y) = -\sum_{k=1}^{K} y_k \log \hat{y}_k = -\log \hat{y}_{y=1}, \tag{5.1}$$

where the first equation is the definition of cross-entropy, and the second equality simplifies the expression because the true distribution in classification settings are usually assumed to have probability of 0 on the wrong elements, and probability of 1 on the single correct element, whose integer index is denoted as $y = 1$.

The dataset is divided into a training set which contains 80% of the images and a testing set which contains 20% of the images. During training, the images are scaled

to 256×256 and randomly cropped to 227×227, which is the input dimension of the network structure. At test time, we use one scale testing by keeping the input size to 227×227 as opposed to the multi-scale testing scheme where the predictions of several randomly cropped 227×227 images from the 256×256 image are averaged.

We run stochastic gradient descent with momentum for the optimization process. Four Titan X GPUs are used to train the model. On each GPU, the batch size is 32, making the effective batch size 128. The base learning rate is set to 0.01 and is multiplied by 0.1 every 300,000 iterations, which corresponds to around 200 epochs. Momentum is set to 0.9. Dropout ratio is fixed at 0.5.

Figure 5.3 shows the training loss and testing accuracy with respect to the number of iterations. After a slow start, the test accuracy gradually converges at 66%. It further increases starting at 300,000 when the learning rate is decreased to 0.001 and reaches a maximum of 87.4%. It took about 30 hours to converge to the final accuracy.

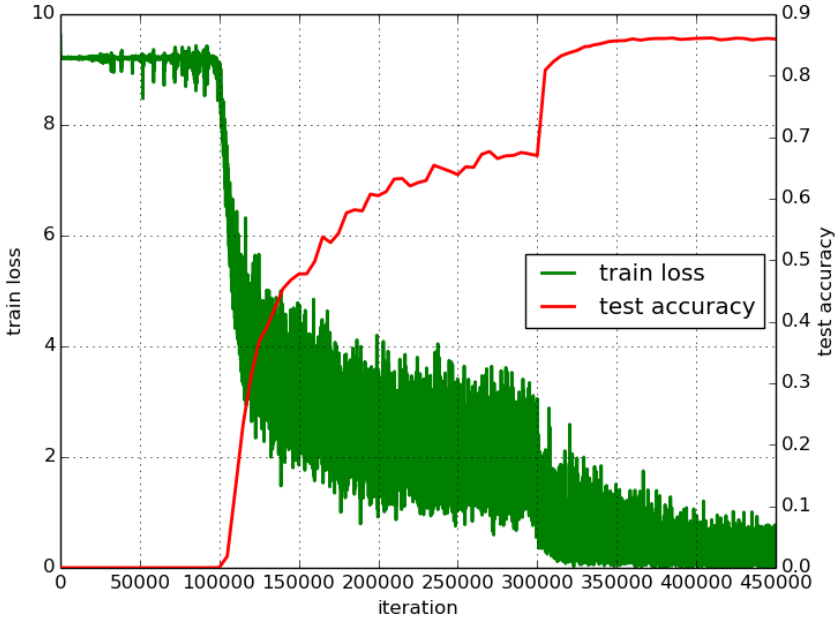


Fig. 5.3. Training loss and test accuracy with respect to the number of iterations for head model training

We should note that both training the head model and extracting features do not require any alignment as is in the case for most face recognition systems.

5.5 Dataset for Person Detection and Recognition System

5.5.1 Public Dataset for Person Detection and Recognition

To validate our person detection and recognition pipeline, we need a dataset that contain images similar to our home/office settings. In the research community of person detection and recognition, most of the effort has been on face detection and recognition. AFW [101] and FDDB [102] are two datasets for face detection. LFW [103] was one of the most popular benchmark dataset for face identification. Most recently, CASIA-WebFace [99], and MegaFace [100] are made open to the public where millions of images from hundreds of thousands of identities are available for training face recognition models. Figure 5.4 shows some sample images from those public face datasets. However, as the name implies, face datasets mostly contain images of frontal faces, which is a much more constrained setting than our use case where there is a large variety in head pose.

Another research area related to person recognition is person re-identification. An illustration of person re-identification is shown in Fig. 5.5. The same person is captured by different surveillance cameras at different time of the day and the task is to retrieve the same person from the gallery set captured by one camera given a query image captured by another camera. There are also a few public datasets for person re-identification. The images in earlier datasets such as VIPeR [104] and CUHK01 [91] are captured by two cameras and each person has exactly one image from each camera. Recent large scale datasets such as CUHK02 [91] and CUHK03 [93] have images captured with more than two cameras. Person re-identification originates from video surveillance applications where all the persons are of very low resolution. Since the face are barely seen clearly, most approaches utilize the clothing information to recognize persons. While the clothing of a person remains unchanged across cameras

in a person re-identification task, this assumption is not necessarily valid for smart home/office applications as we want to recognize persons on different days. The persons are expected to wear totally different clothes and the visual appearance of the clothing can vary a lot.

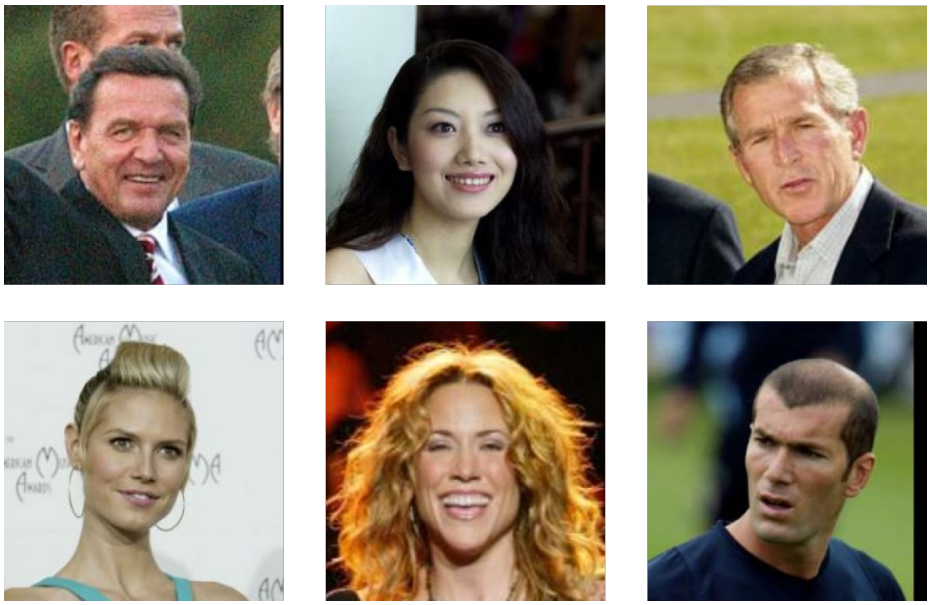


Fig. 5.4. Sample images from LFW dataset

5.5.2 Collecting Annotation for TV Series

Since there is not a public dataset that quite matches our use case, it motivates us to collect our own dataset. We purchased a popular sitcom Modern Family and label the person bounding boxes and identities for the first three of the 25-minute episodes of Series 1. The reason for choosing this TV series is that the setting is highly similar to our use case in home/office applications. Modern Family contains mostly scenes in a home environment where the characters are doing what people would do in everyday life: cooking, talking with each other, playing and so forth. The characters are not necessarily looking towards the camera, which means there is

(a) Images from VIPeR dataset. The upper row shows images from one camera and the lower row shows images from the second camera

(b) Images from CUHK01 dataset

Fig. 5.5. Sample images from person re-identification dataset.

a large variety of head pose. Since there are multiple episodes, the clothing of the characters and the lighting conditions change from time to time.

There are ten main characters and a number of other people in the TV show. Figure 5.6 shows a group photo of all the main characters in Modern Family. The main characters are of different gender, ages and hairstyle. For all but the main characters, they are treated the same as a joint class of "unknown" person. As discussed in Sec. 5.3, we suggest that head is the most effective region for person detection in an unconstrained setting. It is not only more to different clothing invariant than human body but also more robust to different pose and lighting conditions than face detection. Following the hypothesis, for Episode 1, Episode 2, and Episode 3, we annotated the head bounding box and the corresponding identity.

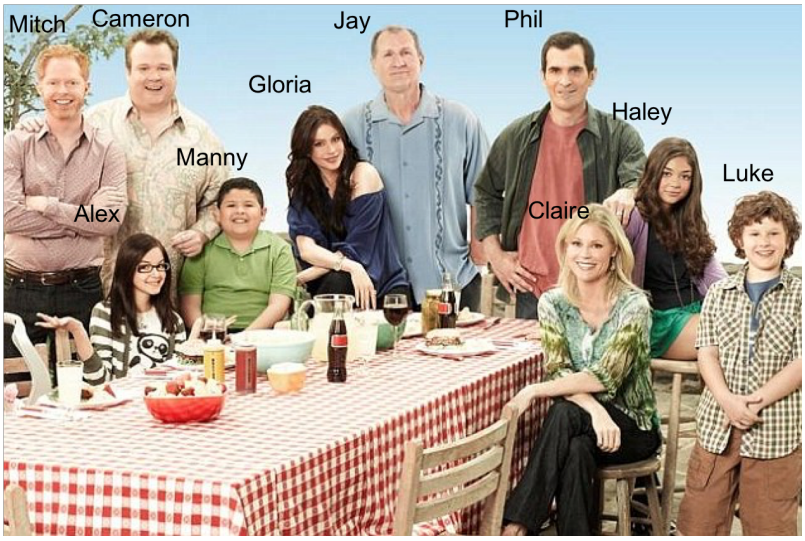


Fig. 5.6. A group photo of the ten main characters

To efficiently annotate large amount of videos, we used an open source annotation tool called Vatic [105]. In Fig. 5.7, the annotation interface of Vatic is shown. The videos are divided into 10-second segments and the annotators work on one segment each time. The annotators are instructed to draw a bounding box around the head of a person, be it frontal view, side view, or back view, and associate it with that person's identity as long as at least half of the head is visible. Thanks to the tracking functionality integrated in Vatic, the annotator only needs to manually annotate some user-defined key frames in the video segment. The frames between consecutive key frames can be interpolated by the tool itself. When the annotation is finished, the tool samples one frame out of every 15 frames as the final output, which avoids keeping too many frames that are highly redundant.

After outputting the annotation results, we apply a simple blurry image detection algorithm based on overall edge intensity to the annotated bounding box. Blurry frames as shown in Fig. 5.8 are mostly due to camera and person movement. Removing these frames can help reduce ambiguity when training the recognition system.

In addition to the four labeled episodes, we also run the VGG16 head detector trained in Sec. 5.3.1 on three more episodes to create a set of unlabeled frames. This

Fig. 5.7. The annotation interface of Vatic

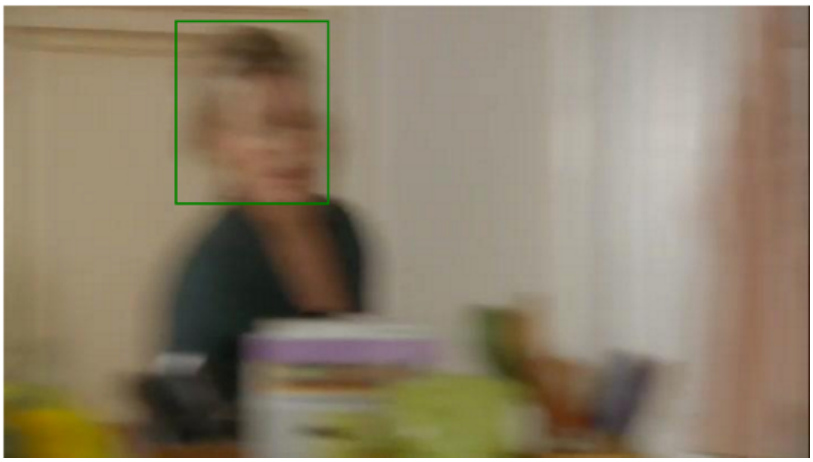


Fig. 5.8. Blurry image that needs to be removed

Table 5.3.
Summary of annotated data statistics

	Unknown	**Alex**	**Cameron**	**Claire**	**Gloria**	**Haley**	**Jay**	**Luke**	**Manny**	**Mitch**	**Phil**
Episode 1	316	150	351	365	455	156	503	112	199	400	432
Episode 2	50	18	244	232	196	56	174	18	29	299	239
Episode 3	483	0	456	121	163	0	342	74	168	362	496

will be used for experiments on semi-supervised learning. Table 5.3 shows a summary of the number of annotated frames for all characters.

5.6 Person Detection and Recognition in Several Real-life Scenarios

In Sec. 5.3 and Sec. 5.4, we introduce a head detector that can reliably detect the head region of a person and a CNN model that extract features over the head. In this section, we consider several real-life scenarios of a person detection and recognition system. These scenarios are mainly different in the amount of labeled data that is available.

5.6.1 Person Recognition with Fully Labelled Data

In a standard supervised classification setting, there is a training set that are fully labelled. The goal is to train a classifier that fits the training set as good as possible without possibly overfitting the data. The testing set is then used to evaluate the classifier. For our first real-life scenario, we consider the case where the system is given a set of labelled images from different identities. The task is to classify the images in the testing set.

For the training set, we assume that both the bounding box and the identities are available. The classifier is trained on fully labelld data. For the testing set, we consider both cases where the bounding box is available or not. When the bounding box is available, we simply extract the features using the head model and test it with the classifier. When the bounding box is not available, we run our head detector and compute the Intersection over Union (IoU) between the detected head with the ground truth head to find out the ground truth identity. If the highest matching IoU is below a threshold, which we set to 0.4 experimentally, the detection is considered as a false alarm. Otherwise, the identity corresponding to the highest matching IoU is assigned to the detected head.

The classifiers we use are Support Vector Machine (SVM) and Nearest Neighbor (NN). For SVM, we simply use the linear kernel and the following cost function:

$$\min_{\mathbf{w},b} \quad \frac{1}{2}||\mathbf{w}||^2 + C\sum_i \max\big(1 - y_i(\mathbf{w}^\mathbf{T}\mathbf{x}_i + b), 0\big), \tag{5.2}$$

where x_i is the 4096-dimensional head features and y_i is the ground truth label. For NN, euclidean distance is used as the distance metric.

5.6.2 Unsupervised Person Clustering

Fully labelled data is not really a realistic scenario as it requires a lot of human effort to annotate. For the second scenario, we consider the case where neither the bounding box nor the identities are available. This is a very common situation in

which no human interaction is engaged in the system. In this case, we perform unsupervised clustering algorithms to group images that are similar in the feature space. Ideally, images of the same person should be clustered together regardless of the pose, viewpoints, and lighting conditions.

Since neither the bounding box nor the identity is available, we need to detect the heads first. Again, we run the head detector trained before and find out the ground truth label associated with it.

The clustering algorithm we use is agglomerative hierarchical clustering with euclidean distance. All the images are initialized to be separate clusters. The nearest two clusters are then iteratively merged to form a new cluster. We use the ward linkage criterion that minimizes the variance of the clusters when merging the nearest two clusters. There are several variants on when to stop the merging process. One approach set a threshold on the linkage criterion and stops it when the distance between the nearest two clusters is below the threshold. Another approach pre-defined the number of clusters we want to keep and stops it when the desired number of clusters is reached.

5.6.3 Interactive Person Recognition with Minimum Annotation

Supervised learning with labelled data can help train a classifier that differentiates between different persons but requires large amount of annotation, while unsupervised clustering group images from the same person together but does not predict the actual identities. It is tempting to combine the two approaches take advantage from both sides. For this scenario, we consider the case where only a set of unlabelled data is available and we want to predict the actual identities that requires the minimum amount of annotation.

First, the set of unlabelled images are clustered using the clustering algorithm described in Sec. 5.6.2. Then a human annotator is asked to assign one identity to each of the clusters that he/she thinks is homogeneous enough. Using the assigned

labels, we train a classifier as in Sec. 5.6.1 and can thus predict the actual identities on the test data. We should note that there could be some images that are mislabelled. We will show that this is acceptable and does not require picking out those outliers in the clusters, which could take a lot of time.

Our approach is related to active learning in the sense that the system can interactively query the users with a small number of clusters for labelling. In this way, instead of labelling thousands of images one by one, only a few clusters need to be labelled.

In our current implementation, the interaction between the system and the annotator is hypothetical. Instead of having real human that annotates the clusters, we mimic the behaviour of a human when determining the homogeneity of the clusters. This is achieved by first finding the dominant identity of the clusters. If the percentage of the dominant class in a particular cluster is above a threshold, we assume that the human annotator will treat it as homogeneous enough and will label the entire cluster with the dominant identity.

Although we have not conducted experiments with real human annotator, we expect that the process will work in a similar manner. We plan to conduct such experiments as our future work.

5.6.4 Semi-supervised Interactive Person Recognition

In Sec. 5.6.3, we introduce an interactive person recognition system that only requires labelling a small number of homogeneous clusters, rather than labelling individual samples as in the supervised learning setting. This can save us a lot of effort annotating the data.

Another way to easily get large amount of labelled data without manually annotating individual samples is to make use of extra unlabelled data. The majority of data out there is unorganized, unstructured, and unlabelled. It would be a great benefit if we can make use of this large amount of unlabelled data. In this scenario, we

build a semi-supervised learning framework that utilizes a large amount of unlabelled data on top of the previously developed interactive person recognition system.

Using the method described in Sec. 5.6.3, we first perform clustering on an unlabelled dataset, and then assign the labels to the clusters that are considered homogeneous to form a training set. Similarly, the interaction between the system and the annotator is simply hypothetical. With the coarsely labelled training dataset, a preliminary support vector machine is trained to classify images into different identities. Next, we apply this preliminary classifier on the extra unlabelled dataset. Only the samples that have decision margin above a threshold are considered confident samples and added to the training set. In practice, the threshold of the margin is set to 1, as this indicates samples with a higher margin does not contribute to the cost function when optimizing SVM. Using the original training set together with the confidently labelled samples from the unlabelled dataset, we train a new classifier that is supposed to be better than the preliminary classifier. We should note that, the preliminary classifier needs to give some kind of measure of how confident the prediction is that can easily be interpreted. This is why we only consider SVM for the preliminary classifier. For the new classifier, either nearest neighbor or SVM can be used.

5.7 Experiments

5.7.1 Experiment Settings

We evaluate the performance of our person detection and recognition system on the Modern Family dataset that is introduced in Sec. 5.5.2. All the real-life scenarios in 5.6 are evaluated. For the clustering task evaluation, Episode 1 and Episode 2. For all other scenarios, Episode 2 is used as the testing set so that there is a fair comparison among different scenarios. We choose to keep images from the same episodes in the same train/test subset as this will prevent images that are highly similar in the scene to appear across the training and testing set, which may lead to

Table 5.4.
The accuracy for the dataset with or without ground truth (GT) bounding box (bbox) with NN and SVM

	NN	SVM
Tested with GT bbox	86.43%	87.90%
Tested with detected bbox	86.13%	88.06%
Tested with detected bbox, evaluated among detected bbox	85.85%	87.7%

trivial solution as described in [96] for the PIPA dataset. For classifier training, we do not explicitly tune any parameters. The models are directly applied to the test set.

5.7.2 Results

Person Recognition with Fully Labelled Data

We first evaluate the performance of person recognition with fully labelled data. As described in Sec. 5.6, both cases where ground truth bounding box is available and not available for the testing set are considered. For person detection, our detector achieves precision of 85.7% and recall of 99.67%. The classification accuracy is shown in Table 5.4. We can see that since the recall is very high, there are only a few heads that are missing, which makes the accuracy with detected bounding box almost as high as accuracy with ground truth bounding box. Regarding the two classifiers, nearest neighbor is only around 1.5% lower than SVM, which means the head features are pretty discriminative.

We also show the two confusion matrices for the two classifiers with detected bounding box to give a little more ideas how each individual class performs. It can be seen that the confusion matrix is mostly diagonal. For both NN and SVM, the unknown class performs the worst. More specifically, unknown class for SVM is worse

Table 5.5.
Homogeneity score with respect to different number of clusters

	30	40	50	60
Homogeneity score	0.73	0.76	0.78	0.79

than unknown class for nearest neighbor. This is probably due to the fact that the unknown class is actually a mixture of different persons and is not very homogeneous. A linear SVM can easily be confused when finding the separation hyperplane for such noisy data.

Unsupervised Person Clustering

For the unsupervised clustering task, we evaluate the algorithm on Episode 1 and Episode 3 together. The result will be used as the preliminary clustering result for the experiment in Sec. 5.7.2. To evaluate the clustering algorithm itself, the ground truth bounding box is used. We then compute the homogenity score of the clusters. In Table 5.5, we show the homogeneity score with respect to different number of clusters. As expected, the homogeneity is affected by the number of clusters. The more clusters there is, the more homogeneous the cluster is.

When ground truth bounding box is not available, our head detector achieves a precision of 81.51% and recall of 96.97% on this subset of data. In Fig. 5.7.2, we show some sample clusters of this subset. We can see that the same person under different pose, viewpoints, and lighting conditions are grouped together, which shows that the feature and clustering algorithm is very effective.

Interactive Person Recognition with Minimum Annotation

In this section, we show the result for the interactive person recognition system. Again, Episode 1 and Episode 3 is used as unlabelled data that will be labelled by a

hypothetical human annotator. Episode 2 is used as the test set. It is shown in Sec. 5.7.2 that the quality of the cluster is affected by the number of clusters. This can also affect the performance in the interactive recognition scenario. Basically, generating more homogeneous clusters while keeping the number of clusters for annotation down is a tradeoff. This tradeoff requires experimenting and understanding the human annotators' tiredness, which is beyond the scope of this dissertation. In this section, we simply fix the number of clusters to 40 for all the experiments. After clustering, all the clusters that have at least 80% of dominant class percentage are assign the identities of the dominant class.

We consider several conditions where the training set bounding box and testing set bounding box could be available or not. Table 5.6 shows the experimental results for the interactive person recognition under different conditions using SVM and nearest neighbor respectively.

When both training and testing bounding box are available, the accuracy is almost the same as that achieved in 5.4. In this setting, 32 out of 40 clusters are considered homogeneous and annotated. With only 32 clusters labelled, the system achieves almost the same performance when thousands of individual images are labelled.

When the training bounding box is not available, 29 out of the 40 clusters that contains $4,136$ heads are kept as homogeneous. Suprisingly, the accuracy is 88.7% which is even higher than the one with fully labelled data. This may be because that as the non-homogeneous clusters are discarded, some of the confusing images are a no longer used for training.

The case when training bounding box is available and testing bounding box is not available is ommitted as it is of very little practical use.

In the last setting, which is the most realistic case, both training and testing bounding box are not available. Since the recall on the test set is 99.87%, the performance is almost the same as compared to the case where testing bounding box is available.

Table 5.6.
Accuracy for the interactive recognition system under different conditions using nearest neighbor and SVM

NN	Training bbox	No training bbox
Testing bbox	84.5%	85.0%
No testing bbox	-	84.8%
SVM	Training bbox	No training bbox
Testing bbox	87.8%	88.7%
No testing bbox	-	88.8%

Table 5.7.
Summary of extra unlabelled data statistics after preliminary prediction

	Unknown	**Alex**	**Cameron**	**Claire**	**Gloria**	**Haley**	**Jay**	**Luke**	**Manny**	**Mitch**	**Phil**
Episode 4-6	1	106	570	683	693	138	716	0	261	1020	1202

In Fig. 5.11(a) and Fig. 5.11(b), we show the confusion matrix for the interactive person recognition system when both training and testing bounding box are not available. There are some columns that are completely blank due to the fact that those classes do not dominate any cluster or do not have a high enough dominant percentage.

Semi-supervised Interactive Person Recognition

In this section, we show the result for the semi-supervised interactive person recognition system. Again, Episode 1 and Episode 3 is used as unlabelled data that will be labelled by a hypothetical human annotator. Episode 2 is used as the test set. We use another three episodes: Episode 4, Episode 5, Episode 6 as extra unlabelled data that is to be labelled by the preliminary SVM. The total number of frames in the extra unlabelled data is 7656. Table 5.7 shows the statistics of the extra unlabelled data statistics after the preliminary prediction using SVM. We should note that one of the classes does not have any predicted samples. This is because that class does not dominate any cluster or its proportion in the cluster does not exceed the threshold in the interactive recognition stage as shown in Sec. 5.7.2.

For simplicity, we only consider the most realistic case where the ground truth bounding box is neither available for training nor testing. All the bounding boxes are generated by the head detector. Table 5.8 shows the accuracy in this scenario using nearest neighbor and SVM. We can see that the performance for both classifiers are improved. Nearest neighbor improves from 84.8% to 86.6%. SVM improves from

Table 5.8.
Accuracy for the semi-supervised interactive recognition system using nearest neighbor and SVM: No ground truth bounding box is used for both training and testing.

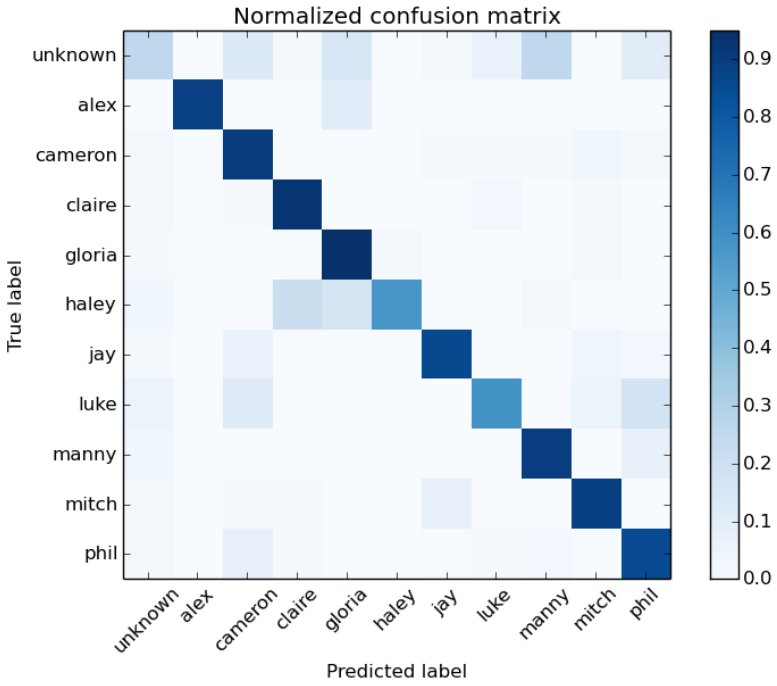
	NN	SVM
Accuracy	86.6%	89.3%

88.8% to 89.3%. This proves that using extra unlabelled data can indeed help improve the recognition performance.

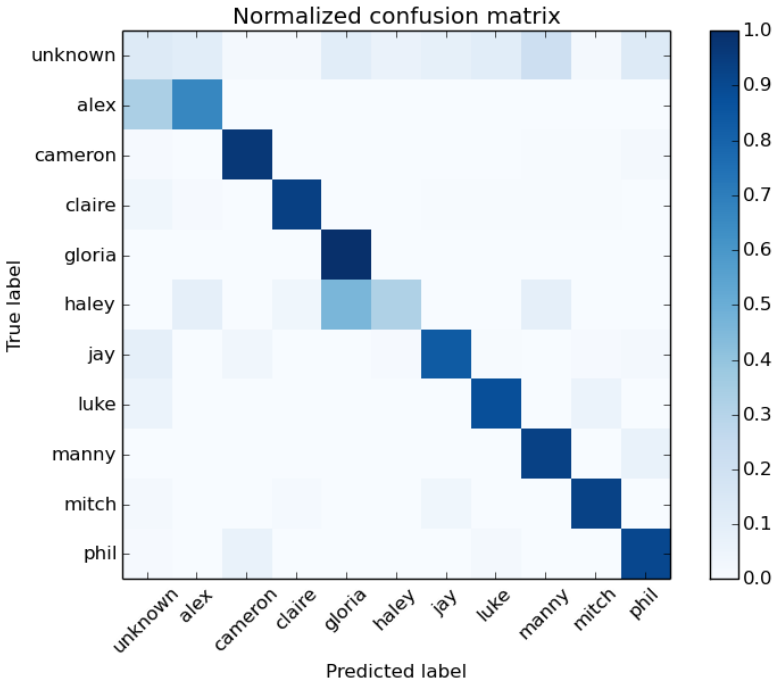
In Fig. 5.12(a) and Fig. 5.12(b), we show the confusion matrix for the semi-supervised interactive person recognition system when both training and testing bounding box are not available. As in Sec. 5.7.2, there are some columns that are completely blank due to the fact that those classes do not dominate any cluster or do not have a high enough dominant percentage.

5.8 Conclusion

In this chapter, we present a person detection and recognition system that can work in a barely constrained encironment. We propose to use head region instead of face or body as the key body part for person detection and recognition. A head detector based on the Faster R-CNN framework is trained and can handle various pose, viewpoints, and lighting conditions. To extract rich features around the head region, we train a deep CNN model for head recognition utilizing large scale external datasets. The detection and recognition pipeline is evaluated on a challenging TV series dataset and proves to be be effective in a simple supervised learning scenario. We further investigate several other scenarios where the amount of labeled data and the effort to label data is very limited. The results show that we can achieve very good results by using unlabelled data with minimum effort of annotation.



(a) NN using detected bounding box



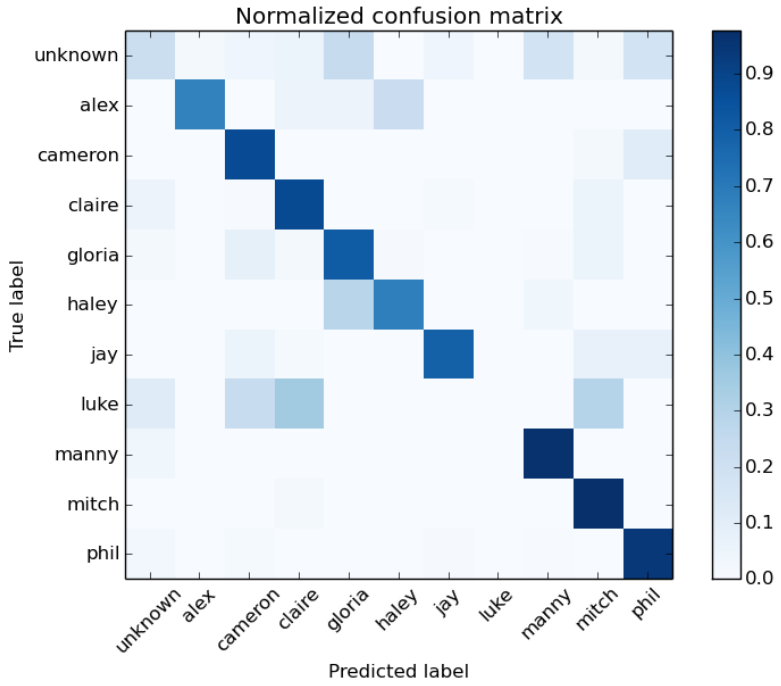
(b) SVM using detected bounding box

Fig. 5.9. Confusion matrix for person recognition with labelled data.

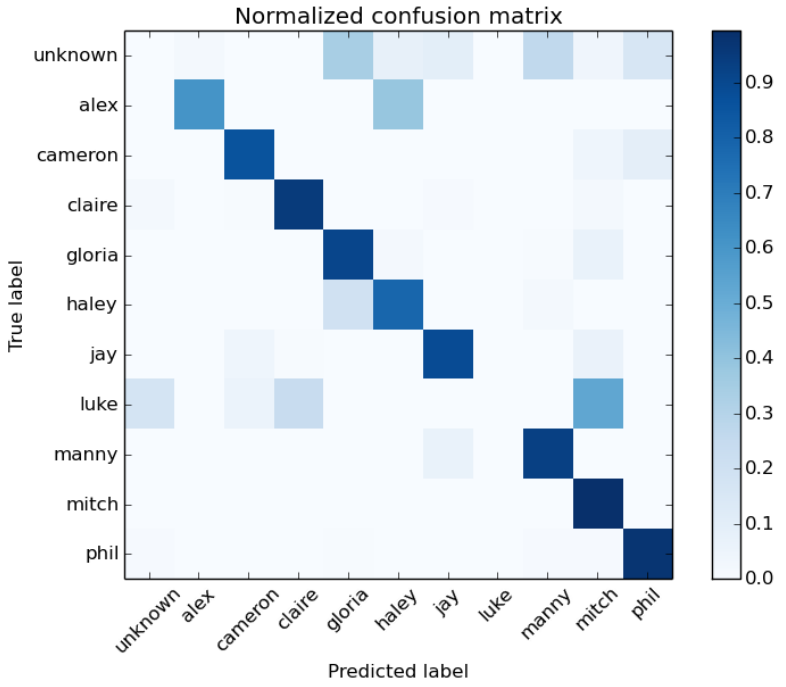
(a)

(b)

Fig. 5.10. Images from two of the 40 clusters

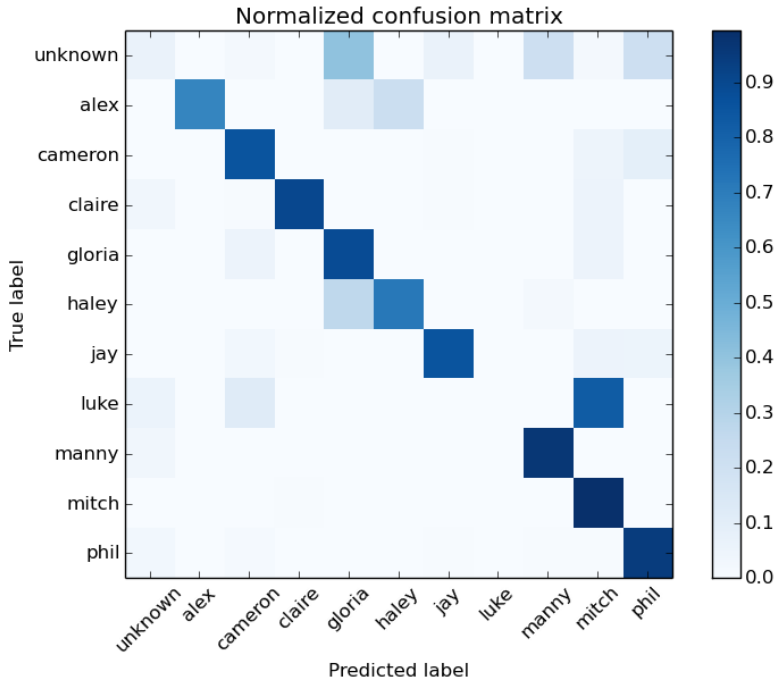


(a) NN using detected bounding box

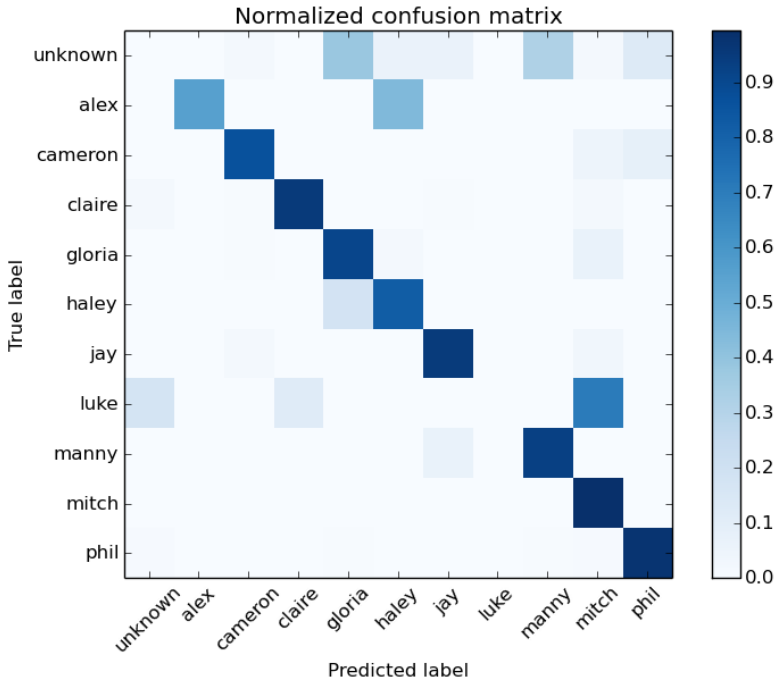


(b) SVM using detected bounding box

Fig. 5.11. Confusion matrix for the interactive person recognition system.



(a) NN using detected bounding box



(b) SVM using detected bounding box

Fig. 5.12. Confusion matrix for the semi-supervised interactive person recognition system.

6. CONCLUSION

In this dissertation, we investigated several high-level image analysis tasks and propose algorithms towards better understanding the image content.

In Chapter 2, we developed a system that can predict the aesthetic quality for photos of fashion products. We utilized global and generic features, salient object detection, compositional rules, and metadata together for aesthetic quality prediction. A database of manually rated photos specifically for photos of fashion products are constructed. In this chapter, we formulated aesthetic quality prediction as both classification and regression. The testing results showed that we can achieve good prediction accuracy using the designed feature sets.

In Chapter 3, we introduced Confidence Ordered Proposal (COP), a general method to improve the performance of region proposal-based multi-label object classification and later apply it to aesthetic attributes learning. The raw confidence score of each proposal is re-arranged to emphasis the relative importance of proposals with higher scores. This method can be applied to any proposal-based object classification framework. From the experimental results on the two benchmark datasets VOC 2007 and VOC 2012, we proved that our proposed method consistently outperforms existing proposal-based method that simply uses max-pooling. By utilizing the best proposal-based multi-label object classification framework Hypothesis-CNN-Pooling, our method achieves the state-of-the-art classification results on both datasets. Later, we briefly introduced our on-going research project of aesthetic attributes learning. We formulate aesthetic attributes learning as a multi-label classification problem. Generic features are used to train a set of classifiers for each aesthetic attributes. Then COP is used to refine the attributes classification result. The attributes that we learn can further serve as a mid-level image representation, which provide a more objective way of aesthetic quality evaluation.

In Chapter 4, we investigated another sub-problem of affective computing: image emotion classification. This is similar to aesthetic quality in the sense that both of them require very subjective evaluation. We utilize the recent development in deep convolutional neural network(CNN) to learn rich features for emotion classification. A multi-scale pooling method using CNN features is proposed to improve the previous best results.

In Chapter 5, we introduced a person detection and recognition that can work in an unconstrained environment. We propose to detect head region and train a head recognition model that does not require any alignment step as opposed to traditional approach based on face recognition. We further consider more realistic scenarios where the amount of labelled data and effort to label data is very limited. Through our semi-supervised learning-based method, we can achieve comparable or even higher results with the minimum amount of annotation.

To summarise, the major contribution of this work is listed below

- Aesthetic quality inference for fashion photos
  - propose novel features tailored for fashion aesthetics
  - construct a fashion photos dataset that are manually labelled with aesthetic ratings
- Confidence ordered proposals and applications in aesthetic attributes learning
  - propose a general method to boost multi-label classification performance
  - outperform previously existing methods on a benchmark dataset
- Image emotion classification
  - propose a multi-scale pooling method for CNN features
- Person detection and recognition in natural settings
  - perform head recognition training without the need for alignment

- collect a dataset of TV series that can be used to study person detection and recognition in natural settings
- extensively study person detection and recogniton in several realistic scenarios
- propose an interactive person recognition approach that requires minimum amount of annotation
- propose a semi-supervised interactive person recognition approach that requires minimum amount of annotation and can further improve the recognition performance with extra unlabelled data

REFERENCES

REFERENCES

[1] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Studying aesthetics in photographic images using a computational approach," in *2006 European Conference on Computer Vision*. Springer, 2006, pp. 288–301.

[2] Y. Ke, X. Tang, and F. Jing, "The design of high-level features for photo quality assessment," in *2006 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1. IEEE, 2006, pp. 419–426.

[3] Y. Luo and X. Tang, "Photo and video quality evaluation: Focusing on the subject," in *2008 European Conference on Computer Vision*. Springer, 2008, pp. 386–399.

[4] L.-K. Wong and K.-L. Low, "Saliency-enhanced image aesthetics class prediction," in *2009 IEEE International Conference on Image Processing*. IEEE, 2009, pp. 997–1000.

[5] L. Marchesotti, F. Perronnin, D. Larlus, and G. Csurka, "Assessing the aesthetic quality of photographs using generic image descriptors," in *2011 IEEE International Conference on Computer Vision*. IEEE, 2011, pp. 1784–1791.

[6] C. Li, A. Gallagher, A. C. Loui, and T. Chen, "Aesthetic quality assessment of consumer photos with faces," in *2010 IEEE International Conference on Image Processing*. IEEE, 2010, pp. 3221–3224.

[7] S.-F. Xue, H. Tang, D. Tretter, Q. Lin, and J. Allebach, "Feature design for aesthetic inference on photos with faces," in *2013 IEEE International Conference on Image Processing*. IEEE, 2013.

[8] W. Luo, X. Wang, and X. Tang, "Content-based photo quality assessment," in *2011 IEEE International Conference on Computer Vision*. IEEE, 2011, pp. 2206–2213.

[9] G. Pavlovic and A. M. Tekalp, "Maximum likelihood parametric blur identification based on a continuous spatial domain model," *IEEE Transactions on Image Processing*, vol. 1, no. 4, pp. 496–504, 1992.

[10] N. D. Narvekar and L. J. Karam, "A no-reference image blur metric based on the cumulative probability of blur detection (cpbd)," *IEEE Transactions on Image Processing*, vol. 20, no. 9, pp. 2678–2683, 2011.

[11] D. Hasler and S. E. Süsstrunk, "Measuring colorfulness in natural images," in *Proc. SPIE 5007, Human Vision and Electronic Imaging VIII*. SPIE, 2003, pp. 87–95.

[12] L. Itti, C. Koch, E. Niebur *et al.*, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[13] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 853–860.

[14] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[15] D. Cohen-Or, O. Sorkine, R. Gal, T. Leyvand, and Y.-Q. Xu, "Color harmonization," *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 624–630, 2006.

[16] Y. Matsuda, "Color design," *Asakura Shoten*, vol. 2, no. 4, 1995.

[17] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[18] W. Jiang, A. C. Loui, and C. D. Cerosaletti, "Automatic aesthetic value assessment in photographic images," in *2010 IEEE International Conference on Multimedia and Exp*. IEEE, 2010, pp. 920–925.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 248–255.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*. MIT Press, 2012, pp. 1097–1105.

[21] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[22] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *2014 European Conference on Computer Vision*. Springer, 2014, pp. 818–833.

[23] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013.

[24] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[25] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan, "Cnn: Single-label to multi-label," *arXiv preprint arXiv:1406.5726*, 2014.

[26] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial intelligence*, vol. 89, no. 1, pp. 31–71, 1997.

[27] Z.-H. Zhou and M.-L. Zhang, "Multi-instance multi-label learning with application to scene classification," in *Advances in Neural Information Processing Systems 19*. MIT Press, 2006, pp. 1609–1616.

[28] O. Yakhnenko and V. Honavar, "Multi-instance multi-label learning for image classification with large vocabularies," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2011, pp. 59.1–59.12.

[29] Y. Hu, M. Li, and N. Yu, "Multiple-instance ranking: Learning to rank images for image retrieval," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[30] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos, "Supervised learning of semantic classes for image annotation and retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 394–410, 2007.

[31] Y. Chen and J. Z. Wang, "Image categorization by learning and reasoning with regions," *The Journal of Machine Learning Research*, vol. 5, pp. 913–939, 2004.

[32] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Advances in Neural Information Processing Systems 15*. MIT Press, 2002, pp. 561–568.

[33] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

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

[35] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[36] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *2010 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 3360–3367.

[37] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *2010 European Conference on Computer Vision*. Springer, 2010, pp. 143–156.

[38] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *2006 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2006, pp. 2169–2178.

[39] Q. Chen, Z. Song, Y. Hua, Z. Huang, and S. Yan, "Hierarchical matching with side information for image classification," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3426–3433.

[40] H. Harzallah, F. Jurie, and C. Schmid, "Combining efficient object localization and image classification," in *2009 IEEE International Conference on Computer Vision*. IEEE, 2009, pp. 237–244.

[41] Z. Song, Q. Chen, Z. Huang, Y. Hua, and S. Yan, "Contextualizing object detection and classification," in *2011 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 1585–1592.

[42] O. Russakovsky, Y. Lin, K. Yu, and L. Fei-Fei, "Object-centric spatial pooling for image classification," in *2012 European Conference on Computer Vision*. Springer, 2012, pp. 1–15.

[43] M. Ranzato, F. J. Huang, Y.-L. Boureau, and Y. LeCun, "Unsupervised learning of invariant feature hierarchies with applications to object recognition," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *arXiv preprint arXiv:1406.4729*, 2014.

[45] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[46] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2014, pp. 512–519.

[47] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," *arXiv preprint arXiv:1310.1531*, 2013.

[48] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 1717–1724.

[49] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 580–587.

[50] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *2014 European Conference on Computer Vision*. Springer, 2014, pp. 392–407.

[51] J. B. Tenenbaum and W. T. Freeman, "Separating style and content with bilinear models," *Neural computation*, vol. 12, no. 6, pp. 1247–1283, 2000.

[52] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Bilinear classifiers for visual recognition," in *Advances in Neural Information Processing Systems 22*. MIT Press, 2009, pp. 1482–1490.

[53] X. Tan, Y. Li, J. Liu, and L. Jiang, "Face liveness detection from a single image with sparse low rank bilinear discriminative model," in *2010 European Conference on Computer Vision*. Springer, 2010, pp. 504–517.

[54] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *2014 European Conference on Computer Vision*. Springer, 2014, pp. 391–405.

[55] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2189–2202, 2012.

[56] E. Rahtu, J. Kannala, and M. Blaschko, "Learning a category independent object detection cascade," in *2011 IEEE International Conference on Computer Vision*. IEEE, 2011, pp. 1052–1059.

[57] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "Bing: Binarized normed gradients for objectness estimation at 300fps," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 3286–3293.

[58] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[59] S. Manen, M. Guillaumin, and L. V. Gool, "Prime object proposals with randomized prim's algorithm," in *2013 IEEE International Conference on Computer Vision*. IEEE, 2013, pp. 2536–2543.

[60] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[61] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *2014 ACM International Conference on Multimedia*. ACM, 2014, pp. 675–678.

[62] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[63] J. Dong, W. Xia, Q. Chen, J. Feng, Z. Huang, and S. Yan, "Subcategory-aware object classification," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013, pp. 827–834.

[64] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.

[65] S. Yan, J. Dong, Q. Chen, Z. Song, Y. Pan, W. Xia, Z. Huang, Y. Hua, and S. Shen, "Generalized hierarchical matching for subcategory aware object classification," in *2012 European Conference on Computer Vision*. Springer, 2012.

[66] M. Oquab, L. Bottou, I. Laptev, J. Sivic *et al.*, "Weakly supervised object recognition with convolutional neural networks," in *2015 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2015.

[67] N. Murray, L. Marchesotti, and F. Perronnin, "Ava: A large-scale database for aesthetic visual analysis," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2408–2415.

[68] L. Marchesotti, F. Perronnin, and F. Meylan, "Learning beautiful (and ugly) attributes." in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2013, pp. 7.1–7.11.

[69] D. Joshi, R. Datta, E. Fedorovskaya, Q.-T. Luong, J. Z. Wang, J. Li, and J. Luo, "Aesthetics and emotions in images," *Signal Processing Magazine, IEEE*, vol. 28, no. 5, pp. 94–115, 2011.

[70] W. Wei-ning, Y. Ying-lin, and J. Sheng-ming, "Image retrieval by emotional semantics: A study of emotional space and feature extraction," in *2006 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, 2006, pp. 3534–3539.

[71] J. Machajdik and A. Hanbury, "Affective image classification using features inspired by psychology and art theory," in *2010 ACM International Conference on Multimedia*. ACM, 2010, pp. 83–92.

[72] J. Jia, S. Wu, X. Wang, P. Hu, L. Cai, and J. Tang, "Can we understand van gogh's mood?: learning to infer affects from images in social networks," in *2012 ACM International Conference on Multimedia*. ACM, 2012, pp. 857–860.

[73] R. Datta, J. Li, and J. Z. Wang, "Algorithmic inferencing of aesthetics and emotion in natural images: An exposition," in *2008 IEEE International Conference on Image Processing*. IEEE, 2008, pp. 105–108.

[74] S. Zhao, Y. Gao, X. Jiang, H. Yao, T.-S. Chua, and X. Sun, "Exploring principles-of-art features for image emotion recognition," in *2014 ACM International Conference on Multimedia*. ACM, 2014, pp. 47–56.

[75] S. Zhao, H. Yao, Y. Yang, and Y. Zhang, "Affective image retrieval via multi-graph learning," in *2014 ACM International Conference on Multimedia*. ACM, 2014, pp. 1025–1028.

[76] S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, and H. Winnemoeller, "Recognizing image style," *arXiv preprint arXiv:1311.3715*, 2013.

[77] S. Dhar, V. Ordonez, and T. L. Berg, "High level describable attributes for predicting aesthetics and interestingness," in *2011 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 1657–1664.

[78] J. Ali, "Quantifying aesthetics of visual design applied to automatic design," Ph.D. dissertation, Purdue University, 2014.

[79] M. Chen and J. Allebach, "Aesthetic quality inference for online fashion shopping," in *Proc. SPIE 9027, Imaging and Multimedia Analytics in a Web and Mobile World*. SPIE, 2014.

[80] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *2001 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2001, pp. I–511.

[81] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool, "Face detection without bells and whistles," in *2014 European Conference on Computer Vision*. Springer, 2014, pp. 720–735.

[82] R. Girshick, "Fast r-cnn," in *2015 IEEE International Conference on Computer Vision*. IEEE, 2015, pp. 1440–1448.

[83] 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 28*. MIT Press, 2015, pp. 91–99.

[84] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[85] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 1701–1708.

[86] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2015, pp. 815–823.

[87] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2015, pp. 41.1–41.12.

[88] M. Everingham, J. Sivic, and A. Zisserman, "Hello! my name is... buffy"–automatic naming of characters in tv video." in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2006, pp. 92.1–92.10.

[89] M. Tapaswi, M. Bäuml, and R. Stiefelhagen, "knock! knock! who is it? probabilistic person identification in tv-series," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2658–2665.

[90] B. Prosser, W.-S. Zheng, S. Gong, T. Xiang, and Q. Mary, "Person re-identification by support vector ranking," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2010, pp. 21.1–21.11.

[91] W. Li, R. Zhao, and X. Wang, "Human reidentification with transferred metric learning," in *2012 Asian Conference on Computer Vision*. Springer, 2012, pp. 31–44.

[92] R. Zhao, W. Ouyang, and X. Wang, "Person re-identification by salience matching," in *2013 IEEE International Conference on Computer Vision*. IEEE, 2013, pp. 2528–2535.

[93] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 152–159.

[94] D. Yi, Z. Lei, and S. Z. Li, "Deep metric learning for practical person re-identification," *arXiv preprint arXiv:1407.4979*, 2014.

[95] N. Zhang, M. Paluri, Y. Taigman, R. Fergus, and L. Bourdev, "Beyond frontal faces: Improving person recognition using multiple cues," in *2015 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2015, pp. 4804–4813.

[96] S. Joon Oh, R. Benenson, M. Fritz, and B. Schiele, "Person recognition in personal photo collections," in *2015 IEEE International Conference on Computer Vision*. IEEE, 2015, pp. 3862–3870.

[97] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 304–311.

[98] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "Ms-celeb-1m: A dataset and benchmark for large-scale face recognition," in *2016 European Conference on Computer Vision*. Springer, 2016, pp. 87–102.

[99] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *arXiv preprint arXiv:1411.7923*, 2014.

[100] D. Miller, E. Brossard, S. Seitz, and I. Kemelmacher-Shlizerman, "Megaface: A million faces for recognition at scale," *arXiv preprint arXiv:1505.02108*, 2015.

[101] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2879–2886.

[102] V. Jain and E. G. Learned-Miller, "Fddb: A benchmark for face detection in unconstrained settings," *UMass Amherst Technical Report*, 2010.

[103] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, 2007.

[104] D. Gray and H. Tao, "Viewpoint invariant pedestrian recognition with an ensemble of localized features," in *2008 European Conference on Computer Vision*. Springer, 2008, pp. 262–275.

[105] C. Vondrick, D. Patterson, and D. Ramanan, "Efficiently scaling up crowdsourced video annotation," *International Journal of Computer Vision*, vol. 101, no. 1, pp. 184–204, 2013.

VITA

VITA

Ming Chen received his B.Eng degree in Electrical and Computer Engineering from the Hong Kong University of Science and Technology, Hong Kong, May 2011. He is currently pursuing his Ph.D. degree in Electrical and Computer Engineering at Purdue University. His current research interest includes digital image processing, multimedia analysis, computer vision, and machine learning.