Distinctive Parts for Relative attributes

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of science(by research)
in
Computer Science Engineering

by

Ramachandruni Naga Sandeep
201207582
nsandeep.ramachandruni@research.iiit.ac.in

Center for Visual Information Technology
International Institute of Information Technology
Hyderabad - 500 032, INDIA
December 2014

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "Distinctive parts for Relative attributes" by Ramachandruni Naga Sandeep, has been carried out under my supervision and is not submitted elsewhere for a degree.

_____ _____

Date Adviser: Prof. C.V. Jawahar

To My parents

Acknowledgments

I would like to first thank my advisor Prof.C.V. Jawahar for guidance, support and encouragement, without which this would not have been possible. Many thanks to Yashaswi Verma for his invaluable support during this work. It would not have been the same without him.

A special thanks to Prof.P.J. Narayanan, Dr.Anoop Namboodri, Prof.Jayanthi Sivaswamy for teaching me the courses Computer Vision, Statistical methods in AI, Image Processing and also for creating wonderful research atmosphere in CVIT and Mr.Satya for the administration work, and Rajan, Phani, Nandini for the annotation work.

It was a great fun and learning experience to be part of CVIT. I would like to thank all the friends at CVIT for the useful discussions Nataraj, Anand Mishra, Devender, Vidyadhar, Vijay, Aniketh, viresh, Praveen, Jaypanda. I would also like to thank my friends Pramod, Venky, vamsi, Pratyush, santosh, Panem, Kaustav, Falak, Mohak for all the fun, food we had and cricket we played without which it would have been very boring. Thanks for my friend Vamsi subhash for his continuous encouragement for learning.

Most importantly, I would like to thank my family for supporting me always, especially my mother for her love, father for his motivation and my brother for all the time we had been together.

Abstract

Visual Attributes are properties observable in images that have human-designated names (e.g., smiling, natural) and they are valuable as a new semantic cue in various vision problems like facial verification, object recognition, generating description of unfamiliar objects and to facilitate zero shot transfer learning etc. While most of the work on attributes focuses on binary attributes (indicating the presence or absence of attribute) the notion of relative attributes as introduced by Parikh and Grauman in ICCV 2011 provides an appealing way of comparing two images based on their visual properties than the binary attributes. Relative visual properties are a semantically rich way by which humans describe and compare objects in the world. They are necessary, for instance, to refine an identifying description (the rounder pillow; the same except bluer), or to situate with respect to reference objects (brighter than a candle; dimmer than a flashlight). Furthermore, they have potential to enhance active and interactive learning, for instance, offering a better guide for a visual search (find me similar shoes, but shinier or refine the retrieved images of downtown Chicago to those taken on sunnier days). For learning relative attributes a ranking svm based formulation was proposed that uses globally represented pairs of annotated images. In this thesis, we extend this idea towards learning relative attributes using local parts that are shared across categories.

First we propose a part based representation that jointly represents a pair of images. For facial attributes, part corresponds to a block around a landmark point detected using a domain specific method. This representation explicitly encodes correspondences among parts, thus better capturing minute differences in parts that make an attribute more prominent in one image than another as compared to global representation. Next we update this part based representation by additionally learning weights corresponding to each part that denote their contribution towards predicting the strength of a given attribute. We call these weights as significance coefficients of parts. For each attribute the significance coefficients are learned in a discriminative manner simultaneously with a max-margin ranking model. Thus the best parts for predicting relative attribute more smiling will be different from those from predicting more eyes open. We compare the baseline method of Parikh and Grauman with the proposed method under various settings. We have collected a new dataset of 10000 pair wise attribute level annotations using images from labeled faces in the wild (LFW) dataset particularly focusing on large variety of samples in terms of poses, lightning conditions etc and completely ignoring the category information while collecting attribute annotation . Extensive experiments demonstrate that the new method significantly improves prediction accuracy as compared to the baseline method. Moreover the learned parts

also compare favorably with human selected parts, thus indicating the intrinsic capacity of the proposed framework for learning attribute specific semantic parts. Additionally we illustrate the advantage of the proposed method with interactive image search using relative attribute based feedback.

In this work, we also propose *relational attributes*, which provide a more natural way of comparing two images based on some given attribute than relative attributes. Relational attributes consider not only the content of a given pair of images, but also take into account its relationship with other pairs, thus making the comparison more robust.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Visual attributes are human interpretable mid level visual concepts shareable across categories eg: furry, smiling and masculine. These are often used in multimedia community to build intermediate representations of images. Attributes have also been gaining a lot of attention in computer vision community over the past few years [2, 6, 21, 17, 29, 30, 40]. Attributes are used in face verification, classification, zero shot learning, object detection, image retrieval, interactive visual search, generating natural language descriptions of images, fine grained recognition etc. Attributes has also used as a mode of communication for the human supervisor to provide an actively learning machine classifier feedback when it predicts an incorrect label for an image. Attributes are used to enhance the mode of communication between humans and machine to improve visual recognition. Automatic learning and recognition of attributes can complement category level recognition and therefore improve the degree to which machines perceive visual objects. Attributes also opens door to appealing applications such as more specific queries in image search engines. Prior work on attributes focuses only on binary attributes indicating the presence or absence of certain visual property in the image.

For a large variety of attributes this binary setting is restrictive but it is also unnatural. For instance it is not clear in the Figure 1.2 (b) whether Sehwag is smiling or not. Different people are likely to respond inconsistently in providing the presence or absence of attribute smiling attribute for this image or for the natural attribute for image.

(a) (b)

Figure 1.1: Examples images containing a) attribute Red b) attribute Striped

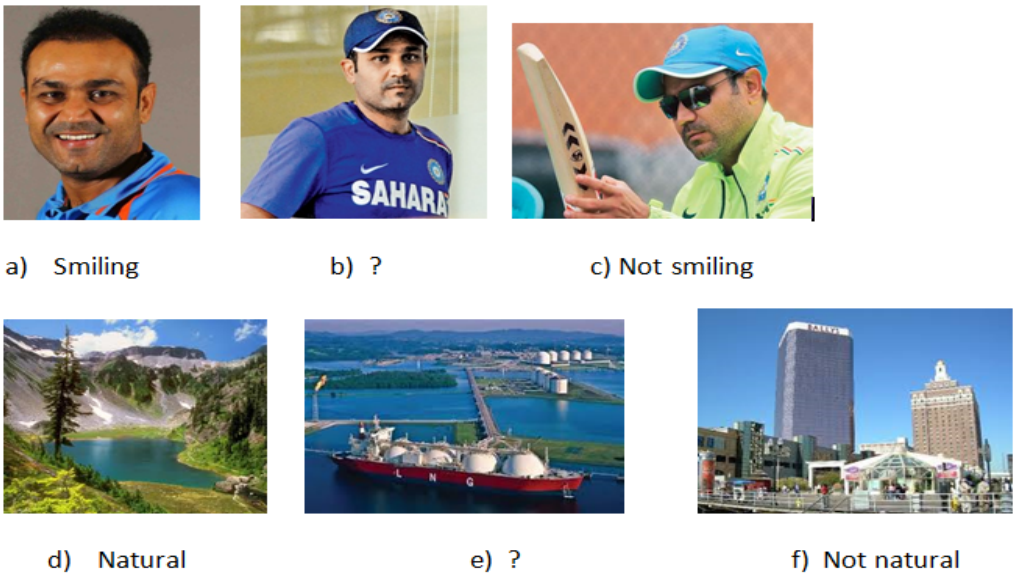
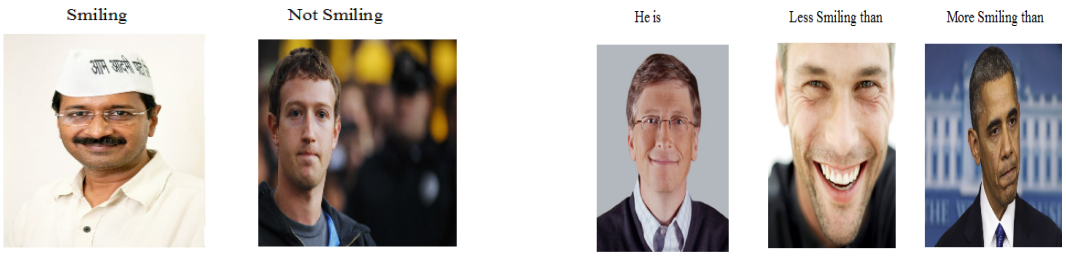


Figure 1.2: Binary attributes are an artificially restrictive way to describe images. While it is clear that (a) is smiling, and (c) is not, the more informative and intuitive description for (b) is via relative attributes: he is smiling more than (a) but less than (c). Similarly, scene (e) is less natural than (d), but more so than (f).

1.1 Relative Attributes

The notion of relative attributes as proposed by Parikh and Grauman in ICCV 2011, provides an appealing way of comparing two images based on their visual properties. Instead of predicting presence of an attribute, relative attributes indicate the strength of an attribute with respect to other images. In addition to being more natural, relative attributes would offer a richer mode of communication, thus allowing access to more detailed human supervision, as well as the ability to generate more informative descriptions of novel images. Given a set of pairs of images depicting similar and/or different strengths of particular attribute, the problem of learning relative attribute classifier is posed as one of learning a ranking model for that attribute similar to ranking SVM. Prior works on relative attributes uses global feature representation like gist and color histograms. Existing datasets for this task in facial domain is pubfig-29 with 29 attributes and 60 categories. The annotations for this dataset are collected at categorical level using pair of categories rather than pair of images. Many applications of relative attributes has been explored. Relative attributes based feedback is used to perform image search.



(a) (b)

Figure 1.3: Part a) explains the use of binary attributes b) explains the use of relative attributes for smiling attribute

1.2 Scope of this thesis

1.2.1 Problem and contributions

In this Thesis, we focus on solving couple of problems related to Relative Attributes. In the basic form, given two images we need to predict which image has more strength of given attribute than the other. To be specific, we are interested in (i) learning a part based decription which can compare two images and provide semantically meaningful comparison of the images.

- We build upon the idea of learning relative attributes models using local parts that are shared across categories. We develop a method for a) Part based representation of images b) learning significance coefficients for each part c) Predicting relative attribute for pair of images.
- To validate our approach of comparing at the level of individual instances rather than categories, we have introduced a new dataset for this task in facial domain called LFW-10. Compared to the recent methods for relative attributes, our method achieves a significant improvement in accuracy.
- We learn the significance coefficients for each part, which reflects the semantic interpretation of attribute. Additionally our method shows improvement in relative feedback based interactive image search.
- We also propose *relational attributes*, which provide a more natural way of comparing two images based on some given attribute than relative attributes.

1.2.2 Significance of this work

This thesis extends the state of the art in relative attributes. This class of work has achieved significance in recent years due to its capacity to bridge the semantic levels of reasoning in humans and machines. This was recognized by many researchers and the Marr award of 2011 was given to appreciate the importance of this direction to the work of Parikh and Grauman [41] appeared in ICCV 2011.

In the last 3-4 years, there has been many extensions and improvements of this work, and our work is one of them.

1.2.3 Challenges

Like many other computer vision tasks, the task of predicting the relative strengths of the attributes is also challenging. This is specially challenging when the attributes are specific to an instance or that describes some of the fine variations within a category. Lack of reliable data for training and testing models is a fundamental problem for any computer vision algorithms that use machine learning in the back end. Additionally, variations in the appearance of the object due to occlusions, variations in imaging conditions, inconsistency in annotation pose non trivial challenges for this problem.

In this thesis, we have been using the face images as a worked out example for demonstrating the potential of the proposed algorithms. In this context, there are many additional challenges related to representation and imaging.

1.2.3.1 Dataset Creation

Although there are some existing datasets like pubfig-29, with 29 attributes and 60 categories, the annotations are collected at category level using pairs of categories rather than pairs of images. Due to this annotation in this data set are not consistent for several attributes. To address this limitation we have collected a new dataset called LFW-10. While collecting the dataset we completely ignore category information, thus making it particularly suitable for the task of learning relative attribute. In order to minimize the chances of inconsistencies in the dataset, each image pair is annotated from 5 trained annotators and final annotation is decided on majority voting.

1.2.3.2 Feature representation

The main challenge for any problem in computer vision is to have a good feature representation. There is always a debate on good feature representation in the community. The answer depends on the particular type of problem. Prior work on attributes uses global level features. We believe that part based representation which uses features extracted from different parts is good representation for the problem as some attributes are local in nature. So we have extracted parts from each image and then represented each image using the features extracted from each part.

1.2.3.3 Occlusion and Pose

Occlusion, pose variations and background clutter can cause similar effect to that of deformation. They alter shape, make parts disappear and add noise to the data. The problem is of particular importance to the attribute because people often tend to appear in different poses and some parts of them may

be occluded. This makes the problem difficult because we cannot extract relevant features responsible for that attribute.

1.3 Related Works

Here, we attempt to provide a brief review of the related works in the broad area of relative attributes. More detailed and specific literature is discussed and compared in the next two chapters.

Apart from predicting the presence or absence or finding relative strength of attribute, attributes have been used extensively, especially in the past few years, for a variety of applications [2, 6, 21, 17, 29, 30, 40]. In [18] ,objects are described using their attributes; e.g., instead of classifying an image as that of a sheep , it is described based on its properties such as has horn, has wool, etc. Attributes are also used in fine grain classification tasks [15] such as classification of birds. Color attributes are used for real time visual tracking [13]. Attributes are used for semantic segmentation of images [56]. Attributes are also used in image search where user provides feedback using attributes [29]. Attribute based feedback has been shown to be useful for anomaly detection within a object category [17] and adding unlabeled samples for category classifier learning [9]. Attributes have also been used for multiple query image search [48] . Attribute are also used in predicting user annoyance [10]. Multi-task learning which uses structured scarcity by jointly learning decorrelated and discriminative attributes [26]. A set wise active learning approach for learning relative attributes has been proposed to minimize the effort to gather relative attribute comparisons. Attributes have been used to learn and evaluate models of deeper scene understanding [17] that reason about properties of objects as opposed to just the object categories. They have also been used for alleviating annotation efforts via zero-shot learning [32, 41] where a supervisor can teach a machine a novel concept simply by describing its properties (e.g. a zebra is striped and has four legs or a zebra has a shorter neck than a giraffe). Attributes being both machine detectable and human understandable provide a mode of communication between the two. This has been exploited for improved image search by using attributes as keywords [30] or as interactive feedback [29], or for more effective active learning by allowing the supervisor to provide attribute-based feedback to a classifier [43], or even at test time with a humanin- the-loop answering relevant questions about the test image [6]. This idea is extended in [3] where the learner learns attribute classifiers along with category classifiers. In [47], a semi-supervised constrained bootstrapping approach is proposed that tries to benefit from inter-class attribute-based relationships to avoid semantic drift during the learning process. In [51], a novel framework for predicting relative dominance among attributes within an image is proposed. A Unified Probabilistic Approach was proposed for Modeling Relationships between Attributes and Objects [53]. Uncertain attributes are used for describing people [45]. In [46], rather than using either binary or relative attributes, their interactions are modeled to better describe images.Relative attributes are used for large scale abandoned object detection [16]. A local learning approach for fine grained visual comparisons was proposed which uses mahalanobis metric to find similarity between pairs [55]. Fine grained attributes are used for understanding objects in detail [52]. Our work closely

relates with recent works [1, 14, 15, 28] that use distinctive part/region-based representations for scene classification [28] or fine-grained classification [1, 14, 15]. However, rather than identifying category-specific distinctive parts, our aim is to compare similar parts that are shared across categories. This makes our problem somewhat more challenging, since our representation is expected to capture small relative differences in the appearance of semantically similar parts, which contribute in making some attribute prominent in one image than another.

1.4 Thesis outline

The next two chapters (chapters 2 and 3) provide a self contained description of our contributions on two sub problems in the related area of relative attibutes. Some of the basic ideas related to machine learning and representation, is provided in the appendix as an additional resource for those who are not familiar with the area. Brief outline of text in this thesis is as follows, in chapter 2, we explain our method of part based approach for learning relative attributes. In chapter 3, we explain our method of learning relational attributes. Finaly, we end the thesis with conclusion remarks and future work.

Chapter 2

Distinctive Parts for Relative Attribute Prediction for Facial Attributes

2.1 Introduction

In Relative Attributes [41], given a set of pairs of images depicting similar and/or different strengths of some particular attribute, the problem of learning a relative attribute classifier is posed as one of learning a ranking model for that attribute similar to Ranking SVM [27].

In this work, we build-upon this idea by learning relative attribute models using local parts that are shared across categories. First, we propose a part-based representation that jointly represents a pair of images. A part corresponds to a block around a landmark point detected using a domain-specific method. This representation explicitly encodes correspondences among parts, thus better capturing minute differences in parts that make an attribute more dominant in one image than other, as compared to a global representation as in [41]. Next, we update this part-based representation by additionally learning weights corresponding to each part that denote their contribution towards predicting the strength of a given attribute. We call these weights as *"significance-coefficients"* of parts. For each attribute, the significance-coefficients are learned in a discriminative manner simultaneously with a max-margin ranking model. Thus, the best parts for predicting the relative attribute "more smiling" will be different from those for predicting "more eyes-open". The steps of the proposed method are illustrated in Figure 2.1. While the notion of parts is not new, we believe that ours is the first attempt that explores the applicability of parts in a ranking scenario, for learning relative attribute ranking models in particular.

We compare the baseline method of [41] with the proposed method under various settings. For this, we have collected a new data set with around $11,000$ pair-wise attribute-level annotations using images from the "Labeled Faces in the Wild" (LFW) data set [24], particularly focusing on (i) large variety among samples in terms of poses, lighting condition, etc., and (ii) completely ignoring the category information while collecting attribute annotations. Extensive experiments demonstrate that the new method significantly improves the prediction accuracy as compared to the baseline method. Moreover, the learned parts also compare favorably with human-selected parts, thus indicating the intrinsic capacity of the proposed framework for learning attribute-specific semantic parts.

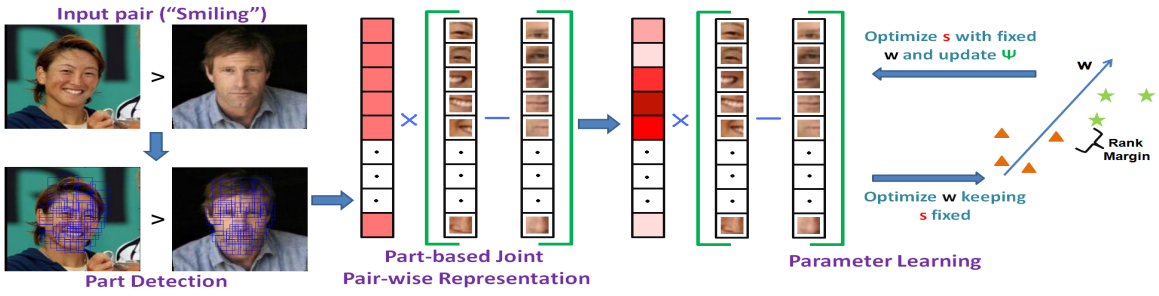


Figure 2.1: Given ordered pair of images, first we detect parts corresponding to different (facial) landmarks. Using these, a joint pair-wise part-based representation is formed that encodes (i) correspondence among different parts, and (ii) relative importance of each part with respect to a given attribute. Using this, a max-margin ranking model $\mathbf{w}$ is learned simultaneously with part weights in an iterative manner.

The chapter is organized as follows. In Sec. 2.2, we discuss the earlier method of [41] for learning relative attribute ranking models. Then we present the new part-based representations in Sec. 2.3, followed by an algorithm for learning model variables in Sec. 2.4. Experiments and results are discussed in Sec. 2.5.

2.2 Preliminaries

In [41], a Ranking SVM based method was used for learning relative attribute classifiers. Ranking SVM [27] is a max-margin ranking framework that learns linear models to perform pair-wise comparisons. This is conceptually different from the conventional one-vs-rest SVM that learns a model using individual samples rather than pairs. Though SVM scores can also be used to perform pair-wise comparisons, usually Ranking SVM has been known to perform better than SVM for such tasks. In [41] also, Ranking SVM was shown to perform better than SVM on the task of relative attribute prediction. We now briefly discuss the method used in [41] for learning relative attribute classifiers.

2.2.1 The Ranking SVM Model

Let $\mathcal{I} = \{I_1, \ldots, I_n\}$ be a collection of n images. Each image I_i is represented by a global feature vector $\mathbf{x}_i \in \mathcal{R}^N$. Suppose we have a fixed set of attributes $A = \{a_m\}$. For each attribute $a_m \in A$, we are given a set $\mathcal{D}_m = \mathcal{O}_m \cup \mathcal{S}_m$ consisting of ordered pairs of images. Here, $\mathcal{O}_m = \{(I_i, I_j)\}$ is such that image I_i has more strength of attribute a_m than image I_j. And, $\mathcal{S}_m = \{(I_i, I_j)\}$ is such that both I_i and I_j have nearly the same strength of attribute a_m. Using $\mathcal{D}_m$, the goal is to learn a ranking function f_m that, given a new pair of images I_p and I_q represented by $\mathbf{x}_p$ and $\mathbf{x}_q$ respectively, predicts which image has greater strength of attribute a_m. Under the assumption that f_m is a linear function of $\mathbf{x}_p$ and

$\mathbf{x}_q$, it is defined as:

$$f_m(\mathbf{x}_p, \mathbf{x}_q; \mathbf{w}_m) = \mathbf{w}_m \cdot \Psi(\mathbf{x}_p, \mathbf{x}_q), \tag{2.1}$$

$$\Psi(\mathbf{x}_p, \mathbf{x}_q) = \mathbf{x}_p - \mathbf{x}_q \tag{2.2}$$

Here, $\mathbf{w}_m$ is the parameter vector for attribute a_m, and $\Psi(\mathbf{x}_p, \mathbf{x}_q)$ is a joint representation formed using $\mathbf{x}_p$ and $\mathbf{x}_q$. Using f_m, we determine which image has higher strength for attribute a_m based on $y^m_{pq} = sign(f_m(\mathbf{x}_p, \mathbf{x}_q))$. $y^m_{pq} = 1$ means I_p has higher strength of a_m than I_q, and $y^m_{pq} = -1$ means otherwise. In order to learn $\mathbf{w}_m$, following constraints need to be satisfied:

$$\mathbf{w}_m \cdot \Psi(\mathbf{x}_i, \mathbf{x}_j) > 0 \;\; \forall (I_i, I_j) \in \mathcal{O}_m \tag{2.3}$$

$$\mathbf{w}_m \cdot \Psi(\mathbf{x}_i, \mathbf{x}_j) = 0 \;\; \forall (I_i, I_j) \in \mathcal{S}_m \tag{2.4}$$

Since this is an NP-hard problem, its relaxed version is solved by introducing slack variables. This leads to the following optimization problem ($OP1$):

$$OP1 : \min_{\mathbf{w}_m} \;\; \frac{1}{2}||\mathbf{w}_m||_2^2 + C_m(\sum \xi_{ij}^2 + \sum \alpha_{ij}^2) \tag{2.5}$$

$$s.t. \; \mathbf{w}_m \cdot \Psi(\mathbf{x}_i, \mathbf{x}_j) \geq 1 - \xi_{ij}, \; \forall (I_i, I_j) \in \mathcal{O}_m \tag{2.6}$$

$$||\mathbf{w}_m \cdot \Psi(\mathbf{x}_i, \mathbf{x}_j)||_1 \leq \alpha_{ij}, \; \forall (I_i, I_j) \in \mathcal{S}_m \tag{2.7}$$

$$\xi_{ij} \geq 0; \quad \alpha_{ij} \geq 0. \tag{2.8}$$

Here, $||\cdot||_2^2$ denotes squared L_2 norm, $||\cdot||_1$ denotes L_1 norm, and $C_m > 0$ is a constant that takes care of the trade-off between regularization term and loss term. Note that along with pair-wise constraints as in [27], the optimization problem now also includes similarity constraints. This is solved in the primal form itself using Newton's method [8].

2.3 Proposed Feature Representation

The Ranking SVM method discussed above uses a joint representation based on globally computed features (Eq. 2.2) while determining the strength of some given attribute. However, several attributes such as "visible-teeth", "eyes-open", etc. are not representative of whole image, and correspond to only some specific regions/parts. This means there exists a weak association between an image and its attribute label. E.g., Figure 2.2 shows the parts corresponding to attributes "visible-teeth" and "eyes-open". This inspires us to build a representation that (i) encodes part/region-specific features, without confusing across parts; and (ii) explicitly encodes the relative significance of each part with respect to a given attribute. With this motivation, next we propose two part-based joint-representations for the task of learning relative attribute classifiers.

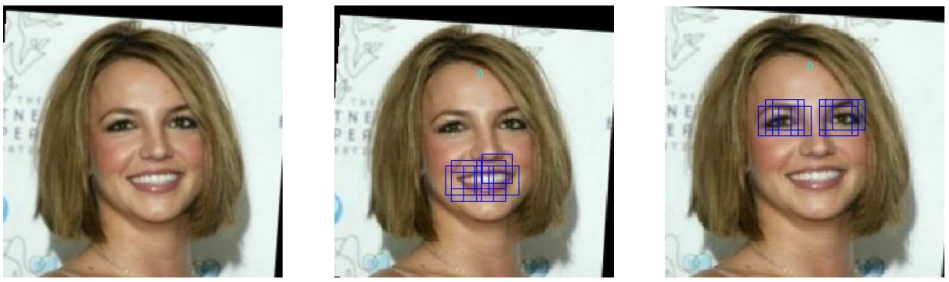


Figure 2.2: Given an input image (left), the parts that correspond to visible-teeth (middle) and eyes-open (right).

2.3.1 Part-based Joint Representation

Given an image I, let $\mathcal{P} = \{p^1, \ldots, p^K\}$ be the set of its K parts. These parts can be obtained using a domain-specific method; example the method discussed in [57] can be used for determining a set of localized parts in face images. Each part $p^k, \forall k \in \{1, \ldots, K\}$ is represented using an N_1-dimensional feature vector $\tilde{\mathbf{x}}^k \in \mathcal{R}^{N_1}$. Here, $N_1 = K \times d_1$ such that each $\tilde{\mathbf{x}}^k$ is a sparse vector with only d_1 non-zero entries in the k^{th} interval representing part p^k. Based on this, given a pair of images I_p and I_q, we define a joint part-based feature representation as below:

$$\tilde{\Psi}(\tilde{\mathbf{x}}_p, \tilde{\mathbf{x}}_q) = \sum_{k=1}^{K} (\tilde{\mathbf{x}}_p^k - \tilde{\mathbf{x}}_q^k), \tag{2.9}$$

where $\tilde{\mathbf{x}}_p = \{\tilde{\mathbf{x}}_p^k \mid \forall k \in \{1, \ldots, K\}\}$. The advantage of this representation is that it specifically encodes correspondence among parts; i.e., now the k^{th} part of I_p is compared with just the k^{th} part of I_q. The assumption here is that such a direct comparison between localized pairs of parts would provide stronger cues for learning relative attribute models than using a single global representation as in Eq. 2.2. (This assumption is also validated by improvements in prediction accuracy as discussed in Sec. 3.3.)

2.3.2 Weighted Part-based Joint Representation

Though the joint representation proposed in the previous section allows direct part-based comparison between a pair of images, it does not provide information about which parts actually symbolize some given attribute. This is particularly desirable in case of local attributes, where only a few parts are important in predicting attribute strength. With this motivation, we update the joint representation of Eq. 2.9 to precisely encode relative importance of parts.

As discussed in Sec. 2.3.1, let each image I be represented by a set of K parts. Additionally, let $s_m^k \in [0, 1]$ be a weight associated with the k^{th} part. This weight denotes the relative importance of the k^{th} part compared to other parts for predicting the strength of attribute a_m; i.e., larger the weight, more important is that part, and vice-versa. Using this, given a pair of images I_p and I_q, the new weighted

part-based joint feature representation is defined as:

$$\tilde{\Psi}_s(\tilde{\mathbf{x}}_p, \tilde{\mathbf{x}}_q, \mathbf{s}_m) = \sum_{k=1}^{K} s_m^k (\tilde{\mathbf{x}}_p^k - \tilde{\mathbf{x}}_q^k), \tag{2.10}$$

where $\mathbf{s}_m = [s_m^1, \ldots, s_m^K]^T$. Since s_m^k expresses the relative significance of the k^{th} part with respect to a_m, we call it as the significance-coefficient of the k^{th} part. These help in explicitly encoding the relative importance of individual parts in the joint representation.

2.4 Parameter Learning

Now we discuss how to learn the parameters for each attribute using the two joint representations discussed above. Note that we still need to satisfy the constraints as in Eq. 2.3 and Eq. 2.4 depending upon the representation followed.

2.4.1 For Part-based Joint Representation

In order to learn a ranking model based on the part-based representation in Eq. 2.9, we optimize the following problem:

$$OP2: \min_{\mathbf{w}_m} \frac{1}{2}||\mathbf{w}_m||_2^2 + C_m(\sum \xi_{ij}^2 + \sum \alpha_{ij}^2) \tag{2.11}$$

$$s.t.\ \mathbf{w}_m \cdot \tilde{\Psi}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j) \geq 1 - \xi_{ij},\ \forall (I_i, I_j) \in \mathcal{O}_m \tag{2.12}$$

$$||\mathbf{w}_m \cdot \tilde{\Psi}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j)||_1 \leq \alpha_{ij},\ \forall (I_i, I_j) \in \mathcal{S}_m \tag{2.13}$$

$$\xi_{ij} \geq 0; \quad \alpha_{ij} \geq 0. \tag{2.14}$$

This is similar to $OP1$, except that now we use part-based representation instead of global representation. This allows us to use the same Newton's method [8] for solving $OP2$.

2.4.2 For Weighted Part-based Joint Representation

For the weighted part-based joint representation in Eq. 2.10, we need to learn two sets of parameters corresponding to every attribute: ranking model $\mathbf{w}_m$, and significance-coefficients $\mathbf{s}_m$. To do this, we solve the following optimization problem ($OP3$):

$$OP3: \min_{\mathbf{w}_m, \mathbf{s}_m} \frac{1}{2}||\mathbf{w}_m||_2^2 + C_m(\sum \xi_{ij}^2 + \sum \alpha_{ij}^2) \tag{2.15}$$

$$s.t.\ \mathbf{w}_m \cdot \tilde{\Psi}_s(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j, \mathbf{s}_m) \geq 1 - \xi_{ij},\ \forall (I_i, I_j) \in \mathcal{O}_m \tag{2.16}$$

$$||\mathbf{w}_m \cdot \tilde{\Psi}_s(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j, \mathbf{s}_m)||_1 \leq \alpha_{ij},\ \forall (I_i, I_j) \in \mathcal{S}_m \tag{2.17}$$

$$\xi_{ij} \geq 0; \quad \alpha_{ij} \geq 0; \tag{2.18}$$

$$s_m^k \geq 0,\ \ \forall 1 \leq k \leq K; \quad \mathbf{e} \cdot \mathbf{s}_m = 1. \tag{2.19}$$

where $\mathbf{e} = [1, \ldots, 1]^T$ is a constant vector with all entries equal to 1. Note that the overall weight of all the parts is constrained to be unit; i.e., $s_m^k \geq 0$, $\mathbf{e} \cdot \mathbf{s}_m = 1$, which ensures that all parts are fairly used. This is equivalent to constraining the L_1-norm of $\mathbf{s}_m$ to be 1 (i.e., L_1-regularization), thus implicitly imposing sparsity on $\mathbf{s}_m$ [38, 50]. This is desirable since usually only a few parts contribute towards determining the strength of a given attribute.

2.4.2.1 Solving the optimization problem

We solve $OP3$ in the primal form itself using a block co-ordinate descent algorithm. We consider each set of parameters $\mathbf{w}_m$ and $\mathbf{s}_m$ as two blocks, and optimize them in an alternate manner. In the beginning, we initialize all entries of $\mathbf{w}_m$ to be zero, and all entries of $\mathbf{s}_m$ to be equal to $1/K$.

First we fix $\mathbf{s}_m$ to optimize $\mathbf{w}_m$. For a fixed $\mathbf{s}_m$, the problem becomes equivalent to $OP2$ (Eq. 2.11 to 2.14), and hence can be solved in the same manner using [8].

Then we fix $\mathbf{w}_m$ to optimize $\mathbf{s}_m$. Let $\tilde{\mathbf{X}}_i = [\tilde{\mathbf{x}}_i^1 \ldots \tilde{\mathbf{x}}_i^K] \in \mathcal{R}^{N_1 \times K}$ be a matrix formed by appending features corresponding to all parts of image I_i. Using this, we compute $\tilde{\mathbf{z}}_{im} = \tilde{\mathbf{X}}_i^T \mathbf{w}_m \in \mathcal{R}^K$. This gives

$$\mathbf{w}_m \cdot \tilde{\Psi}_s(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j, \mathbf{s}_m) = \mathbf{s}_m \cdot \tilde{\mathbf{z}}_{ijm}, \tag{2.20}$$

$$\tilde{\mathbf{z}}_{ijm} = \tilde{\mathbf{z}}_{im} - \tilde{\mathbf{z}}_{jm}. \tag{2.21}$$

Substituting this in $OP3$ leads to the following optimization problem for learning $\mathbf{s}_m$ (for fixed $\mathbf{w}_m$):

$$OP4: \min_{\mathbf{s}_m} \; C \; \Big(\sum_{(I_i, I_j) \in \mathcal{Q}_m} (1 - \mathbf{s}_m \cdot \tilde{\mathbf{z}}_{ijm})^2 \; + \sum_{(I_i, I_j) \in \mathcal{S}_m} ||\mathbf{s}_m \cdot \tilde{\mathbf{z}}_{ijm}||_1^2 \; \Big) \tag{2.22}$$

$$s.t. \quad s_m^k \geq 0, \;\; \forall 1 \leq k \leq K; \quad \mathbf{e} \cdot \mathbf{s}_m = 1. \tag{2.23}$$

where $\mathcal{Q}_m \subseteq \mathcal{O}_m$ is the set of pairs that violate the margin constraint. Note that $\mathcal{Q}_m$ is not fixed, and may change at every iteration. We solve $OP4$ using an iterative gradient descent and projection method similar to [54].

2.4.3 Computing Parts

The two joint representations as proposed in Sec. 2.3 are based on an ordered set of corresponding parts computed from a given pair of images. Given a method for computing such parts, our framework is applicable irrespective of the domain. This makes our framework domain independent.

In this work, we consider the domain of face images. To compute parts from a given face image, we use the method proposed in [57]. It is based on a mixture-of-tress model to learn a shared pool of facial parts. Given a face image, it computes a set of 68 parts covering facial landmarks such as eyes, eyebrows, nose, mouth and jawline. Figure 2.3 shows a face image (left) and its parts (middle)

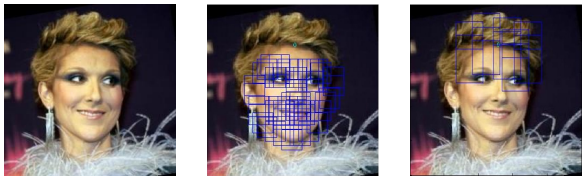


Figure 2.3: Input image (left), parts detected using [57] (middle), and additional parts detected by us (right).

Figure 2.4: Example pairs and their ground-truth annotations from Pubfig-29 data set. Due to category-level annotations, there exist inconsistencies in (true) instance-level attribute visibility.

computed using this method. Though these parts can be used to represent several attributes such as "smiling", "eyes-open", etc., there are few other attributes which are not covered by these parts such as "bald-head", "visible-forehead" and "dark-hair". In order to cover these attributes as well, we compute additional parts using image-level statistics such as image-size and distance from the earlier 68 parts. This gives an extended set of 83 parts for a given face image. Figure 2.3 (right) shows this extended set of parts computed for the given image on the left.

2.4.4 Relation with Latent Models

In the last few years, latent models have become popular for several tasks, particularly for object detection [19]. These models usually look for characteristics (e.g., parts) that are shared within a category but distinctive across categories. (Recent works such as [14, 1, 28] also have similar motivation, though they do not explicitly investigate the latent aspect.) Our work is similar to theirs in the sense that we also seek attribute-specific distinctive parts by incorporating significance-coefficients. However, in contrary to them, we require these parts to be shared across categories. This is because our ranking method uses these parts to learn attribute-specific models which are independent of categories being depicted in training pairs.

Figure 2.5: Example pairs from LFW-10 data set. The images exhibit high diversity in terms of age, pose, lighting, occulusion, etc.

2.5 Experiments

We compare the proposed method with that of [41] under different settings on two data sets. First is the PubFig-29 data set as used in [3]. It consists of 60 face categories and 29 attributes, with attribute annotations being collected at category-level; i.e., using pairs of categories rather than pairs of images. Due to this, the annotations in this data set are not consistent for several attributes (see Figure 2.4) ; e.g., Scarlett Johansson may not be smiling more than Hugh Laurie in all their images. To address this limitation, we have collected a new data set using a subset of the LFW [24] images. The new data set has attribute-level annotations for around $11,000$ image pairs and 10 attributes, and we call this as LFW-10 data set. While collecting the annotations, we particularly ignore the category information, thus making it more suitable for the task of learning relative attributes. The details of this data set are described next.

2.5.1 LFW-10 Data set

We randomly select 2000 images from the LFW data set [24]. Out of these, 1000 images are used for creating training pairs and the remaining (unseen) 1000 for testing pairs. The annotations are collected for 10 attributes, with at least 500 training and testing pairs per attribute. On an average, there are 557 training and 591 testing pairs per attribute. The annotations are collected using a reward based set-up, thus minimizing the chances of inconsistency in the data set. Figure 2.5 shows example pairs from this data set.

2.5.2 Features for Parts

We represent each part using a Bag of Visual Words (BoVW) histogram over dense SIFT (DSIFT) [36] features. We consider two settings for learning visual-word vocabulary: (1) In the first setting, we learn a part-specific vocabulary for every part. This is possible since our parts are fixed and known. In practice, we learn a vocabulary of 10 visual words for each part. This gives a 830-dimensional ($= 83$ parts $\times 10$) (sparse) feature vector per part. (2) In the second setting, we learn a single vocabulary for all the parts consisting of 100 visual words. This results in a 8300-dimensional (=83 parts $\times 100$) sparse vector representing each part.

Method	Accuracy
Global DSIFT + RSVM [41]	61.28
Global GIST + RGB + RSVM [41]	59.18
Unweighted parts + Part-specific vocab. (Ours)	62.54
Unweighted parts + Single vocab. (Ours)	62.83
Learned parts + Part-specific vocab. (Ours)	62.67
Learned parts + Single vocab. (Ours)	**63.08**

Table 2.1: Results on PubFig-29 data set. Though all the methods perform comparable, these results are not really indicative of their actual behaviour due to inconsistency in ground-truth annotations.

2.5.3 Baselines

We compare with the Ranking SVM method of [41] We use four features for comparison: (i) BoVW histogram over DSIFT features with 1000 visual words, (ii) global 512-dimensional GIST descriptor [39], (iii) global 512-dimensional GIST and 30-dimensional RGB histogram (which was also used in [41]), and (iv) spatial pyramid (SPM) [33] upto two and three levels using DSIFT features and the same vocabulary as in (i).

As another baseline, we compare the quality of our part-learning framework (Sec. 2.4.2) against human-selected parts. For this, we ask a human expert to select the subset of most representative parts corresponding to every attribute. For a given attribute a_m, all the selected parts are assigned equal weights and the remaining parts are assigned zero weight, and then a ranking model $\mathbf{w}_m$ is learned based on these part weights. The intuition behind this experiment is to analyze the trade-off between the performance obtained using manually selected parts and learned parts.

2.5.4 Results

Table 2.1 compares different methods on PubFig-29 data set. For each attribute, we consider 1500 training pairs from 40 classes, and 1500 testing pairs from the remaining 20 classes. As discussed before, since this data set was collected using category-level annotations, there remain inconsistencies in terms of attribute-level annotations. Due to this, average accuracy of all the methods is close to chance accuracy (which is 50% for pair-wise predictions). Hence, we believe that the LFW-10 data set is more suitable for comparisons.

Table 2.2 shows the average accuracies over all the attributes obtained by different methods on LFW-10 data set. Several observations can be made from these results: (1) The performance for SPM is comparable to chance accuracy. This is probably because the blocks are not big enough to capture minute differences in small parts for learning attributes. This results in learning bigger parts that are not really distinctive with respect to different attributes. (2) The part-based representations always performs

Figure 2.6: For the three attributes with best accuracies ("smiling", "visible-forehead" and "eyes-open" resp.) the first block shows the top five parts and their weights learned using our method, and the second block shows the top five parts selected by human expert.

Method	**Accuracy**
Global DSIFT + RSVM [41]	64.60
Global GIST + RSVM [41]	68.88
Global GIST + RGB + RSVM [41]	69.89
SPM (Upto 2 levels) + RSVM [41]	50.73
SPM (Upto 3 levels) + RSVM [41]	50.01
Human selected parts + Part-specific Vocab. (Ours)	81.43
Human selected parts + Single Vocab. (Ours)	80.22
Unweighted parts + Part-specific vocab. (Ours)	81.33
Unweighted parts + Single vocab. (Ours)	80.07
Learned parts + Part-specific vocab. (Ours)	**81.62**
Learned parts + Single vocab. (Ours)	80.32

Table 2.2: Average relative attribute prediction accuracies using different methods on LFW-10 data set.

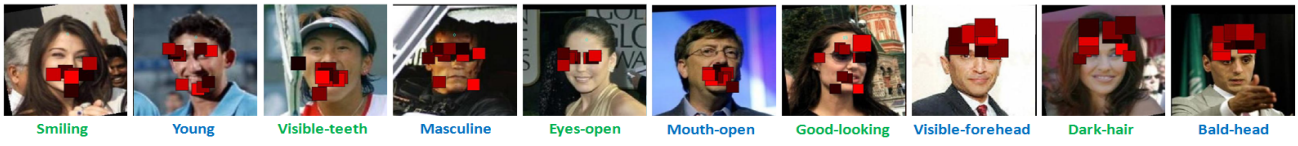


Figure 2.7: Top 10 parts learned using our method with maximum weights for each of the ten attributes in the LFW-10 data set. Greater is the intensity of red, more important is that part, and vice-versa.

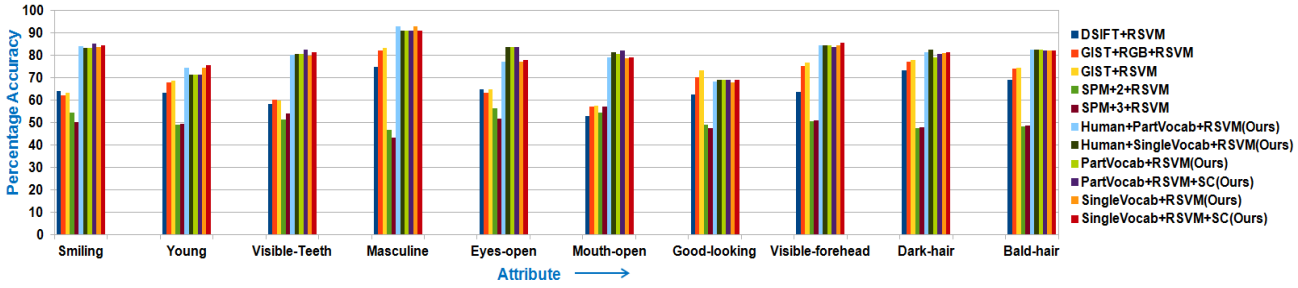


Figure 2.8: Performance for each of the ten attributes in LFW-10 data set using different methods and representations.

significantly better (atleast by 8% on absolute scale) than [41] with different features. This clearly validates the significance of these representations for learning relative attribute ranking models. (3) Single vocabulary always performs better than part-specific vocabulary. One possible reason for this could be the significantly larger (ten times) feature vector for single vocabulary than part-specific vocabulary. Investigating the effect of vocabulary size for these two settings would be an interesting future direction for this work. (4) The performance after combining learned significance-coefficients with parts is always better than unweighted parts (last two blocks of Table 2.2). This reflects the importance of learning and incorporating part-specific weights into the joint representation. (5) The results obtained using learned parts are better or comparable than those using human selected parts. This could be because for humans, it is difficult to precisely assign a weight to every part (hence we used equal weights for all human selected parts). However, this limitation is overcome by our optimization framework ($OP4$) that allows to learn part-specific weights for a given attribute. Figure 2.6 shows the top five parts with highest significance-coefficients, and the top five parts selected by human-expert for the top three attributes with highest accuracies. Figure 2.7 shows the top ten learned parts with highest significance-coefficients for all the ten attributes in the LFW-10 data set. These demonstrate that even by using weak associations between image pairs and their attribute annotations, our framework can efficiently learn discriminative as well as semantically representative parts for different attributes.

In Figure 2.8 we show the performance of different methods for each of the ten attributes on the LFW-10 data set. Here, we can observe that the proposed methods always performs better (sometimes significantly) or comparable to the baseline method of [41]. Also, for each attribute, our performance closely matches with that obtained using human selected parts, thus demonstrating the effectiveness of our method.

2.5.5 Application to Interactive Image Search

Now, we illustrate the advantage of the proposed method on the task of interactive image search using relative attribute based feedback. Our feedback collection set-up is similar to that of [42]. Given

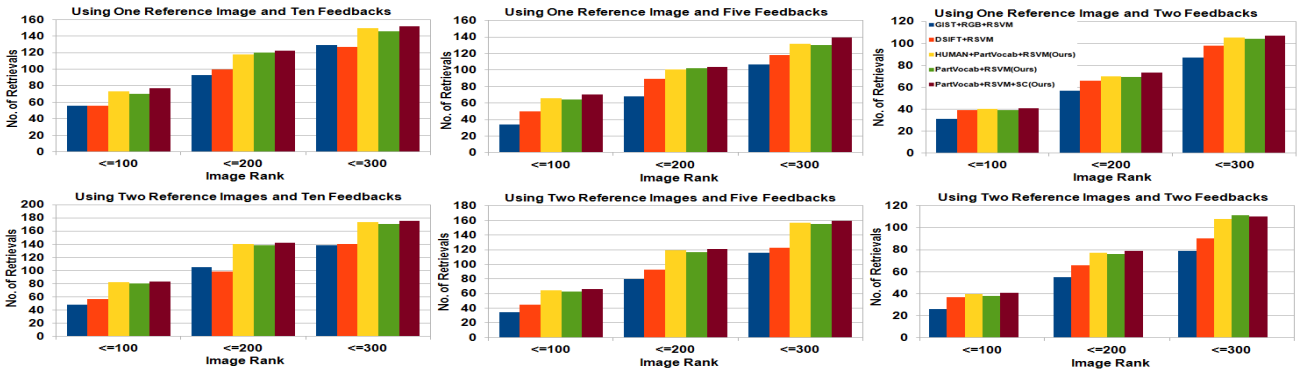


Figure 2.9: Performance variation of different methods on interactive image search with number of reference images and number of feedbacks. Each plot shows the number of searches in which the target image is ranked below a particular rank. Larger is the number of searches falling below a specified rank, better is the accuracy.

a target image, it needs to be described relative to a few reference images (which are different from the target image) based on relative attributes. For a given attribute's feedback with respect to a reference image, the search set is partitioned into two disjoint sets using that attribute's scores. The rank of all the images in the search set are averaged over all feedbacks over all reference images. To break-up ties, absolute classifier score difference with respect to reference image is used. The intuition behind this set-up is that the images which match maximum with attribute feedback should be ranked towards the top.

The 1000 test images of the LFW-10 data set comprise our search set. We keep number of reference images to be either one or two, and vary the number of attribute-based feedbacks per reference in $\{2, 5, 10\}$. A total of 275 searches are performed for each of the six settings, by collecting feedbacks from 30 human evaluators. Figure 2.9 shows the performance of different methods for the six settings. For a given rank, we compute how many target images are predicted below that rank. This means that more is the number of search images falling below a specified rank, better is the performance. From the results, we can observe that the performance of all the methods improves with increase in number of feedbacks and/or number of reference images. This is expected since more interactions (feedbacks) result in better describing the target image. These results demonstrate that here also our method consistently outperforms the baseline method, and achieves performance comparable to that using human selected parts, thus validating its efficacy.

Chapter 3

Relative attributes to Relational attributes

Relative attributes [41] provide a very appealing way of comparing two images based on their visual properties such as *"chubby-face"* for face images, *"naturalness"* for outdoor images, etc. In this work, we propose *relational attributes*, which provide a more natural way of comparing two images based on some given attribute than relative attributes. Relational attributes consider not only the content of a given pair of images, but also take into account its relationship with other pairs, thus making the comparison more robust. For this, we present a Gaussian Process based regression model. Unlike the usual practice of performing regression over samples, we perform regression over pairs for predicting the relative strength of some attribute given a pair of images. We perform thorough experiments to provide insights into our method, and to demonstrate its advantage over the model of [41].

3.1 Introduction

In [41], given a set of pairs of images with similar and/or different strength of an attribute, the problem of learning relative attributes is posed as one of learning a randking model similar to Ranking-SVM (or RSVM) [27] for that attribute. However, given a new pair of images, this approach is limited to the content information of the given pair, and does not consider the situation where there exist relationships *among* the pairs. A simple analogy to this is given two images of *"cars"*, we need to come-up with a prediction of the form *"one image has more car than other"*. In such a scenario, what *"more car"* means needs to be learned by comparing the given pair with other pairs of images of cars. This leads us to the notion of *"relational attributes"*, where the prediction of relative strength of some attribute for a given pair of images relates with the strength of that attribute in other pairs. We claim that the concept of relational attributes is more general and natural than that of relative attributes. This is because in latter (in RSVM), when we use the difference between features of images, the *contribution* coming from the relative strength of some attribute may not be very significant on its own. However, when we consider its consistent presence among several other pairs, the contribution from attribute's presence becomes dominant, and the impact of image-specific properties gets smoothed-out. The major contribution of this work is thus to propose the notion of relational attributes, where given a new pair of images, the

prediction on attribute strength depends on both the content of this pair as well as its relationship with available pairs. To model this task, we define a ranking function based on a regression model. Precisely, we use Gaussian Process Regression (GPR) [44] and improvise it internally to perform pair-wise ranking. This inspired from the widespread use of Gaussian models in modeling diverse real-life sample distributions, where it is assumed that outliers would be uniformed-out and true samples would contribute more in the distribution. We experimentally demonstrate its advantage over the model of [41], and show that it not only shows quantitative improvement, but also exhibits more robust behaviour under different scenarios. The rest of the chapter is organized as follows. In section 3.2, first we discuss the method of [41] and then present our model. We present experiments and results in section 3.3.

3.2 Our Method

As discussed in 2.8.1 The method used in [41] uses a ranking function g_m as given below:

$$g_m(\mathbf{x}_p, \mathbf{x}_q) = \mathbf{w}_m^T(\mathbf{x}_p - \mathbf{x}_q) \tag{3.1}$$

Given a new pair of images I_p and I_q, we determine which image has higher strength for some attribute a_m based on $y_{pq} = sign(g_m(\mathbf{x}_p, \mathbf{x}_q))$, where higher strength of a_m in I_p than I_q corresponds to $y_{pq} = 1$, and lower strength otherwise. From equation 3.1, it can be seen that given a new pair of images, g_m considers (only) the content of these images to make the final prediction. Next we discuss a regression model for pair-wise ranking that considers both the content of a given pair of images as well its relationship with other (known) pairs while prediction.

3.2.1 Similarity-based Scoring Function

Here we present a model that also considers relationship of a given pair of images with other (seen) pairs. This relationship is based on visual similarity of the given pair other pairs. Due to the inherent complex nature of this similarity, it is likely to be a highly non-linear function of image-pair features. As discussed in [44], Gaussian Processes provide a suitable platform to model such similarities, allowing to model non-linear regression under a Bayesian formulation. A Gaussian Process (GP) is collection of random variables such that each such variable has a Gaussion distribution [44]. In our case, each random variable is an image-pair. Thus, we define a pair-wise ranking function h_m as

$$h_m(\mathbf{x}_p, \mathbf{x}_q) = \mathbf{K}(\mathbf{x}_{pq}, \mathbf{X}_m)[\mathbf{K}(\mathbf{X}_m, \mathbf{X}_m) + \sigma^2\mathbf{I}]^{-1}\mathbf{y}_m \tag{3.2}$$

where $\mathbf{K}$ is a pre-defined kernel function that gives the gram matrix, $\mathbf{x}_{pq} = \mathbf{x}_p - \mathbf{x}_q$ is difference of features of the two given images, $\mathbf{X}_m$ is a matrix of difference of feature vectors for all image-pairs in $\mathcal{D}_m$, $\mathbf{y}_m \in \{-1, 0, 1\}^{|\mathcal{D}_m|}$ is a vector indicating the relative strength of corresponding image-pairs in $\mathbf{X}_m$, σ is a noise parameter, and $\mathbf{I}$ is the identity matrix. The above equation can be rewritten as:

$$h_m(\mathbf{x}_{pq}) = \phi_m(\mathbf{x}_{pq})^T\mathbf{v}_m \tag{3.3}$$

where $\phi_m(\mathbf{x}_{pq})^T = \mathbf{K}(\mathbf{x}_{pq}, \mathbf{X}_m)$ and $\mathbf{v}_m = [\mathbf{K}(\mathbf{X}_m, \mathbf{X}_m) + \sigma^2\mathbf{I}]^{-1}\mathbf{y}_m$. Since $\mathbf{v}_m$ can be pre-computed, this means that given $\mathbf{x}_{pq}$, the function h_m can be computed efficiently as a dot product. In pracice, we use the Gaussion kernel $k(x, y) = exp(\frac{-||x-y||^2}{2\gamma^2})$ to compute similarity between two samples. Note that unlike the usual practice of performing regression over individual samples, we do it over image-pairs. This suits our problem, where we have to predict a score given two images.

3.2.2 Optimization

Since the constraints in equation 2.3 and equation 2.4 are relaxed into a convex problem, it can be efficiently optimized guaranteeing global optima. In practice, we optimize this problem in its primal form itself using the Newton's method as discussed in [8]. In equation 3.2, we set $\sigma = 0.3$ and optimize the γ-parameter of the Gaussian kernel using cross-validation. Instead of using some complicated procedure for optimizing the GP model, we found this to be practically more efficient in terms of both time as well as performance.

3.2.3 Discussion

Here we try to give an intuitive explanation of function h_m. When comes a new image-pair, this function first uses *their* content ($\phi_m(\mathbf{x}_{pq})$ in equation 3.3), and then *ask* its peers (i.e., other available pairs) about which of the two images has higher strength of an attribute ($\mathbf{v}_m$ in equation 3.3). The peers then propagate their opinions to the new pair depending on their relationship (similarity) among themselves as well as with this pair. This way, the final score takes care of both self as well as neighbourhood opinions, thus building a relational coherence among all the pairs.

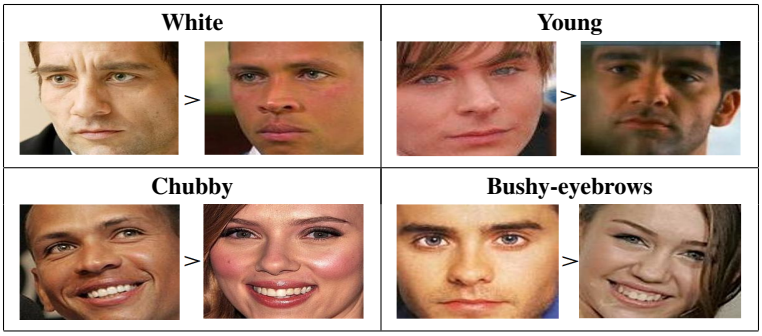
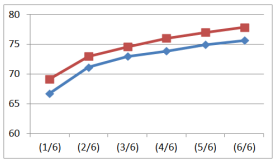


Figure 3.1: Example pairs from the PubFig [31] dataset.



Method	$(1/6)th$	$(2/6)th$	$(3/6)th$	$(4/6)th$	$(5/6)th$	$(6/6)th$
RSVM [41]	66.76	71.15	72.98	73.88	74.95	75.67
GPR (Ours)	69.16	72.97	74.61	76.01	77.03	77.88

Figure 3.2: Variation in performance on changing the number of training pairs. The blue line corresponds to RSVM [41] and the red line corresponds to GPR (ours). The first column in the table on the right shows what ratio of training data is used for training. See section 3.3.2.1 for details.

3.3 Experiments

3.3.1 Dataset and Features

We use a subset of the Public Figure (PubFig) Face Database [31] to validate the efficacy our approach. This dataset has close-up face images of several popular icons such as actors, politicians, etc. We follow the same set of 8 entities (e.g. *Hugh Laurie*, *Scarlett Johansson*, etc.) and 11 attributes (e.g. *smiling*, *young*, etc.) as in [41]. We collect ~ 400 pairs per attribute using feedback from human subjects. From these, we use ~ 300 pairs per attribute for training, thus giving $\sim 3,300$ training pairs; and ~ 100 pairs per attribute for testing which gives $\sim 1,100$ testing pairs. Figure 3.1 shows example pairs for some of the attributes from this dataset. Similar to [41], we represent each image using a 512-dimensional GIST feature [39]. It is a global feature that captures perceptual properties of an image such as naturalness, ruggedness, roughness, etc.

3.3.2 Experimental Set-up and Results

We perform several experiments to analyze the performance of the two models; i.e. RSVM and GPR. In our experiments, all the results are averaged over 10 random train-test splits, and hyper-parameters are tuned using five-fold cross-validation. The performance is evaluated based on the percentage of correct predictions relative to ground-truth ordering.

3.3.2.1 Experiment-1: Effect of Training Data Size

In this experiment, we examine how the two methods perform on varying the size of training data (keeping the size of the test data unchanged). This will give us clue about the effect of data size on performance. For this, we pick $50, 100, \ldots, 300$ training pairs and learn the different models using only these. Figure 3.2 shows the variation in performance of different methods. As obvious, the performance of both the methods improves on increasing the size of training data. We can see that GPR is relatively less sensitive to dataset size than RSVM. This is because (as discussed in section 3.1) GPR makes use of the relative strength of an attribute among all the available pairs which makes the contribution coming

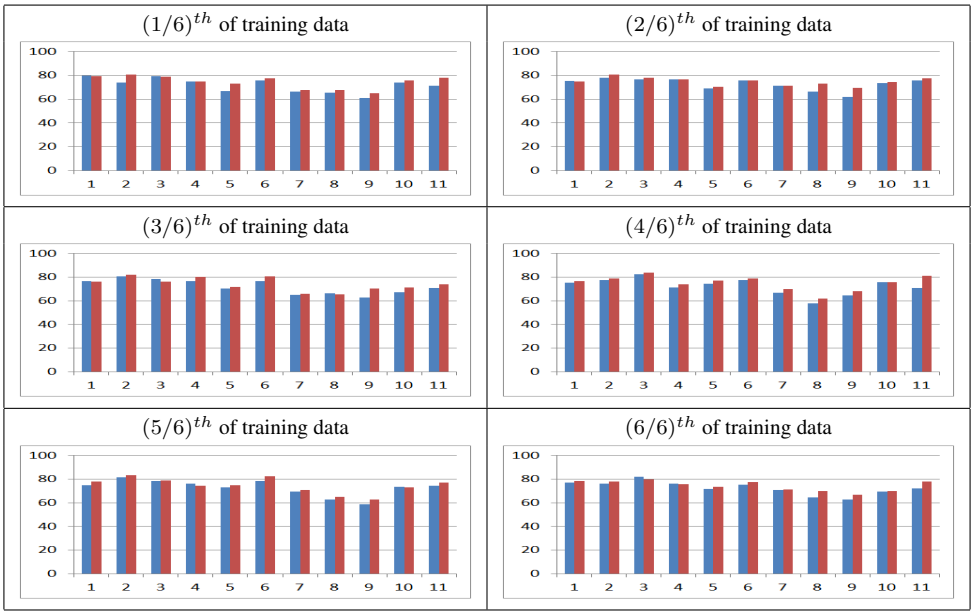
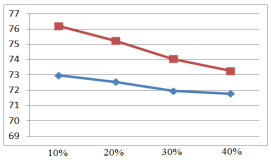


Figure 3.3: Performance of each attribute with variation in size of training data. The blue bar corresponds to RSVM [41] and the red bar corresponds to GPR (ours). See section 3.3.2.1 for details.

from *relative attribute* dominant and thus suppresses image-specific properties. Figure 3.3 shows the variation in performance of all the attributes under this set-up. Here, we can observe that our method performs better than RSVM for most of the attributes, sometimes by even upto 10%, thus verifying the consitent averaged improvement achieved by it across attributes as shown in Figure 3.2.

3.3.2.2 Experiment-2: Effect of Noise in Training Data

In this experiment, we examine how the two methods perform when the training data is noisy. This will help us in gaining insights about the relative robustness of these methods against noisy data. For this, we use the whole training data (i.e., 300 pairs per attribute), explicitly flip the ground-truth of 10%, 20%, 30% and 40% random pairs, and then learn the different models. The test data size is kept unchanged as before. Figure 3.4 compares the performance of the two methods under this set-up. For both the methods, performance reduces on increasing the noise in training data, which is obvious. It can be noticed that the robustness against noisy training data consistently improves from RSVM to GPR, thus establising its effectiveness over RSVM. Figure 3.5 shows how the performance of all the attributes varies under this set-up. We can notice that our method mostly shows more robust behaviour than RSVM, thus validating its efficacy for variety of attributes.



Method	10%	20%	30%	40%
RSVM [41]	66.76	71.15	72.98	73.88
GPR (Ours)	69.16	72.97	74.61	76.01

Figure 3.4: Variation in performance with different noise-level in training data. The blue line corresponds to RSVM [41] and the red line corresponds to GPR (ours). See section 3.3.2.2 for details.

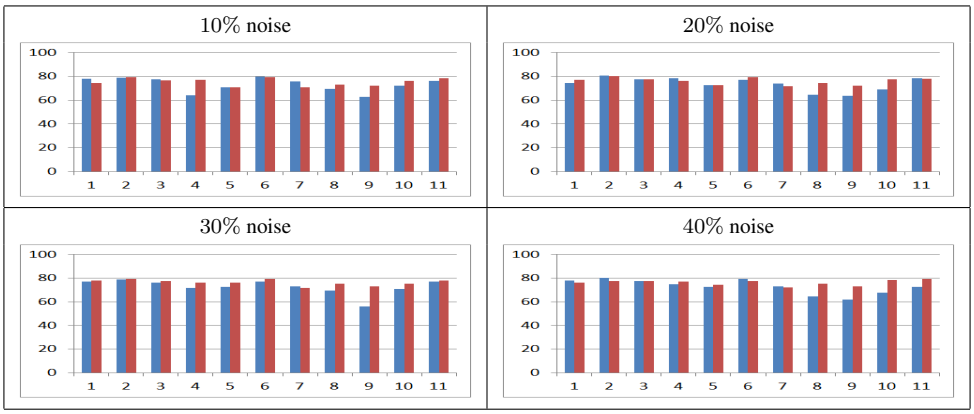


Figure 3.5: Performance of each attribute with variation in noise-level in training data. The blue bar corresponds to RSVM [41] and the red bar corresponds to GPR (ours). See section 3.3.2.2 for details.

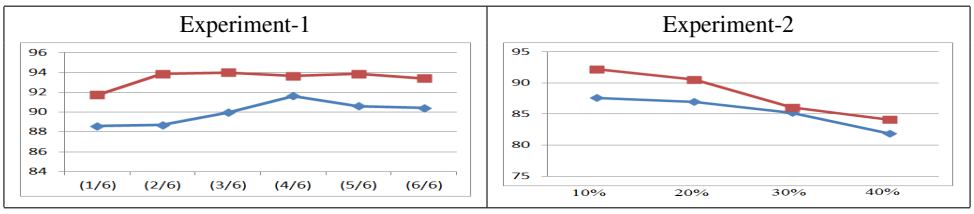


Figure 3.6: Variation in conditioned-performance as discussed in section 3.3.2.3. The blue line corresponds to RSVM [41] and the red line corresponds to GPR (ours).

3.3.2.3 Experiment-3: Conditioned-Performance

In the above two experiments, we have compared the performance of the two methods in an independent manner. Here we try to compare the *conditioned*-performance of the two methods under the above two experiments. For this, first we identify those test pairs that are correctly predicted by RSVM and evaluate the performance of our method (GPR) only on these pairs. Second, we identify those test pairs that are correctly predicted by our method and evaluate the performance of RSVM only over these pairs. This evaluation is performed for both Experiment-1 and Experiment-2. Figure 3.6 shows the conditioned-performance of the two methods. These results clearly demonstrate the relative advantage of GPR over RSVM. The performance of GPR is always above 91% on the pairs that are correctly predicted by RSVM. Whereas, RSVM always shows inferior conditioned-performance than GPR for both Experiment-1 as well as Experiment-2.

3.3.3 Discussion

Figure 3.7 shows some qualitative results corresponding to Experiment-1 (with complete training data). These are the pairs where GPR gave correct predictions while RSVM gave incorrect predictions. Even for hard pairs such as those corresponding to *Visible-forehead* and *Big-lips*, GPR correctly predicts the ordering. From these results, we can also see that even for image pairs where the difference in relative strength of some attribute clearly looks significant (such as *Smiling*), RSVM fails to capture this. This is probably because of its rigid behaviour From all these quantitative and qualitative experiments, we can conclude that GPR always performs better than RSVM. However, the main contribution of this work is not just to propose a better model for learning relative attributes as originally proposed in [41], but also to advocate the need for investigating more natural and robust ways of comparing two images based on some given attribute. For this, we presented relational attributes, where the relative stregth of some attribute given a pair of images also depends on their relationship with other pairs. Along with better quantitative results, out method presents a new dimension of looking at the broader problem of understanding attributes from a collection of image pairs. Conceptually, both relative and relational attributes are trying to capture the same thing: relative strength of some attribute given a pair of images. It is the learning procedure that makes them fundamentally different. Since two images can also be compared based on multiple attributes, this allows us to look at this problem as that of predicting multiple relative attributes given two pairs. The success of generative models in the image annotation task [20, 22] can thus be related with the better performance of GPR over RSVM, encouraging the need for further exploring generative models while comparing two images based on some given attribute.

Figure 3.7: Example orderings of pairs that are correctly predicted by our method, but incorrectly by the method of [41].

Chapter 4

Conclusions and Future Work

In this thesis, we have explored the problem of Relative attribute Prediction and presented part based solution to solve the problem. We have presented a novel method that learns relative attribute models using local parts that are shared across categories. We have evaluated our method in facial domains. In facial domain, parts are acquired using facial landmark detection. The part-based representation combines a pair of images that specifically compares corresponding parts. Then, with each part we associate a locally adaptive significance coefficient that represents its discriminative ability with respect to a particular attribute. For each attribute, the significance-coefficients are learned simultaneously with a max-margin ranking model in an iterative manner. The Part based representation that we have proposed for learning relative attributes gives significant improvement in accuracy over the previous methods. The learned parts which are learned by adding significant coefficients also gives a semantically interpretation of attributes. Apart from the relative attribute prediction, we have shown advantage of our method on Interactive image search.

Inspired from the success of attributes and relative attributes, we present relational attributes. Similar to relative attributes [41], it is equally applicable to several applications such as zero-shot learning, relative image description, constrained bootstraping [47] etc. Despite its simplicity, our method shows superior results than that of [41]. Also, the robustness of such learning methods against the size of training data and noisy training data has been examined for the first time.

In the future, we would like to extend the work of Relative parts to outdoor scenes where the definition of parts itself is not properly defined because of large variety of images. We are trying to solve the problem using an unsupervised learning approach to find parts for predicting relative attributes for outdoor scenes. Also we would like to investigate even better learning models that would further add to the semantic richness of relative and relational attributes.

Related Publications

1. Ramachandruni N Sandeep, Yashaswi Verma, C.V. Jawahar
   Relative Parts: Distinctive Parts for Learning Relative Attributes
   IEEE Conference on Computer Vision and Pattern Recognition (CVPR),2014

Chapter 5

Appendix

5.1 Image Representation

Human visual system is very complex system. Even with all the scientific advances, a very little is known about the functioning of the system. Humans can easily discriminate between different objects they observe irrespective of their sizes, lighting conditions, view points, quality and various attributes. A computer vision system designed to recognize facial attributes also needs to be invariant to these parameters. E.g,: A smiling attribute detector is supposed to detect a smiling face despite pose changes and different lightning conditions. Similary, a face detector should detect a face regardless of persons ethnicity, and other facial attributes. Images in their raw pixel representations are not often useful to achieve this invariance. Good features representations are therefore necessary to describe characteristics of objects properly . A good feature representation should be invariant to different sources of variations in the imaging conditions and also in the appearance of the objects they represent. The process of representing images in a meaningful way is called feature extraction. There are many feature representations developed for computer vision tasks[36, 39, 12].

In this section, we will review the feature representations we use in our work The Bag Of Visual Words Model [49] and Spatial Pyramid Representation [4] and Gist features.

5.1.1 Bag of Visual Words

Bag of Words or Bag of Features builds upon the research in the fields of Natural Language Processing and Information Retrieval. A text document consists of several words from a set defined by a vocabulary. These words or collection of words can then be used to identify the topic and contents of the given document. Indexing based on words helps in retrieving relevant documents based on a query. In computer vision domain, an image is analogous to a document, and visual words in the image are analogous to the words in the document. These visual words are obtained from vector quantizing local features computed in the image. Visual vocabulary necessary for this quantization is generated by clustering local features.

Bag of words feature computation consists of following steps:

1. Finding regions of interests
2. Computation of local descriptors
3. Vector quantization of descriptors to form a vocabulary
4. Computation of histogram of visual words.

Finding Regions of Interest: Process of finding regions of interest is known as feature detection. These are the regions in the image which are invariant to scale variation, rotation and affine transformations. Local descriptors are computed on these regions which inherit and enhance these properties. Several region detectors have been introduced in the past.

Some examples of these region detectors are

i) Harris Points

ii) Harris-Laplace regions

iii) Hessian- Laplace regions

iv) Harris-Affine

v) Hessian-Affine

vi) Maximally Stable Extremal Regions.

These detectors use low level image processing operations to find the regions in the image. Smoothing by a Gaussian kernel in a scale-space representation followed by local derivative operations are typically the steps in finding such regions. These regions are mainly used for solving correspondence problems. More detailed discussion and comparison of these detectors can be found in [37]. For classification tasks, uniform sampling of the points from an image was shown to be beneficial than using sparse detected points [34]. In uniform sampling, a spatial grid is laid on top of the image and interest points are chosen to be the intersection points of this grid. 5x5 pixels spacing of the grid is common in practice but this parameter can also be determined experimentally.

5.1.1.1 Scale Invariant Feature Transform(SIFT)

Scale invariant feature transform [36] is a local feature descriptor. Sift descriptor is a histogram of gradient location and orientation and is computed on normalized image patches. The location is quantized into a 4x4 location grid and the gradient direction is quantized into 8 orientation bins. The result is a 128 dimensional feature vector.

5.1.1.2 Dense Scale Invariant Feature Transform (DSIFT)

Dense Sift is roughly equivalent to running SIFT on a dense grid of locations at a fixed scale and orientation. This type of feature descriptors is often uses for object categorization. Bin size vs keypoint scale. DSIFT specifies the descriptor size by a single parameter, size, which controls the size of a SIFT spatial bin in pixels. In the standard SIFT descriptor, the bin size is related to the SIFT key point scale by a multiplier, denoted magnif , which defaults to 3. As a consequence, a DSIFT descriptor with bin

size equal to 5 corresponds to a SIFT key point of scale $5/3 = 1.66$. Smoothing. The SIFT descriptor smoothes the image according to the scale of the keypoints (Gaussian scale space). By default, the smoothing is equivalent to a convolution by a Gaussian of variance $s^2 - .25$, where s is the scale of the keypoint and $.25$ is a nominal adjustment that accounts for the smoothing induced by the camera CCD.

5.1.1.3 Vocabulary Construction

Having computed features descriptor, vocabulary is computed by collecting similar features into clusters. Cluster centers representing similar features are called as visual words. Clustering algorithms such as K means are used for finding the cluster centers. Number of cluster centers is determined experimentally.

5.1.1.4 Histogram Computation

To compute feature histograms, local feature descriptors are mapped to their nearest visual word by using some distance metric. This mapping assigns a visual word to every feature descriptor. Histograms of visual words are then computed by assigning a bin to every visual word. Thus an image with varying number of local descriptors always results in the same feature vector dimension as long as the underlying vocabulary remains constant. Loss of spatial information is one of the drawbacks of this scheme. In practice, it is often important to encode spatial information into features. To preserve the spatial structure, spatial pyramid representation is used for computing histograms.

5.1.2 Spatial Pyramid Representation

It is an extension of bag of features, locally order less representation at several levels of resolution. Method is based on Pyramid match kernels. It works by partitioning the image into increasingly fine sub-regions and computing histograms of local features found inside each sub-region. Number of divisions increase as the level is increased. All computed histograms are concatenated to form a final feature vector. There is no fixed formula for determining number of cells on every level. Levels of the pyramid are also determined experimentally. However, it is common practice to divide an image into 4l cells at every level l and representations going upto 2 levels. Figure 2.2 shows the concept of spatial pyramids diagrammatically.The resulting spatial Pyramid is a simple and computationally efficient extension of an order less bag of features image representation.

5.1.3 GIST Features

This feature computation uses context based approach, and consider the input image as a whole and extract a low-dimensional signature that summarizes the image statistics and semantics. Computing gist involves accumulating image statistics over the entire scene. Process of extracting the gist of an image using features from several domains, calculating its holistics characteristics but still taking into

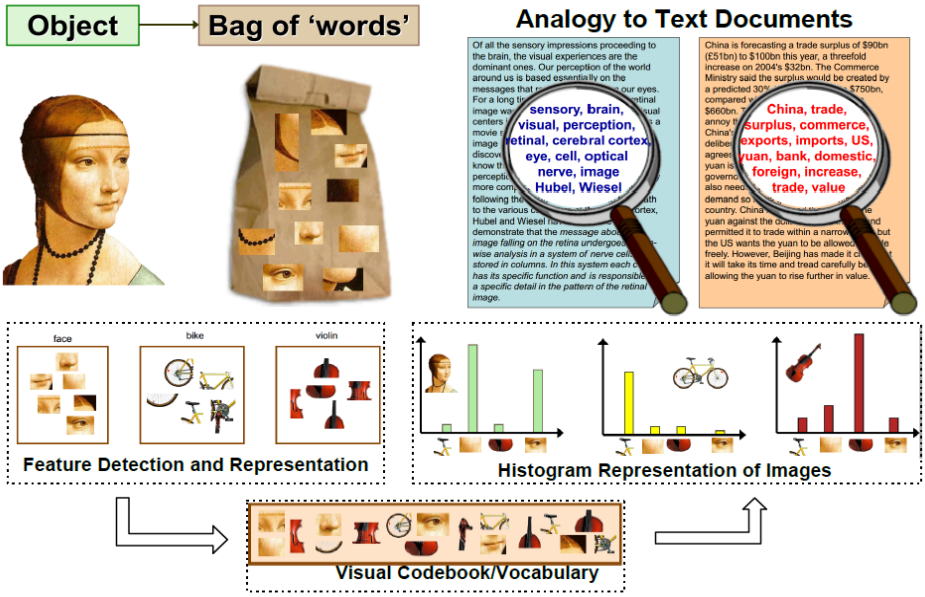


Figure 5.1: A graphical representation of Bag of Words (BoW) model [49], that shows how an image of a object is represented as a bag or multiset of visual words. Its analogy with text documents is also clear. In the example, 3 objects (face, bike and violin) can be seen to be intuitively composed of their local interest regions and a histogram of visual word vocabulary (or the parts) is used to represent them.

account coarse spatial information. An input image is filtered in a number of low-level visual feature channels- color, intensity, orientation, flicker and motion at multiple spatial scales. Such channels have several sub channels one for each color type, orientation or direction of motion. Each sub channel has a nine scale pyramidal representation of outputs. Within each sub channel the model performs center-surround operations between filter output at different scales to produce feature maps. The different feature maps for each type allows the system to pick up regions at several scales with the added lighting invariance. The intensity channel output for the illustration image of figure below shows different sized regions being emphasized according to their respective center-surround parameter. After the center-surround features are computed, each sub-channel extracts a gist vector from its corresponding feature map. Saliency and gist emphasize two complementary aspects of the data in the feature maps: saliency focuses on the most salient peaks of activity while gist estimates overall activation in different image regions.

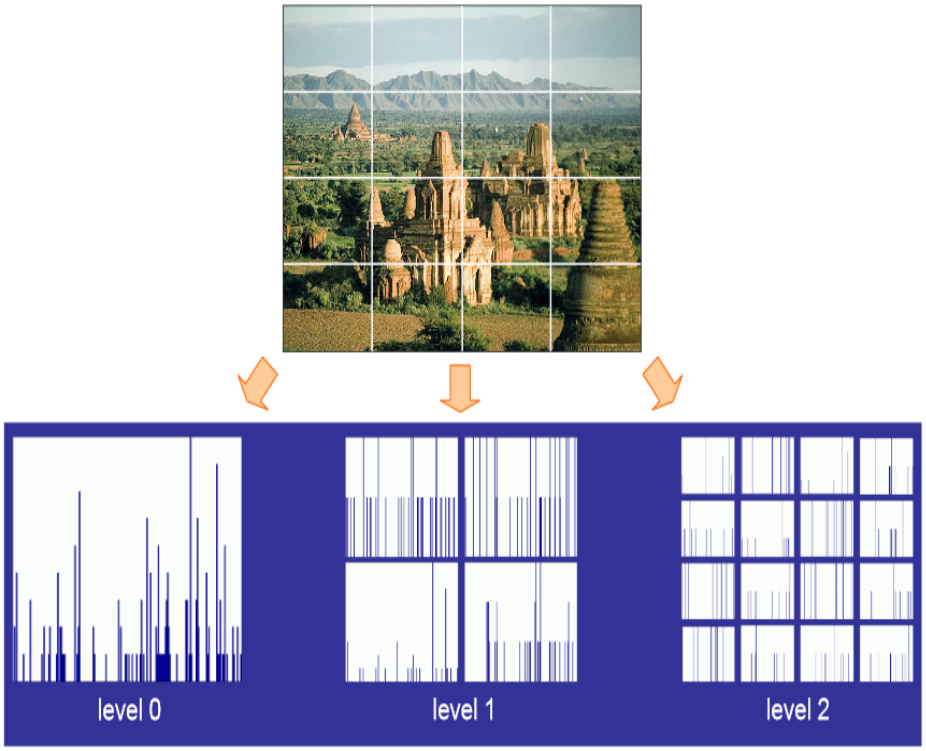
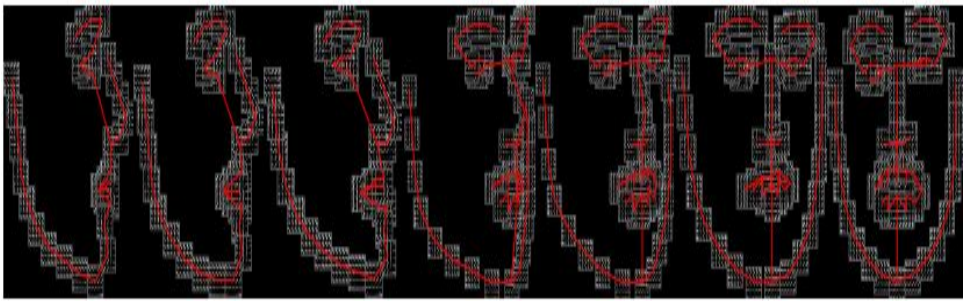


Figure 5.2: SpatialPyramid Feature Extraction [33]: computing BOW for image at different regions in various scales

5.2 Detecting Parts on Human Faces

In our work , we have to detect parts for face like eyes, mouth, nose etc. After detecting landmark points we draw bounding boxes around it and call it as part. For detecting the landmark points in face, we have used the method zhu and ramanans face detection, pose estimation and landmark localization in the wild[57]. The model is based on mixture of trees and shared pool of parts. They model every facial landmark as a part and use global mixtures to capture topological changes due to view point. All parameters of the model including part templates, modes of elastic deformation and view based topology are discriminately trained in max margin framework. To learn the model, they assume a fully-supervised scenario, where they are provided positive images with landmark and mixture labels, as well as negative images without faces. Learn both shape and appearance parameters discriminatively using a structured prediction framework. We use this method to detect landmark points for faces in our LFW-10 dataset. After getting the landmark points we draw a bounding box around each of the 68 landmark points and consider them as different parts of the face. There are some other attributes which are not covered by these parts. In order to cover all the attributes, we compute additional parts using image-level statistics such as image-size and distance from the earlier 68 parts.This gives an extended set of 83 parts for a given face. Below figure shows the 68 parts and 83 parts computed using the method.



.

Figure 5.3: Mixture of trees model encodes topological changes due to view point. Red lines denote springs between pairs of parts. All trees make use of a common, shared pool of part templates, which makes learning and inference efficient[57]

Figure 5.4: Results of the model[57] shows accurate detection of face and estimate pose and estimate deformations in real world and cluttered scenes

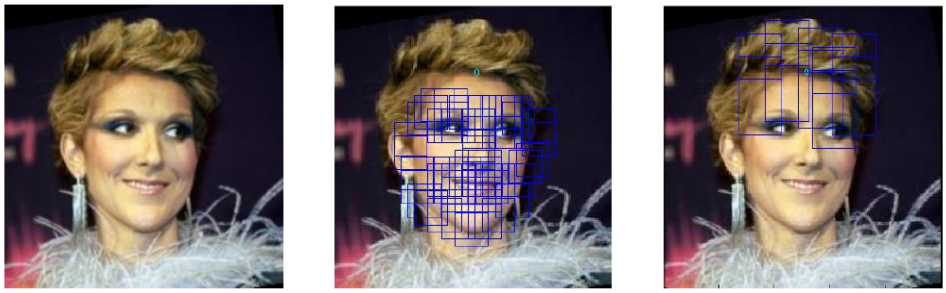


Figure 5.5: Input image (left), parts detected using [57] (middle), and additional parts detected by us (right).

5.3 Learning Models

The machine learning community has developed so many methods for supervised, unsupervised and semi-supervised ways of learning models for various tasks like classification, regression, clustering etc. In our work we learn models to find the relative strength of attribute in two images. In order to rank the two images based on the strength of attribute we have used Ranking SVM in our method. In this section we review learning how to rank and the most successful support vector machines which were later extended to ranking task named Ranking SVM.

5.3.1 Support Vector Machines

Support Vector Machine [5, 11] is a popular and powerful classification learning tool. It is a supervised learning method, i.e. it learns from a given set of labelled training data and predicts the label for an unseen test sample. We will explain SVMs for two-class case, which is also called as a binary classifier. The basic idea behind a linear classifier is to separate the given D-dimensional data points with a (D-1) dimensional hyperplane. For a given set of points, there may exist multiple hyperplanes which can separate the data Figure:5.6a. The best classifier of all these hyperplanes is the one which provides the maximum separation of the data points Figure:5.6b. It essentially means that the best hyperplane should maximise the distance between the nearest points on both sides of the hyperplane (nearest points to the hyperplane from each class). This distance is defined as the margin, and the SVM selects the hyperplane with the maximum margin. The hyperplane obtained is called maximummargin hyperplane and the linear classifier is called maximum-margin classifier.

Given a set of n labelled training samples,

$$S = \{\{x_i; y_i\} | x_i \in \Re^D, y_i \in \{-1, 1\}\} \tag{5.1}$$

x_i is the D-dimensional data point, y_i represents the class to which the point x_i belongs.

A seperating hyperplane with W as the normal vector can be written as

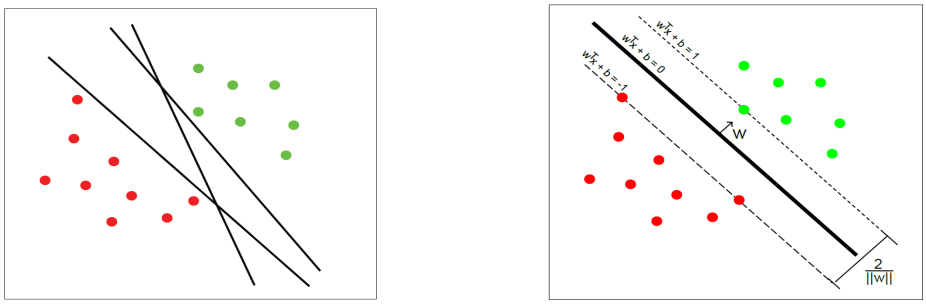
$$W^T X + b = 0 \tag{5.2}$$

Here b is called the bias term, $\frac{b}{\|W\|}$ gives the perpendicular distance from the origin to the hyperplane.Our goal is to find the w and b, such that the margin is maximized. We can select two parallel hyperplanes which separate the data and are as far as possible. These hyperplanes can be written as follows:

$$W^T X + b = -1 \tag{5.3}$$

$$W^T X + b = 1 \tag{5.4}$$

Now, the distance between the two parallel hyperplanes is $\frac{2}{\|W\|}$. Since the distance needs to be maximized, it translates to minimizing $\| W \|$. Since we do not want any data points falling in between the



(a) Multiple Seperating Hyperplanes (b) Max-Margin Hyperplane

Figure 5.6: Example showing multiple separating hyperplanes and the max-margin hyperplane output by SVM.

two hyperplanes, the following constraints are added:

$$W^T X_i + b \geq 1 \;\; \forall X_i \; s.t \;\; y_i = 1 \tag{5.5}$$

$$W^T X_i + b \leq -1 \;\; \forall X_i \; s.t \;\; y_i = -1 \tag{5.6}$$

The two constraints can be combined and rewritten as:

$$y_i \left(W^T X_i + b \right) \geq 1 \;\; \forall X_i \tag{5.7}$$

We can substitute $\|W\|$ with $\frac{1}{2}\|W\|^2$, without changing the solution, this makes the optimization problem easier to solve. The optimization problem can now be written in primal form as:

$$min_{w,b} \; \frac{1}{2}\|W\|^2 \tag{5.8}$$

$$subject \; to \; y_i \left(W^T X_i + b \right) \geq 1 \;\; \forall X_i \tag{5.9}$$

5.3.2 Learning to rank

Ranking is a central part of many information retrieval problems, such as document retrieval, collaborative filtering, sentiment analysis, computational advertising (online ad placement). Learning to rank has received exten- sive attention in the machine learning literature [7, 27, 35] for information retrieval in general and image retrieval in particular [25, 23].Given a query image, user preferences (often captured via click-data) are incorporated to learn a ranking function with the goal of retrieving more relevant images in the top search results. In [27] Ranking SVM algorithm is proposed which is used to learn rank documents in a infomation retrieval system for optimizing search engines using click through data.

Bibliography

[1] T. Berg and P. N. Belhumeur. POOF: Part-based one-vs-one features for fine-grained categorization, face verification, and attribute estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[2] T. L. Berg, A. C. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web data. In *European Conference on Computer Vision (ECCV)*, 2010.

[3] A. Biswas and D. Parikh. Simultaneous active learning of classifiers & attributes via relative feedback. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[4] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. In *ACM International Conference on Image and Video Retrieval*, 2007.

[5] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pages 144–152, New York, NY, USA, 1992. ACM.

[6] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Perona, and S. Belongie. Visual recognition with humans in the loop. In *In European Conference on Computer Vision (ECCV)*, 2010.

[7] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. Learning to rank: From pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

[8] O. Chapelle. Training a support vector machine in the primal. In *Neural Computation*, 2007.

[9] J. Choi, M. Rastegari, A. Farhadi, and L. Davis. Adding unlabeled samples to categories by learned attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[10] G. Christie, A. Parkash, U. Krothapalli, and D. Parikh. Predicting user annoyance using visual attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[11] C. Cortes and V. Vapnik. Support-vector networks. *Mach. Learn.*, 20(3):273–297, Sept. 1995.

[12] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[13] M. Danelljan, F. Shahbaz Khan, M. Felsberg, and J. Van de Weijer. Adaptive color attributes for real-time visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014.

[14] J. Deng, J. Krause, and L. Fei-Fei. Fine-grained crowdsourcing for fine-grained recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[15] K. Duan, D. Parikh, D. Crandall, and K. Grauman. Discovering localized attributes for fine-grained recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[16] Q. Fan, P. Gabbur, and S. Pankanti. Relative attributes for large-scale abandoned object detection. In *IEEE International Conference on Computer Vision*, 2013.

[17] A. Farhadi, I. Endres, and D. Hoiem. Attribute-centric recognition for cross-category generalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[18] A. Farhadi, I. Endres, D. Hoiem, and D. A. Forsyth. Describing objects by their attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[19] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 32(9):1627–1645, 2010.

[20] S. L. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[21] V. Ferrair and A. Zisserman. Learning visual attributes. In *NIPS*, 2007.

[22] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbour models for image auto-annotation. In *IEEE International Conference on Computer Vision*, 2009.

[23] Y. Hu, M. Li, and N. Yu. Multiple-instance ranking: Learning to rank images for image retrieval. cvpr, 2008.

[24] 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. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[25] V. Jain and M. Varma. Learning to re-rank: Query-dependent image re-ranking using click data. In *Proceedings of the 20th International Conference on World Wide Web*, 2011.

[26] D. Jayaraman, F. Sha, and K. Grauman. Decorrelating Semantic Visual Attributes by Resisting the Urge to Share. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[27] T. Joachims. Optimizing search engines using clickthrough data. In *KDD*, 2002.

[28] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman. Blocks that shout:Distinctive parts for scene classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[29] A. Kovashka, D. Parikh, and K. Grauman. WhittleSearch: Image search with relative attribute feedback. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[30] N. Kumar, P. Belhumeur, and S. Nayar. Facetracer: A search engine for large collections of images with faces.

[31] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attributes and simile classifiers for face verification. In *IEEE International Conference on Computer Vision*, 2009.

[32] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[33] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[34] F.-F. Li and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[35] T.-Y. Liu. Learning to rank for information retrieval. *Found. Trends Inf. Retr.*, 3(3):225–331, Mar. 2009.

[36] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[37] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65:2005, 2005.

[38] F. Nie, H. Huang, X. Chai, and C. Ding. Efficient and robust feature selection via join $l_{2,1}$-norms minimization. In *NIPS*, 2010.

[39] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[40] D. Parikh and K. Grauman. Interactively building a discriminative vocabulary of nameable attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[41] D. Parikh and K. Grauman. Relative attributes. In *IEEE International Conference on Computer Vision*, 2011.

[42] D. Parikh and K. Grauman. Implied feedback: Learning nuances of user behavior in image search. In *IEEE International Conference on Computer Vision*, 2013.

[43] A. Parkash and D. Parikh. Attributes for classifier feedback. In *European Conference on Computer Vision (ECCV)*, 2012.

[44] C. E. Rasumussen and C. K. I. Williams. *Gaussian Processes for Machine Learning.* The MIT Press, 2005.

[45] A. Sadovnik, A. C. Gallagher, and T. Chen. It's not polite to point: Describing people with uncertain attributes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3089–3096, 2013.

[46] A. Sadovnik, A. C. Gallagher, D. Parikh, and T. Chen. Spoken attributes: Mixing binary and relative attributes to say the right thing. In *IEEE International Conference on Computer Vision*, 2013.

[47] A. Shrivastava, S. Singh, and A. Gupta. Constrained semi-supervised learning using attributes and comparative attributes. In *European Conference on Computer Vision (ECCV)*, 2012.

[48] B. Siddiquie, R. S. Feris, and L. S. Davis. Image ranking and retrieval based on multi-attribute queries. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[49] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *IEEE International Conference on Computer Vision*, 2003.

[50] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of Royal Statistics Society B*, 58:267–288, 1996.

[51] N. Turakhia and D. Parikh. Attribute dominance: What pops out? In *IEEE International Conference on Computer Vision*, 2013.

[52] A. Vedaldi, S. Mahendran, S. Tsogkas, S. Maji, B. Girshick, J. Kannala, E. Rahtu, I. Kokkinos, M. B. Blaschko, D. Weiss, B. Taskar, K. Simonyan, N. Saphra, and S. Mohamed. Understanding objects in detail with fine-grained attributes. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[53] X. Wang and Q. Ji. A unified probabilistic approach modeling relationships between attributes and objects. In *IEEE International Conference on Computer Vision*, pages 2120–2127, 2013.

[54] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, 10:207–244, 2009.

[55] A. Yu and K. Grauman. Fine-Grained Visual Comparisons with Local Learning. In *Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[56] S. Zheng, M.-M. Cheng, J. Warrell, P. Sturgess, V. Vineet, C. Rother, and P. H. S. Torr. Dense semantic image segmentation with objects and attributes. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[57] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.