2011

Gender Recognition and Appearance Description in Unconstrained Images of Human Body

Qin Wu
West Virginia University

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Wu, Qin, "Gender Recognition and Appearance Description in Unconstrained Images of Human Body" (2011). *Graduate Theses, Dissertations, and Problem Reports*. 4814.
<https://researchrepository.wvu.edu/etd/4814>

This Thesis is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Thesis has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

Gender Recognition and Appearance Description in Unconstrained Images of Human Body

Qin Wu

Thesis submitted to the
College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Master of Science
in
Computer Science

Guodong Guo, Ph.D., Chair
Arun A. Ross, Ph.D.
Cun-Quan Zhang, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2011

Keywords: Gender recognition, articulated human body, data fusion,
unconstrained images, support vector machine (SVM).

Copyright 2011 Qin Wu

ABSTRACT

Gender Recognition and Appearance Description in Unconstrained Images of Human Body

Qin Wu

Gender recognition has many useful applications, ranging from business intelligence, through to image search and social activity analysis. Traditional research on gender recognition focuses on face images in a constrained environment. In this work, we propose a novel problem of gender recognition in articulated human body images acquired from an unconstrained environment in the real world. Our empirical study answers the question of whether gender recognition can be performed in articulated body images, and discovers important issues such as which body parts are informative, how many body parts are needed to combine together, and what representations are good for articulated body-based gender recognition. We also pursue data fusion schemes and efficient feature dimensionality reduction based on the partial least squares estimation. Extensive experiments are performed on two unconstrained databases that have not been explored before for gender recognition. At the end of this work, we also present some preliminary results on the automatic description of the appearance of the upper body.

Acknowledgments

First, I would like to express my sincere thanks to my advisor, Dr Guodong Guo for his guidance, encouragement and help. It is a great pleasure to work under his supervision. What I learned from him will benefit me all my life.

Second, I'd also like to thank my committee members: Dr Cun-Quan Zhang and Dr Arun A. Ross, for their help during the completion of my thesis.

Finally, I would like to thank the Lane Department of Computer Science and Electrical Engineering at West Virginia University for providing me with an excellent study environment during my years as a graduate student.

Contents

1	Introduction	1
1.1	Overview of gender recognition	2
1.2	Motivation	4
1.3	Articulated Body Parts Based Approach	5
2	Overview of Articulated Human Body Parts	8
2.1	Upper-Body Detection and Part Localization	9
2.2	Body Parts Normalization	12
2.3	How Many Body Parts Are Needed?	14
3	Feature Descriptors	16
3.1	Histogram of Oriented Gradients (HOG)	17
3.2	Local Binary Patterns (LBP)	18
3.3	Scale Invariant Feature Transform (SIFT)	19
4	Supervised Machine Learning	22
4.1	Support Vector Machine	22
4.2	Partial Least Squares (PLS)	25
4.3	Comparison between SVM and PLS	26
4.4	Data Fusion	27
5	Experiments and Discussion	30
5.1	Datasets Description	30
5.2	Gender Recognition Results with Annotated Body Parts	32
5.3	Gender Recognition Results with Automated Body Parts	35
5.4	Results Based on the PLS Method	37

5.5	Summary of Gender Recognition Results	38
6	Color and Pattern Description	40
6.1	Color Description of Clothes	40
6.2	Pattern Description of Clothes	42
7	Conclusions and Future Work	44

Chapter 1

Introduction

Given an arbitrary photo, people are easily to tell the story presented in the photo. Say for example, if we look at the photo shown in Fig 1.0.1, we may describe it as following: (1) there is a young girl, (2) she wears white top and pink trousers, (3) she is riding bicycle. That is, people can get the following information from the picture: (1) gender of the person in the image, (2) the color (and pattern) of the clothes, (3) what she/he is doing.

Figure 1.0.1: A photo with a young girl riding bike

Our question is: Can computer find all these information automatically? So far, this is still a tough job. It is not a easy job to detect the person on the image, it is even a harder job to detect each body part and tell the action of the person. In this work, we mainly focus on the gender recognition for a given image. And we also present some preliminary work on detection of color and pattern of clothing.

In Chapter 1, we review the traditional gender recognition approach and proposed a new approach for gender recognition. Articulated body parts detection and normalization are presented in Chapter 2. Different feature descriptors for body part representations are discussed in Chapter 3. After we represent each image by some feature descriptor, we introduced support vector learning and partial least square methods for gender classification. Data fusion schemes is also proposed in Chapter 4. In Chapter 5, experiments and some discussion of gender recognition based on body parts are presented. Then we present some preliminary work on appearance description such as color and pattern of clothing in Chapter 6. Some conclusions and future work are presented in Chapter 7.

1.1 Overview of gender recognition

Gender or sex is an important clue of a human being. Humans are divided into two groups: male and female. Different gender groups may present different habits. For example, boys and girls may like different toys; men and women may prefer different clothes (e.g., color, style, pattern, etc.). In business intelligence, knowing the gender (and/or age) of the customers may help business managers to advertise their products according to different customer groups, and collect the valuable demographic information about the customers, e.g. how many women entering a retail store or a shopping mall within a given period of time. In security, the gender information may be used for access control, e.g. restricting access to certain areas based on gender. In image retrieval, gender (and/or age) can be a useful semantic concept for photo organization and search, e.g., finding men or women in certain scenes. In social interactions or activities, males and females may perform differently and have different roles. Therefore recognizing the gender of a person has broad applications.

Humans have the capability to perceive the gender of each other. It has been an active research in psychology to study the perception of gender by humans [3, 27, 24]. In psychology studies, the stimulus is usually the face photos for gender perception. In computational visual analysis for gender classification or recognition, most of the existing works used face images. For example, [2, 13, 12, 4, 17].

Recently, some approaches, e.g., [15, 5], showed that human bodies can be used for gender classification. Successful results are shown on pedestrian images where a manually drawn box containing a pedestrian is used globally for body information extraction.

Figure 1.1.1: Gender recognition using different modalities. Left: face images; Middle: upright and whole body, e.g., pedestrian, where the alignment problem is trivial; Right: articulated bodies which have various body part variations and even only partial body being visible.

The advantages of using body over face images for gender recognition include:

(a) Image resolution. When the face region has a low resolution, or motion blurred, the facial features might not be useable for gender classification. However, the body image may still be useful for separating males from females.

(b) View point change. When the head pose is very different from frontal views, the face-based gender recognition might have low performance or even not applicable. However, the body image can still be used. Even the back view of the body can be used for gender recognition [15, 5].

(c) Acquisition distance. When the camera is far away from the person, the face image will not provide sufficient information for gender discrimination. However, the body image may still be useable.

(d) Occlusion. When the facial part is occluded, the face image might not be used to extract gender information. However, the body image is still useful, even some body parts are occluded.

However, in those pioneering approaches to gender recognition from body, e.g., [15, 5], only the *upright body images with the whole body appearance* are investigated. The data typically used are pedestrian images, i.e., the MIT pedestrian database [18] which is a

common database for pedestrian detection. One important requirement is that the whole body in upright appears in each image, as shown in the middle of Fig. 1.1.1, and thus there is no need to worry about aligning the body images in matching. Features can be extracted from the whole body images and used to train a classifier.

In reality, the human body can undergo many different articulations, e.g., the arms and legs may rotate, the head and torso could bend, and different body parts could move together or separately. And also, human body parts can interact with objects and/or the environment by performing various actions. Further, an articulated body may be captured by a camera just partially in a typical image and in such case, the gender recognition from whole body can not be applied to those image with partial body parts. All these realistic variations, as shown in the right of Fig. 1.1.1, make the body appearance change significantly. Then a research issue is raised: Is it possible to recognize gender from an articulated body?

Our goal is to automatically detect the gender in the images under uncontrolled conditions such as images take by amateur photographer. In this work, we studied the following two problems:

- (1) Can gender recognition be performed or not in articulated body images using a computational method?
- (2) How to extract the gender information from an articulated body?

1.2 Motivation

As shown in the third column of Fig. 1.1.1, human bodies can undergo various articulations, bending, rotations, movements, and interactions with objects. But *our human viewers can perceive gender from those articulated bodies without difficulty*. Please note that some of the example images in the right of Fig. 1.1.1 have no face (a back view) or heavily occluded faces, but human viewers can still perceive the gender. So we believe that some cues from the body other than the face are used by humans to recognize gender. This motivates us to explore a computational method to recognize gender from the articulated body. It is certainly a great challenge. To the best of our knowledge, no previous work has investigated this problem.

Further, humans can perceive gender from partial of the body, not necessarily using the whole body. This motivates us to think about a new issue: gender recognition may

use partial body rather than the whole body. In this work, we use upper body only, because upper bodies convey the majority information for gender recognition.

It is well-known that men and women do have significant biological differences in their bodies. Their appearances are different. Say for example, men and women may wear different clothes (color, style, and patterns). These evidences support the body-based discrimination between males and females.

Interestingly, it is quite common to use symbols of the body (rather than the face) to indicate the male and female in practice, as shown in Fig. 1.2.1.

Figure 1.2.1: Symbols that are quite common to indicate the male and female using the body rather than face.

1.3 Articulated Body Parts Based Approach

In this work, we present a new problem called **articulated body-based gender recognition**. To our best knowledge, this is for the first time to study the gender recognition problem using articulated human body images.

We developed a method to recognize gender from an articulated body. A systematic study of some critical issues in body-based gender recognition, such as how informative each body part is, how many body parts are needed, what are good representations of body parts, and how accurate the gender recognition system can achieve in challenging, unconstrained, real-world images.

To recognize gender from the articulated body, we detect the body parts first, which itself is a very challenging problem. Fortunately, recent research has made significant progress on human body detection and body part localization [10, 11, 8]. Then we propose

to normalize the detected body parts into standard sizes and orientation. Various feature descriptions can be used on each body part and combined together to execute gender recognition. The overall flow chart of our approach is shown in Fig 1.3.1. And a simple illustration of the flow chart is shown in Fig. 1.3.2.

Figure 1.3.1: Framework for gender recognition from articulated body.

Figure 1.3.2: An example of our framework for gender recognition from articulated body.

Chapter 2

Overview of Articulated Human Body Parts

The human body has many degrees of freedom (as illustrated by the joints in Fig. 2.0.1) to change its shape and perform different actions (e.g., in various sports games and even daily life activities). Thus there exist many possible variations in an articulated body, such as upper and lower arms rotations, head movements, torso bending, and leg kicking. Furthermore, body parts can occlude each other, or be occluded by other objects, depending on the view points. All these factors have to be considered in gender recognition from an articulated body.

Figure 2.0.1: An illustration of the many degrees of freedom for articulation in a human body.

A general paradigm in object recognition is to take the supervised learning approach. That is to say, some training images of each object class have to be collected in the learning stage in order to learn a classification function and then apply it to the testing

examples. For our problem of gender recognition from an articulated body, we also choose the procedure of supervised learning.

To employ a set of articulated bodies for visual learning of gender, the first and most important thing is how to align the images of articulated bodies. This is very critical in developing a system for gender recognition from articulated bodies. One cannot simply extract features from the given images without considering the alignment problem. As shown in the right of Fig. 1.1.1, it is not trivial to align the articulated bodies with various body pose variations and occlusions.

In order to align human bodies with various poses, we perform human body detection and body parts detection first. Then we “*normalize*” the detected body parts, e.g., torsos or upper arms, from different individuals into the same location and size in a common two-dimensional Cartesian coordinate system.

Another issue in body-based gender recognition in real-world images is that sometimes only partial human bodies are visible in the image frame, or some body parts are heavily occluded by other objects. To deal with “partial views” of human bodies, we attempt to use the upper-body only for gender recognition. Actually the upper-body conveys the majority of information for human perception of gender. Although the lower body may contain some information when visible, we do not use it in order to make our system general and robust to deal with many possible situations in reality. In fact, the lower bodies are frequently occluded by other objects or do not appear in the camera field of view (FOV) in many real-world images, but it will not be a problem for our system.

In the following sections, some simple description about upper body and part detection and normalization will be presented.

2.1 Upper-Body Detection and Part Localization

In movies, TV shows, or consumer photos, in many cases only the upper body is visible in the camera field of view. Thus detecting the upper body is more general and robust than the whole body detection. Our approach follows the human pose estimation algorithm of [8] which employs the Pictorial Structures framework. An illustration of the main steps of upper body detection and local part localization is shown in Fig. 2.1.1.

The scheme of using a sliding window followed by non-maximum suppression is adopted for upper body detection [11, 8]. In our approach, the part-based model [11] is adopted,

since it can do an excellent job in upper-body detection. Detections carry information about the rough position and scale of persons in the image. This reduces the search space by setting bounds on the possible (x, y) locations of the body parts and by fixing their scale, thus removing a dimension of the Pictorial Structures' state space entirely. In practice, for each detected person the state space is reduced to a region of the image around the detection, covering the possible arms extent of the person.

After the upper body is detected, we highlight the foreground. At this stage, the search for body parts is only limited by the maximum extent possible for a human of that scale centered on the detected position. And we apply some prior knowledge about the structure of the detection window to determine the area which contains parts of the person. Then an image segmentation can be performed using the Grabcut method [20] to separate the whole upper body from the background, see fig. 2.1.1(3).

Figure 2.1.1: Main steps of upper body detection and body parts localization. (1) Input image, (2) upper-body detection, (3) segmentation of the whole upper-body from background, and (4) body part inference.

Next step, we apply the pictorial structure model to estimate the location of each body parts of the upper body. the body parts of a person are usually represented as a tree in the pictorial structure model. Each node in the tree represents a body part, see Fig 2.1.2(b), where l_h represents head, l_t represents torso, l_{lua}, l_{rua} , l_{lla} , l_{rla} represent left

upper arm, right upper arm, left lower arm and right lower arm (view from the picture).

Generally, we use $L = \{l_i\}$ to represent the set of body parts which are used in the pictorial structure model. Each body part l_i is actually a rectangular image patch, as in Fig 2.1.2(c). The position of a body part is parametrized as location (x, y) , orientation θ and scale s .

Figure 2.1.2: Pictorial Structure of body parts. (b) Kinematic tree representation for body parts. Each node represents a body part, l_h -head, l_t - torso, l_{lua} , l_{rua} , l_{lla} , l_{rla} -left/right upper/lowerarms. (c) Cardboard representation for body parts. Each body part is represented by rectangular patch.

Given an image I , the posterior of a configuration of body parts $L = \{l_i\}$ in the image is given as:

$$P(L|I, \Theta) \propto \exp \left(\sum_{(i,j) \in E} f(l_i, l_j) + \sum_i g(I|l_i, \Theta) \right), \quad (2.1.1)$$

where the unary potential g is about the local image likelihood for a part in a particular position, the pairwise potential f is to model the prior about the relative position of parts,

and Θ denotes the parameters of the model.

2.2 Body Parts Normalization

After upper-body detection and body parts localization, the position of each body part in a two-dimensional image plane is known. However, for different images, the detected body parts may have different sizes at various scales and the body parts may have arbitrary articulations when performing distinct actions or interacting with varied objects. To learn gender from the example body parts with arbitrary articulations, and match with unseen human bodies, a method has to be developed that can align the same body parts, e.g., torso, but from different persons, at various scales, and with distinct articulations into a common coordinate. We call this important procedure *body parts normalization*. But how to perform the normalization?

Figure 2.2.1: An illustration of the body parts normalization process.

Suppose we have already represented each body part as a rectangular patch. If a body part is centered at O (see left upper arm in Fig 2.2.1(a)), we denote the two ends of a body part as U (upper side) and L (lower side), Y_1Y_2 is the vertical line pass through the center O . We define the angle $\theta = \angle ROY_1$ as rotation angle, as illustrated in Fig. 2.2.1(a). If the body part is the left upper arm or the left lower arm, we rotate the body part clockwise by an angle of θ around the center O ; If the body part is the right upper arm or the right lower arm, we rotate the body part counterclockwise by an angle of θ around the center O . After the rotation, the rectangle patch is changed to the vertical

direction. By performing the in-plane rotation for each body part rectangular patch, one can make all detected body parts have the same (vertical) orientation.

Next, We deal with the scale variations due to different viewing distances and distinct individuals. For any given image I_j , let us denote the six upper body parts rectangular patches as $R_i(I_j)$ ($i = 1, 2, \dots, 6$). The width (denote as $w_i(I_j)$) and height (denote as $h_i(I_j)$) of the image I_j are known from the body parts detection. Suppose the standard width and height of the body part i are given as w_i^s, h_i^s , compute the scaling factor for body part i as following:

$$\begin{cases} s_i^w(I_j) = w_i^s/w_i(I_j) \\ s_i^h = h_i^s/h_i(I_j), \end{cases} \quad (2.2.1)$$

and then apply the scaling factors $s_i^w(I_j)$ and $s_i^h(I_j)$ to the rectangular patch $R_i(I_j)$ of the body part. After this scaling operation, each body part is normalized into standard size.

The standard width w_i^s and height h_i^s for each body part patch can be pre-defined based on the human knowledge about body shapes and required image resolutions or adjusted from the image data. In our approach, we compute the mean values of each body part rectangular patch sizes,

$$\begin{cases} w_i^s = \frac{1}{n} \sum_{j=1}^n w_i(I_j), \\ h_i^s = \frac{1}{n} \sum_{j=1}^n h_i(I_j), \end{cases} \quad (2.2.2)$$

where $i = 1, 2, \dots, 6$ indicates the body part, and n is the total number of detected humans in the training set. We use the mean values as the standard sizes for each body part. Please note that different body parts may have different sizes as their standards.

In practice, the rotation and scaling are performed by image re-sampling. After the rotation and scale changes, we are done with the body parts normalization. There is no need to perform any translations, since we will extract features to represent each body part, rather than using raw image patches.

2.3 How Many Body Parts Are Needed?

Since it is for the first time to recognize gender through decomposition of the articulated body into local parts, it might be interesting to study how informative each body part is and how many body parts are needed to combine together for the purpose of gender recognition.

A key question is, do these body parts have equal contributions to gender recognition? Since no previous work has studied this problem, it is not clear to us (and other researchers) about how informative each body part will be. We perform an empirical study about this using various representations (see Chapter 3) and check the results in Chapter 5. The basic idea is to use each single body part for learning the classifiers and testing, and compare the recognition accuracies to measure the informativeness of each body part. We believe that this result is important and will provide practical guidance for body-part-based gender recognition.

Another key question is, how many body parts are needed for gender recognition? One may think that the more body parts to use, the higher the recognition accuracy will be. To verify whether this conjecture is true or not, we perform an empirical study by combining different body parts together based on various part representations. From our study (see results in Chapter 5), we found that the conjecture is not true. Why the gender recognition accuracy is not necessarily higher when using more body parts? There are three reasons, we think, to interpret it:

- (1) Some body parts are not as informative as others.
- (2) Some body parts cannot be detected reliably and accurately in practice, especially in a challenging image set.
- (3) Some body parts might be occluded by other parts or objects more frequently than others, which causes uncertainties in gender recognition.

As a result, one should use only the body parts that can be detected reliably and contain discriminative information, and combine them together for gender recognition.

The next question is, how to combine the selected body parts? One solution is to extract features from each body part, and then combine the features from all selected body parts. Another solution is to put all normalized parts into a fixed coordinate system, and then extract features from the warped and aligned body parts altogether. We used the first solution in our system, since it is easier to implement and may suppress noises

from the background.

Chapter 3

Feature Descriptors

After obtaining the normalized body parts, we need to extract some information from the normalized images to represent those images. In computer vision, the concept of feature is used to denote the piece of information which is useful for solving certain problems. Feature descriptors provide a means for comparing images or image regions. They are used in many different applications, such as object recognition, image alignment, 3D reconstruction, Motion tracking, etc. The most frequently used image feature descriptors are histograms of oriented gradients (HOG), scale invariant feature transform (SIFT), shape descriptors and color descriptors.

Since there is no previous work that has studied the articulated body or body parts based gender recognition, it is unknown about what kinds of features are useful and discriminative. To discover this in our study, we present some features that might be useful, compare the performance of these representations, and then find what representations are good for gender recognition in articulated body images.

In this section, we will have a brief review of the histogram of oriented gradient (HOG) [6], local binary patterns (LBP) [1], scale-invariant feature transform (SIFT) [16], and RGB colors features (where the histogram with 32 bins is computed for each color channel and concatenated). These features will be extracted from the normalized images to represent the body parts. These features were originally proposed for other computer vision problems. We will evaluate whether they are useful or not for our problem in section 5.

3.1 Histogram of Oriented Gradients (HOG)

Histogram of Oriented Gradient descriptors was first introduced by Navneet Dalal and Bill Triggs in their June 2005 paper [6], where they applied HOG to their algorithm to do the pedestrian detection in static images.

The main idea of the Histogram of Oriented Gradient descriptors is to represent the local object appearance and shape of an input image as the distribution of intensity gradients or edge directions. In practice, the HOG descriptors can be achieved as following:

(1) Divide the image into small patches. And these small patches are called cells. Cells can be either rectangular or radial.

(2) For each cell, obtain a histogram of gradient directions or edge orientations for the pixels within the cell.

Then we combine these histograms from all cells into one vector to represent the image.

Figure 3.1.1: process of getting HOG descriptor

In order to make the HOG descriptor invariant to changes in illumination or shadowing, an energy over a larger patch of the image (called a block) is defined and measured, and then use the energies to normalized all cells in the block. The normalized descriptor blocks are call Histogram of Oriented Gradient descriptors.

The most popular way to compute the value of gradient of the image intensity function,

is to apply the 1-D centered, discrete filter in one or both of the horizontal and vertical direction. The filter kernels corresponding to the horizontal direction is $[-1, 0, 1]$. And the filter kernels corresponding to the vertical direction is $[-1, 0, 1]^T$. More complex masks, e.g. 3×3 Sobel operator, can also be used to get an approximation of the gradient. Dalal and Triggs also tried to compute the gradient with Gaussian smoothing before applying a discrete filter. Their experiment results show that using larger masks always decrease performance and smoothing damages it significantly. For color images, the gradient of each color channel is computed separately, and the one with the largest norm is used as the pixel's gradient.

To get the histogram of gradient directions, gradient orientations, which is usually spread over 0 to π in the case of unsigned gradient or spread over 0 to 2π in the case of signed gradient, are evenly divided into several bins. At each pixel, a weighted vote is assigned for an edge orientation histogram bin. Then the weighted votes for all pixel in a cell are accumulated into a histogram. For the purpose of reducing aliasing, Votes are interpolated bi-linearly between the neighboring bins centered in both orientation and position. The vote function is a function of the magnitude of gradient, such as the magnitude its self, square root of the magnitude or square of the magnitude. In Dalal and Triggs's paper, they mentioned that the best result is achieved by using the magnitude itself.

3.2 Local Binary Patterns (LBP)

Texture information in an image is an important information for some image processing problems. Local Binary Patterns It is a simple yet powerful feature for texture classification. It was first introduced by T. Ojala et al.[28] in 1994. At each pixel in an image, LBP compute the difference between the pixel and its neighborhood, then represent the result as a binary number.

The process of getting the LBP feature vector can be organized as following steps:

Step 1. Partition the input image into small sub-images (see Fig 3.2.1(a)).

Step 2. For each pixel in a sub-image, compare the intensity of the pixel to each of its eight neighbors (left-top, left-middle, left-bottom, middle-top, middle-bottom, right-top, right-middle and right-bottom). For any neighbor of the center pixel, if its intensity is less than the intensity of the center pixel, then one is assigned to the neighbor. Otherwise,

zero is assigned to the neighbor (see Fig 3.2.1(c)).

Step 3. Following the center pixel along a circle (clockwise or counter-clockwise), write down the 8 binary values assigned to its eight neighbors . This eight digital number is usually converted to a decimal by some function and is assigned to the center pixel, see Fig 3.2.1(c).

Step 4. Compute the histogram for the sub-image based on the value assigned to the center pixel. i.e., count how many times each "number" occurring (see Fig 3.2.1(c)).

Step 5. Normalize the histogram.

Step 6. Concatenate the normalized histograms of all sub-images to get the feature vector for the image.

Due to its computational simplicity and discriminative power, LBP descriptor has become a very popular approach in computer vision problems such as face analysis.

3.3 Scale Invariant Feature Transform (SIFT)

Scale Invariant Feature Transform (SIFT) is an approach for detecting and extracting local feature descriptors based on the appearance of the object at particular interest points. Lowe presented the Scale Invariant Feature Transform and its application for image matching and recognition in his paper in 2004[16]. SIFT features are invariant to change in illumination, image noise, rotation, scaling and change in 3D viewpoint. In addition to these properties, the SIFT features are highly distinctive, which allows a single feature to find its correct match from a large data set with high confidence.

The algorithm for extracting the SIFT descriptors could be summarized as following four steps:

1. Scale-space extrema detection.
2. Keypoint localization.
3. Orientation assignment.
4. Generation of keypoint descriptors.

The scale-space extrema in the convolution of difference of Gaussians (DoG) and the intensity of the image are used to find the locations which are invariant to scale change. Mathematically, the convolution could be expressed as:

Figure 3.2.1: flow of getting LBP descriptor

$$D(\mathbf{x}, \sigma) = [G(\mathbf{x}, k\sigma) - G(\mathbf{x}, \sigma)] * I(\mathbf{x})$$

where $D(\mathbf{x}, \sigma)$ is the Gaussian function, $\mathbf{x}$ is the location of the pixel, σ is the scale, k is the scale factor, and $I(\mathbf{x})$ is the intensity function of the image.

If the value of $D(\mathbf{x}, \sigma)$ is a local maximum or a local minimum at a pixel, this pixel is selected as a keypoint candidate.

Once a keypoint candidate is found, a further examination is applied to determine whether this candidate will be accept or reject as a keypoint based on their stability. Interpolation of the data around the keypoint candidate is applied to determine the accurate location of the keypoint. By shifting $\mathbf{x}$ to the origin, the Taylor expansion (up to the quadratic terms) of $D(\mathbf{x}, \sigma)$ is expressed as

$$T(\mathbf{t}) = D + \frac{\partial D^T}{\partial \mathbf{t}} \mathbf{t} + \frac{1}{2} \mathbf{t}^T \frac{\partial^2 D^T}{\partial \mathbf{t}^2} \mathbf{t}$$

where $\mathbf{t}$ is the offset from the pixel $\mathbf{x}$. If we take the derivative of $T(\mathbf{t})$ and set it to zero, the location of the extremum (denote as $\tilde{\mathbf{t}}$) is given as

$$\tilde{\mathbf{t}} = -\frac{\partial^2 D^{-1}}{\partial \mathbf{t}^2} \frac{\partial D^T}{\partial \mathbf{t}}.$$

If $\tilde{\mathbf{t}}$ is larger than some threshold (e.g., 0.5), it implies that the location of the extremum is closer to some other neighboring sample point which is used to calculate $\frac{\partial D}{\partial \mathbf{t}}$ and $\frac{\partial^2 D}{\partial \mathbf{t}^2}$. In that case, a new neighbor is used to perform the interpolation and get the new estimation for the keypoint.

If the value of $|T(\tilde{\mathbf{t}})|$ is less than a threshold (e.g., 0.03), then the keypoint candidate will be discarded.

To determine the orientation of the keypoint, an orientation histogram is computed based on the gradient orientations of neighbors around the keypoint. The gradient magnitude of each neighbor is used as the weight of the neighbor and added to the orientation histogram. The absolute maximum in the orientation histogram correspond to the dominant directions and it will be deleted from the histogram. Any local maximum within 80% of the absolute maximum is used to create a keypoint with that orientation.

Once the location and orientation of a keypoint is determined, the key point descriptor is computed as a set of orientation histograms on 4×4 neighborhoods. And 360 degrees are divided into eight parts and used for the orientation histograms. A 16×16 square window around detected feature is chosen and is divided into 4×4 sub-regions. For each sub-region, compute gradient direction over the 8 directions for each sub-region. The SIFT feature vector is obtained by concatenating the gradient histogram of the 4×4 sub-regions. Since each histogram contains eight bins, the length of the SIFT feature vector is $4 \times 4 \times 8 = 128$. This vector is normalized to enhance invariance to changes in illumination.

In our application, we run SIFT on a dense grid of locations at a fixed scale and orientation, which means that we will skip step 1 and step 2 of the algorithm for extracting the SIFT descriptors.

Chapter 4

Supervised Machine Learning

Supervised machine learning is the machine learning task of learning a predict function from the training samples. Each training sample consists an vector representation of an object and its class (called label which is usually provided manually by some experts). The main task of supervised machine learning is to predict the class of the testing sample based on the properties learned from the training samples. The generalized procedure of supervised machine learning is shown in Fig 4.0.1.

A wide range of supervised learning methods, such as support vector machine (SVM) and artificial neural network, have been introduced in the last decade. We will present the Support Vector Machine (SVM) and Partial Least Squares in this chapter since we are going to use SVM and PLS to do the classification in later chapters.

4.1 Support Vector Machine

A support vector machine (SVM) is a supervised learning method which analyzes data and recognize patterns. It is introduced by by Cortes and Vapnik [30]. SVM has been widely used as a classification tool with a great success.

Given n input data, say $x_i \in R^N$, and corresponding binary prediction $y_i \in \{-1, 1\}$, a support vector machine compute a maximum-margin hyper-plane in a higher dimensional space such that:

$$\begin{cases} x_i \cdot w + b \geq 1, & \text{if } y_i = 1 \\ x_i \cdot w + b \leq -1, & \text{if } y_i = -1 \end{cases}$$

Figure 4.0.1: General procedure of Supervised Machine Learning

1

Figure 4.1.1: Nonlinear Kernel

To find the maximum-margin hyper-plane, it is equivalent to solve w and b for the following problem

$$\text{minimize } P(w, b) = w^T w / 2$$

$$\text{s. t. } y_i [w^T x_i + b] \geq 1$$

For some training samples there is no separating hyperplane, then we could try soft margin, which is to solve w and b for the following problem:

¹change images for SVM.

minimize $P(w, b, \xi) = w^T w / 2 + C \sum_{i=1}^n \xi_i$

s. t. $y_i[w^T x_i + b] \geq 1 - \xi_i$ with $\xi_i \geq 0$

where $\sum_{i=1}^n \xi_i$ is an upper bound on the number of training errors. C is a parameter that controls trade-off between margin and error.

Figure 4.1.2: Nonlinear Kernel

Some classification problems may be very complicated and have a non-linear structure and no hyperplane is sufficiently accurate (see figure 4.1.2(a)). In such case, kernels are introduced to project original features to higher dimensional space so that the projected features are linear separable in the new space (see figure 4.1.2(b)). The following are some commonly used kernels:

(1) Linear Kernel: $K(x, z) = x^T z + c$;

(1) Gaussian Radial basis function (RBF): $K(x, z) = e^{-||x-z||^2/2\sigma^2}$

(2) Polynomial Kernel: $K(x, z) = (c_1 x^T z + c_2)^p$;

(3) Chi-Square Kernel: $K(x, z) = 1 - \sum_{i=1}^n \frac{2(x_i - z_i)^2}{(x_i + z_i)}$.

The final classification rule is quite simple:

$$\hat{y}_t = b + \sum_{s \in S} w_s K(x_t, x_s)$$

where S is the set of support vectors, x_t is the testing data, x_s is one of the support vector in the training set. $\hat{y}_t$ is the classification result for the testing data x_t .

In this work, we tried different kernels in our experiments and find that the RBF performs best for our problem.

4.2 Partial Least Squares (PLS)

Partial least squares (PLS) regression is a statistical method to find a linear regression model for the independent variables and dependent variables. It is realized by projecting the independent variables and dependent variables to new spaces. PLS model tries to find the multidimensional direction in the space of the independent variables that explains the maximum variance direction in the space of the dependent variables. The PLS method simultaneously perform the dimension reduction and regression.

Partial least squares was introduced by Wold in 1985 [26]. Recently, the Partial Least Squares (PLS) method [25, 19] has shown good performance in some computer vision problems, such as pedestrian detection [22], face recognition [21] and age estimation [14]. Inspired by this, we are interested in evaluating the performance of the PLS method in the new problem of articulated-body-based gender recognition.

In previous applications to solve computer vision problems, the PLS was used mainly for dimensionality reduction [22, 21]. Guo and Mu [14] adapted the PLS to age estimation and showed the successful use of the PLS for both feature dimensionality reduction and age regression. Actually the PLS method itself can do classification [19], which has seldom been explored in previous approaches. Here we want to evaluate the PLS's capability of both dimensionality reduction and classification.

The linear PLS algorithm [19, 26] is to model the relation between two sets of variables. Denote by $\mathcal{X} \subset \mathcal{R}^N$ an N -dimensional latent space of variables representing the first block and similarly by $\mathcal{Y} \subset \mathcal{R}^M$ a space representing the second block of variables. PLS models the relations between these two blocks by means of score vectors. It decomposes the $(n \times N)$ matrix of zero-mean independent variables $\mathbf{X}$ and the $(n \times M)$ matrix of zero-mean dependent variables $\mathbf{Y}$ into the form

$$\begin{aligned} \mathbf{X} &= \mathbf{T}\mathbf{P}^T + \mathbf{E} \\ \mathbf{Y} &= \mathbf{U}\mathbf{Q}^T + \mathbf{F} \end{aligned} \tag{4.2.1}$$

$$\mathbf{U} = \mathbf{T}\mathbf{D} + \mathbf{H} \tag{4.2.2}$$

where $\mathbf{T}$ and $\mathbf{U}$ are $(n \times r)$ matrices of the r extracted score vectors (latent vectors), the $(N \times r)$ matrix $\mathbf{P}$ and the $(M \times r)$ matrix $\mathbf{Q}$ represent matrices of loadings, and the $(n \times N)$ matrix $\mathbf{E}$ and the $(n \times M)$ matrix $\mathbf{F}$ are the matrices of residuals. $\mathbf{D}$ is a $r \times r$

diagonal matrix which relates the latent scores of X and Y . The PLS method, which in its classical form is based on the nonlinear iterative partial least squares (NIPALS) algorithm [26], finds weight vectors $\mathbf{w}, \mathbf{c}$ such that

$$\begin{aligned} [\text{cov}(\mathbf{t}, \mathbf{u})]^2 &= [\text{cov}(\mathbf{X}\mathbf{w}, \mathbf{Y}\mathbf{c})]^2 \\ &= \max_{|r|=|s|=1} [\text{cov}(\mathbf{X}\mathbf{r}, \mathbf{Y}\mathbf{s})]^2 \end{aligned} \quad (4.2.3)$$

where $\text{cov}(\mathbf{t}, \mathbf{u}) = \frac{\mathbf{t}^T \mathbf{u}}{n}$ denotes the sample covariance between the score vectors $\mathbf{t}$ and $\mathbf{u}$. The NIPALS algorithm starts with random initialization of the $\mathcal{Y}$ -space score vector $\mathbf{u}$ and repeats a sequence of iterations until convergence [26].

In our gender recognition, $\mathbf{X}$ denotes the body part representations, while $\mathbf{Y}$ is reduced to a scale value of class labels. After the dimension reduction, X is reduced to

$$\hat{X} = XW$$

where W is the $(N \times r)$ Projection matrix from X to the latent space.

For an input x_t, y_t is estimated as

$$\hat{y}_t = (x_t W) Z^T$$

where Z is a $(M \times r)$ matrix representing the linear transformation from the r dimensional latent space to Y .

An obvious question is to find the number of latent variables needed to obtain the best generalization for the prediction of new observations. This is, in general, achieved by cross-validation techniques such as bootstrapping.

In the following chapter, we will validate two things of the PLS method:

- (a) How efficient for dimensionality reduction?
- (b) How accurate for classification without using the SVMs?

4.3 Comparison between SVM and PLS

As we mentioned in previous sections, the dependent variables Y in PLS is $(n \times M)$ matrix, i.e., the output could be multiple classes, multiple observations. On the contrary, SVM is mainly used for binary classification. In order to deal with multiple classes, the

dominating approach is to reduce the single Multi-class problem into multiple binary classification problems. In this sense, PLS could be applied more widely to different applications.

PLS is mainly used for dimension reduction, but it can also be applied to classification. Comparing the classification rules of SVM and PLS,

$$\begin{aligned}\hat{y}_t &= b + \sum_{s \in \mathcal{S}} w_s K(x_t, x_s), & (SVM) \\ \hat{y}_t &= (x_t W) Z^T, & (PLS)\end{aligned}$$

It is easy to see that, for the same input x_t , the computation by PLS classification is much less than the computation by SVM classification, especially there are many support vectors.

We also propose an PLS and SVM combined classification scheme, which is shown in Algorithm 4.1.

Algorithm 4.1 PLS and SVM combined classification

Step-1: We first use PLS to do the dimension reduction, we get the reduced training data

$$\bar{X} = XW$$

Step-2: Use the dimension reduction results $\bar{X}$ and Y as the input of the SVM machine.

Step-3: Learn support vectors $\bar{x}_s, \bar{b}, \bar{w}_s$ from $\bar{X}$ and Y .

Step-4: For input testing data x_t , we reduce the dimension of x_t to $\bar{x}_t = x_t W$.

Step-5: Find the class for the testing data based on the rule $\hat{y}_t = \bar{b} + \sum_{s \in \mathcal{S}} \bar{w}_s K(\bar{x}_t, \bar{x}_s)$.

4.4 Data Fusion

In our gender recognition in unconstrained and articulated body images, various noise and uncertainties exist, e.g., the body parts might be localized not accurately, or occluded by other body parts or objects; the illumination may change significantly; images are highly cluttered; contrast may be poor; motion blur may present because of camera movement or shaking, etc. If we apply the data fusion idea to a recognition system, we would expect that the recognition system be more robust, more confident, and even with higher accuracy.

To make our system more robust and possibly with higher accuracy, we also pursue a data fusion idea, and validate if data fusion could be helpful for our problem.

Data fusion is the process of putting together different data from different sources into a single composite data to improve the performance of a system. By combining different sources of data, inferences could be achieved, and the information is more accurate than they were achieved when using data only from single source.

Data fusion is widely applied to computer vision problems such as detection, recognition, identification, decision making, etc. By using an efficient fusion scheme, one may expect significant improvement on their results due to the use of complementary information.

Based on the processing stage at which fusion takes place, fusion processes are often divided into three fusion levels: low level, intermediate level or high level.

Low level fusion is the simplest one. It combines raw data from different sources into new raw data that is expected to be more informative than the inputs.

Intermediate level fusion, which is also called feature level fusion, combines various features obtained from different raw data sources (several sensors, different moments, etc.) or combine different features extracted from the same raw data. By combining different features extracted from the same raw data together, we are able to find relevant features among available features that might come from several feature extraction methods. The objective is to obtain a limited number of relevant features.

High level fusion, which is also called decision fusion, combines decisions coming from several experts.. Methods of decision fusion include voting methods, statistical methods, fuzzy logic based methods, etc. In Table 4.1, we show an example of high level fusion on the sample S_1, S_2, S_3, S_4 . The decision based on the data set A_i is shown in the last column of Table 4.1(a), The decision based on the data set A_i is shown in the last column of Table 4.1(b), The decision based on the data set A_i is shown in the last column of Table 4.1(c). We apply a simple voting scheme to the three datasets: the vote result is the same as the majority decision from the three data sets. Say for example, for the sample S_1 , its decision based on the dataset $\{A_i\}$ is 'Y', its decision based on the dataset $\{B_i\}$ is 'N' and its decision based on the dataset $\{C_i\}$ is 'N'. Since two results are 'N' and only one result is 'Y', the majority is 'N' and we finally label the fusion result as 'N', as in the last column of Table 4.1(d).

	A_1	A_2	A_3	Decision		B_1	B_2	Decision		C_1	C_2	Decision		Decision
S_1	1	x	0.5	Y	S_1	a	100	N	S_1	4	-1	N	S_1	N
S_2	2	y	0.7	N	S_2	c	80	Y	S_2	3	1	N	S_2	N
S_3	3	z	1.5	Y	S_3	d	105	Y	S_3	5	-1	N	S_3	Y
S_4	4	x	2.8	Y	S_4	a	50	N	S_4	8	1	Y	S_4	Y
(a)					(b)				(c)				(d)	

Table 4.1: An example of low level fusion

Thanks to the multiple representations of the data (see chapter 3), we can explore data fusion schemes by combining different representations of body parts. We investigate two schemes. One is to the feature-level fusion, which combines the features from different body parts together into a long feature vector, and then use the SVM for training and classification. The other is a decision level fusion, which combines the classification results of several representations to make a new decision by majority voting, i.e, given the data x , the final decision $D(x)$ is based on a voting from multiple classifiers, c_i , $i = 1, 2, \dots, m$. Suppose m is an odd number, $k = \frac{m+1}{2}$, then the decision $D(x)$ will take the classification result of at least k classifiers agree with.

We will examine both of the feature-level fusion and decision-level fusion , and compare their results to find out which one is better in which case of our problem.

Chapter 5

Experiments and Discussion

5.1 Datasets Description

We applied our articulated body part based gender recognition method on the following two data sets that contain unconstrained and articulated human body images. The Buffy data set could be downloaded from [\[36\]](#) and the PASCAL data set can be downloaded from [\[37\]](#).

(1) Buffy dataset: it contains video frames from the “Buffy: The Vampire Slayer” TV show, episodes 2 to 6, season 5. There are 748 frames in this data set.

(2) The PASCAL VOC 2008 challenge [\[9\]](#), it consists of 538 amateur photos (317 males, 221 females).

The annotated stickmen on both data sets are also provided with the image data.

The images of the Buffy data set is from TV show, they are in uncontrolled conditions with very cluttered background, dark illumination, and persons appearing with a large range of scale changes. The photos in the PASCAL data set were taken by amateur photographers, most of these photos are with difficult illumination, low image quality, and many different articulations. Both databases are public available and used for human pose estimation. But these two databases have not been evaluated for gender recognition yet. Some example images are shown in Fig. 5.1.1. Some images are taken from the back side and some are with very low resolution.

Both databases are challenging for human body detection and parts localization [\[8\]](#), while our focus is to recognize gender in those unconstrained and articulated body images.

Figure 5.1.1: Some example images in Buffy (top) and PASCAL (bottom).

To our best knowledge, this is for the first time to recognize gender from the articulated body and perform experiments on these two databases. Some example images in Buffy (top) and PASCAL (bottom) are show in Fig 5.1.1.

In order to understand the gender recognition performance in details, we experimented in several cases:

Case I: the body parts are known, i.e., the location of each body part is annotated manually.

Case II: the body parts are unknown, but detected automatically by body parts estimation.

In case I, our purpose is to show how accurate our gender recognition methods can achieve, without the influence of incorrect body part localization. In case II, we want to show the performance of an automated system that can be developed for real applications.

We first use SVM to evaluate the accuracy of gender recognition for the two data sets. And standard four-fold cross-validation scheme is used to evaluate all results under various experimental conditions.

5.2 Gender Recognition Results with Annotated Body Parts

To avoid the possible influence of incorrect body part detection, we first use the annotated stickmen to find the body parts in evaluating the gender recognition performance. We study which representations are better, how informative each body part is, how many body parts are needed to combine together to achieve a higher recognition accuracy and how to use data fusion schemes to further improve the accuracy.

Given an image and its annotated stickman, we determine the rectangular patch of each body part as following: for head and torso, we assign the length of the stickmen as both width and height of the rectangle; for the limbs, we choose the length of the stickman as height and let the width to be half of the height. Then we normalize each body part into a standard size and orientation using the method presented in Section 2.2. To determine the standard size for each body part, we computed the mean values of each body part size from the training data. Here we set the standard size of each body part to be an integer which is close to the mean value of the body part. The standard sizes to normalize each body part are list in the table 5.1. One can also see that the parts are smaller in PASCAL than Buffy. Remember that we also need to rotate each body part into the vertical direction for both the training and test body parts.

	Buffy			PASCAL	
	length	width		length	width
torso	170	170		80	80
head	90	90		40	40
upper left arm	100	50		60	30
upper right arm	100	50		60	30
lower left arm	70	35		60	30
lower right arm	70	35		60	30

Table 5.1: standard size for each body part

We extracted four different features (HOG, LBP, SIFT and RGB) for each body part. Dense sampling were used with the same intervals for HOG, LBP, and SIFT. For RGB, we compute the color histograms of 32 bins in each color channel and then concatenate the three (R, G, B) channels into a feature vector. Different results based on different body parts and feature descriptor for Buffy and PASCAL data sets are shown in Table

5.2 and 5.3, respectively. Now we discuss the result based on different aspects in gender recognition.

(1) Which feature descriptor works better?

From the first four rows of Table 5.2 and 5.3, one can see that the color features using RGB histograms work best on Buffy, but worst on PASCAL. The other three features, HOG, LBP, and SIFT, can work similarly well on both databases, with the SIFT slightly worse on Buffy.

Table 5.2: Gender recognition with annotated body parts on Buffy. The abbreviations are: T for torso, H for head, LUA for left upper arm, RUA for right upper arm, LLA for left lower arm, and RLA for right lower arm. They are also used in other tables and figs.

	RLA	LLA	RUA	LUA	T	H	H+T	H+T +LUA	H+T +RUA	H+T +RUA+LUA
HOG	83.32	85.40	91.26	88.37	90.60	92.23	93.15	92.34	94.23	94.11
LBP	75.58	79.48	94.19	92.14	95.07	94.23	96.12	96.12	96.12	96.12
SIFT	69.39	75.40	85.52	84.28	82.16	92.07	93.18	93.14	93.18	93.18
RGB	81.70	83.29	92.22	89.33	90.41	91.26	97.12	96.19	95.26	94.22
F: Decision-Level	85.33	88.33	92.19	94.19	97.12	96.15	95.15	96.12	94.23	96.12
F: Feature-Level	78.67	84.45	96.19	92.22	96.12	95.19	98.04	97.12	98.00	97.12

Table 5.3: Gender recognition with annotated body parts on PASCAL.

	RLA	LLA	RUA	LUA	T	H	H+T	H+T +LUA	H+T +RUA	H+T +RUA+LUA
HOG	58.90	58.90	62.40	60.25	59.40	67.87	67.99	68.28	71.03	70.93
LBP	58.90	58.90	64.80	63.96	63.78	67.68	67.66	68.90	70.92	71.62
SIFT	58.90	58.90	64.16	62.15	59.06	65.49	66.18	68.35	70.07	69.88
RGB	58.90	58.90	58.90	58.90	58.90	58.90	58.90	59.36	59.97	59.09
F: Decision-Level	58.90	58.90	67.43	64.51	64.73	69.55	70.62	70.99	73.82	73.42
F: Feature-Level	58.90	58.99	65.30	63.21	64.60	67.83	69.08	71.34	72.30	73.14

(2) How Informative is Each Body Part?

If we look at the first two columns in Table 5.3, one can find that (both left and right) lower arms cannot separate the male from female, while other body part, such as upper

Figure 5.2.1: Gender recognition results with annotated parts on Buffy (left) and PASCAL (right).

arms, head or torso works relatively better (column 3 to column 6 in Table 5.3) with the representations of HOG, LBP, and SIFT. The lower arms show higher accuracies on Buffy (Table 5.2) than on PASCAL (Table 5.3), but their accuracies are still significantly lower than other body parts. So the lower arms are not informative body parts for gender recognition. Further, we also observe that using single body part is not a good strategy and not robust. So it is better to combine several body parts together based on the discriminative power for each of them.

(3) How Many Body Parts are Needed for Best Result?

Combining several body parts together will make it more robust and have a better gender recognition performance. If we look at column 7 to column 10 in Table 5.2 or Fig. 5.3, we can easily see that the effect of combining body parts on PASCAL. Since the lower arms are not discriminative, we do not use them in part combination. Although some single body parts already have high accuracies on Buffy, combining them can still improve the recognition accuracy further (see Table 5.2). Interestingly, having more parts not necessarily delivers a higher accuracy. That is why we need to study which parts to combine together, rather than using all available parts.

(4) Is Data Fusion Schemes better?

As stated in section 4.4, we will evaluate two schemes for data fusion. One is the feature-level data fusion, which combines different feature vector to get a new feature vector to represent the input image, and then do the classification based on the combined feature. The other is the decision level data fusion which is to do the majority voting based on the classification result of each separate feature. The results of the two data fusion schemes are shown in Tables 5.2, 5.3 (the last two rows) and Fig. 5.2.1. In Buffy, we use the three best features, i.e., RGB, LBP and HOG for fusion. If the SIFT feature is added into data fusion, the result becomes worse (not shown here). The feature-level fusion scheme gets the highest accuracy of 98.04% using only two body parts: head and torso. It is slightly higher than the decision-level fusion with an accuracy of 97.12% using only one body part torso, which has the same value as using the single feature of RGB on head and torso.

In PASCAL, the best three features are HOG, LBP and SIFT. So we use these three features to do the data fusion. The experiment results show that the scheme of decision-level fusion gets the highest accuracy (73.82%) on three body parts (head, torso and right upper arm), which is slightly higher than the 73.14% accuracy using feature-level fusion on four body parts (head, torso left upper arm and right upper arm). The recognition accuracies of both fusion schemes are higher than the best (71.62%) when using single features (LBP feature with four body parts: head, torso, and upper arms, see the last column of row 2 in Table 5.3).

5.3 Gender Recognition Results with Automated Body Parts

In the second case, we evaluate the performance of the **automated** gender recognition system that we developed. Instead of using annotated body parts as described in section 5.2, the automated gender recognition system detects the upper body and related body parts automatically for any input image *without using part annotations* in either gender learning or testing. Because the body parts are detected by some body parts detection system, the body parts might be detected imprecisely or incorrectly. Here we want to verify how robust and accurate our system can achieve in automatic gender recognition. The standard four-fold cross validation is used again to measure the performance of the

automated gender recognition system.

Similar to gender recognition using annotated body parts, we evaluate several important things in automatic gender recognition, including body part representations, how informative each body part is, how many body parts to be combined together and how to apply data fusion schemes. The experimental results are shown in Table 5.4, Table 5.5 and Fig. 5.3.1.

Table 5.4 and Fig. 5.3.1 (left) show the recognition performance on Buffy. If we only use single feature on each body part, the highest accuracy that we can achieve is 90.22% by using the LBP feature on torso. If we try different body parts combination, the highest accuracy we can achieve is 93.30% by using the LBP feature on parts of head, torso and left upper arm. From Table 5.4, we find that the accuracy by based on the SIFT feature is lower than the accuracy based on the other three features in most cases. So the data fusion scheme is based on three feature descriptors: RGB, LBP and HOG. The feature-level fusion can achieve the highest accuracy of 96.11% (with head, torso, and right upper arm), which is better than the decision-level fusion of 94.19% on four body parts. Based on these experiment results, we conclude that the data fusion can improve the accuracy and the feature-level fusion scheme is better than the decision-level fusion. Comparing Table 5.4 with Table 5.2, the accuracy of automatic gender recognition (96.11%) is not as good as the accuracy of annotation-based (98.04%), but the difference is very small. This tells that our system is pretty robust with high accuracy in automatic gender recognition on Buffy.

Table 5.5 and Fig. 5.3.1 (right) show the automatic recognition performance on PASCAL. If we use single feature on each body part separately, the highest accuracy we could achieve is 66.29%, which is by the HOG feature on head. If we combine different body parts together, the highest accuracy could be achieved is 66.14% by the LBP feature on parts of head, torso and right upper arm. The result based on the RGB feature has lower accuracy comparing with the other three features in most cases. So the data fusion is applied on the following three features: LBP, HOG, and SIFT. The decision-level fusion can achieve the highest accuracy of 67.74% (with head, torso and right upper arm), which is better than the best accuracy (66.44%) achieved by feature-level fusion on four body parts. Based on the experimental results, we observed that: (1) Data fusion can improve the accuracy; (2) the decision-level scheme is better than the feature-level scheme. Comparing Table 5.5 with Table 5.3, the accuracy of automatic gender recognition (67.74%) is

lower than the annotation-based (73.82%), but the difference is not very big. This shows that our approach is pretty robust in automatic gender recognition on PASCAL. On the other hand, the recognition accuracy on PASCAL is much lower than the accuracy on Buffy, this is because that the PASCAL database is much more challenging than Buffy. We think that more effort is needed on PASCAL data set and our study may inspire new research work.

Table 5.4: Automated gender recognition on Buffy.

	RLA	LLA	RUA	LUA	T	H	H+T	H+T +LUA	H+T +RUA	H+T +RUA+LUA
HOG	68.53	71.50	87.18	81.40	90.18	84.44	92.29	93.26	92.10	92.26
LBP	65.94	73.44	86.18	83.63	90.22	80.62	92.34	93.30	91.22	92.11
SIFT	66.84	65.91	78.74	73.67	79.24	79.47	83.47	87.40	87.40	89.37
RGB	80.39	69.46	93.18	84.33	86.48	89.22	91.33	92.30	89.37	92.33
F: Decision-Level	70.76	74.36	92.99	81.43	92.26	86.33	95.23	91.26	93.26	94.19
F: Feature-Level	67.95	74.44	89.99	84.60	92.18	84.52	93.26	93.15	96.11	94.26

Table 5.5: Automated gender recognition on PASCAL.

	RLA	LLA	RUA	LUA	T	H	H+T	H+T +LUA	H+T +RUA	H+T +RUA+LUA
HOG	58.90	58.90	61.63	63.98	58.90	66.29	60.90	62.53	63.39	63.79
LBP	58.90	58.90	64.28	59.93	58.90	63.89	63.90	63.75	66.14	65.84
SIFT	59.37	61.11	59.60	60.55	60.62	62.33	63.05	64.33	65.32	65.04
RGB	58.90	58.90	58.90	58.90	58.66	58.90	58.90	58.90	58.90	58.90
F: Decision-Level	58.90	58.90	64.21	63.07	58.90	64.93	65.69	64.78	67.74	66.72
F: Feature-Level	58.90	60.81	63.91	64.00	59.40	64.34	63.21	65.52	66.26	66.44

5.4 Results Based on the PLS Method

The PLS method can be used for dimensionality reduction and also classification. We will evaluate these two properties in our problem separately. Here we only show the results using annotated body parts on PASCAL. Similar performance was observed in automated recognition (not shown here). The experimental results are given in Table 5.6, where “*PLS_1*” denotes that the PLS is used just for dimensionality reduction and “*PLS_2*”

Figure 5.3.1: Automated gender recognition on Buffy (left) and PASCAL (right).

denotes that the PLS is used for both dimensionality reduction and classification which avoids the use of SVMs. We found that the PLS method fail to work on SIFT features. One possible reason is that dense SIFT produced too many zeros which prohibits the PLS computation. But the PLS can work on other features without any problem. When single feature is used, the recognition accuracy is 70.11% by the approach of HOG + PLS + SVM on three body parts, or 69.41% by the LBP + PLS (no SVM classification) on the same three body parts. When the HOG and LBP features are combined together, the PLS can get an accuracy of 70.57% on three body parts. Overall, the accuracy based on PLS is slightly lower than the one without using PLS (see Table 5.3), but the difference is small. Please note that, no matter what the original feature dimension is (see Table 5.7), the PLS method can efficiently reduce the dimension to a small number of **40**. Empirically we found that using this small number of latent variables is sufficient, even if the original dimension could be as high as 4,355 or 5,360 as shown in Table 5.7. For time-critical or real-time applications, the PLS method is a good choice to obtain an extremely low dimension and even avoid the use of complex classifiers (e.g., SVMs) in recognition.

5.5 Summary of Gender Recognition Results

We have performed comprehensive experiments on gender recognition from unconstrained and articulated human body on two databases, the Buffy and PASCAL. Four different

Table 5.6: Gender recognition using the PLS method for dimensionality reduction (denoted by PLS_1) or both dimensionality reduction and classification (denoted by PLS_2) on PASCAL data with annotated body parts.

	RLA	LLA	RUA	LUA	T	H	H+T	H+T +LUA	H+T +RUA	H+LUA +T+RUA	ALL
HOG_PLS_1	58.90	58.90	62.38	60.28	58.90	65.23	65.21	65.97	70.11	69.56	67.12
LBP_PLS_1	58.90	58.90	62.30	60.06	62.59	65.25	67.28	67.36	69.93	69.01	68.11
RGB_PLS_1	57.37	58.91	58.70	58.12	57.73	58.52	58.90	58.90	58.20	58.90	58.90
HOG_PLS_2	52.82	48.89	56.85	56.53	57.02	62.04	59.64	58.05	58.59	58.73	56.65
LBP_PLS_2	50.98	50.69	54.95	53.00	57.84	63.09	65.40	66.69	69.41	68.72	65.72
RGB_PLS_2	53.27	51.64	58.58	50.66	58.27	56.04	52.66	53.97	54.10	53.84	51.80
HOG+LBP_1	58.90	58.90	63.10	61.59	63.19	67.09	66.40	68.14	70.57	69.08	68.18
HOG+LBP_2	48.07	50.50	55.70	54.08	60.25	66.11	67.67	67.59	69.80	69.57	66.98

Table 5.7: Feature dimensions with various setups on PASCAL. No matter what the original dimensions are, the PLS method can efficiently reduce each number to a small value of **40**, which is very useful for real-time applications.

	RLA	LLA	RUA	LUA	T	H	H+T	H+T +LUA	H+T +RUA	H+T +RUA+LUA
HOG	120	120	120	120	200	200	400	520	520	640
LBP	885	885	885	885	1475	1475	2950	3835	3835	4720
SIFT	1920	1920	1920	1920	3200	3200	6400	8320	8320	10240
RGB	96	96	96	96	96	96	192	288	288	384
HOG+LBP	1005	1005	1005	1005	1675	1675	3350	4355	4355	5360

representations have been evaluated on each body part and various combinations of body parts. Two data fusion schemes have been validated. The feature-level fusion scheme is slightly better than the decision-level on Buffy, but slightly worse on PASCAL.

The recognition accuracies that we can achieve are 98.04% (with annotated parts) and 96.11% (automated detection and recognition), respectively, on Buffy; while 73.82% and 67.74%, respectively, on PASCAL.

The PLS method can efficiently reduce the feature dimensions from very high numbers (e.g., 5,360) to an extremely small number of 40.

The PASCAL database is more challenging than Buffy for unconstrained gender recognition from the articulated human body.

Chapter 6

Color and Pattern Description

As mentioned in chapter 1, given a photo, we not only want to determine the gender of the person, but also want to describe the appearance of the person. Say for example, we want to tell what kind of clothes he or she wears, what he or she is doing, what is the background in the image. In this chapter, we will focus on the color and the pattern of the clothes.

We will present some preliminary results on describing color and pattern of upper body clothes based on the annotated body parts. The Pascal data set is used for the color and pattern description. Because some images in the Pascal dataset are grayscale images, we will not use those grayscale image for color and pattern description, 522 images are used for the experiments.

6.1 Color Description of Clothes

Color names are useful in real-world applications such as image retrieval and image annotation. Some basic color terms were proposed by Berlin and Kay [31] from the linguistic point of view in 1969. And this work has achieved an important influence in the study of color naming. The color naming process in computer vision is to map RGB values to the predefined color names. This mapping is usually realized by labelling hundreds of color chips within a well-defined experimental setup as in [32, 33, 34]. The shortcoming of the chip-based color naming is that it learns color names from images obtained in a controlled laboratory setting, which is different from the real world. In real world, images are taken under varying illumination, different shadows, unknown camera settings, etc. Different

from the chip-based color naming model, Weijer etc proposed to learn color names from real world images [35]. Based on their PLSA-ind model for color naming, we propose the following clothes description algorithm.

Given a pixel x in an image I , the probability of the color name of the pixel is given as

$$p(c|x) \propto p(c)p(x|c)$$

where c is the color name, $p(c)$ is the color name prior and it is taken to be uniform.

Follow the work color terms proposed by Berlin and Kay [31], we use the eleven basic colors: black, blue, brown, grey, green, orange, pink, purple, red, white, yellow. The model parameter is learned from 250 Google images. Then we apply this PLSA-ind model to our clothes color description.

We manually label the upper body clothes in Pascal set as: black, blue, brown, grey, green, orange, pink, purple, red, white, yellow and OTHER. Each clothes is labelled as only one color. If the person in an image wear a non-uniform color clothes, we use the majority color of the clothes to name the clothes color.

Given an image I , we determine the color for the upper body clothes as in Algorithm 6.1.

Algorithm 6.1 Clothes Description Algorithm

1. Find the rectangle patch P of the torso of the image I .
 2. Determine the color name of each pixel of P by the PLSA-ind model.
 3. Compute the percentage of each color in P .
 4. Find the color i with maximum percentage.
 5. If $percent(i) > \beta$, where β is a threshold, then name the color of upper body clothes as i ;
else, naming as OTHER.
-

Figure 6.1.1 is the confusion matrix which provides the information of the color description by our color naming algorithm. Row represents the ground truth color, and the column represents the predicted color. The number at the right most column represents the number of ground-truth images for each color. Say for example, there are 118 images are labeled as black in the labelled image set. The overall correct rate 68.2%. From the confusion matrix, we observe that 'black', 'blue' and 'orange' are recognized with high

accuracy. But 'white' has a low accuracy, it may be because that white color is easily influenced by different illumination.

Figure 6.1.1: Color confusion matrix for Pascal dataset

6.2 Pattern Description of Clothes

Despite the description of clothes colors, we may tell something more about the clothes such as whether it is uniform color, or a plaid, etc. That is, we want to describe the pattern of the clothes. For the pattern description, we labelled the upper body clothes in the Pascal data set as the following four patterns: {'Vertical strip', 'Horizontal Strip', 'Plaid' and 'Others'}. We first detect the rectangle patch of the torso. Then we use the information in the rectangle patch of the torso to infer the pattern of the upper body clothes.

Given an image I , we use the following rule to determine which pattern it is:

- (1) Get the rectangle patch of the torso.
- (2) Find the vertical edges in the rectangle patch by using Sobel filter (or Canny filter).
- (3) Find horizontal edges in the rectangle patch by using Sobel filter (or Canny filter).
- (4) If (the pixel number of vertical edges $\geq \alpha$) and (the pixel number of horizontal edges $\geq \alpha$), the pattern is described as 'Plaid';

if (the pixel number of vertical edges $> \alpha$) and (the pixel number of horizontal edges $\leq \alpha$), the pattern is described as 'Vertical Strip';

if (the pixel number of vertical edges $\leq \alpha$) and (the pixel number of horizontal edges $> \alpha$), the pattern is described as 'Horizontal Strip';

Otherwise, we say the pattern is a 'Others'.

The decision rule we proposed for the pattern description is pretty simple, however the result is good. The result based on these rules are shown in Fig 6.2.1. In the pattern confusion matrix in Fig 6.2.1, we use 'V', 'H', 'G', 'O' to represent 'Vertical strip', 'Horizontal Strip', 'Plaid' and 'Others' respectively. The overall correct rate of the confusion matrix is 69.54%. And the accuracy for 'Plaid' and 'Vertical Strip' are 92% and 75%.

V	.75	.13		.13	8
H	.05	.48	.05	.43	21
G			.92	.08	13
O	.15	.03	.12	.70	480
	V	H	G	O	

Figure 6.2.1: Pattern confusion matrix for Pascal dataset

Chapter 7

Conclusions and Future Work

In this work, we first studied the new problem of gender recognition in unconstrained images of the articulated human body. A novel gender recognition system has been presented based on upper body detection, body parts localization, and parts normalization. Through comprehensive empirical studies, we have discovered that the lower arms are not informative, the torso, head, and upper arms can be combined together to recognize gender, and data fusion schemes have advantages over single representations to deliver a more robust solution with good performance. The PLS method can efficiently reduce the dimensionality and classify the patterns.

From the results of our experiments, we find that the PASCAL data set is much more challenging than the Buffy data set. We think it is because that PASCAL images are taken by different people under different illumination with different cameras. But the Buffy images are produced by professional TV show producer. In the future, we should spend more time on the study related to the PASCAL data set and develop advanced learning methods to improve the gender recognition accuracy on the PASCAL database.

We also showed some preliminary results of color description and pattern description of the clothes in chapter 6. So far, the result is not good enough, we will do further research in the field. Our ultimate goal is to develop automatic system which will describe the person in an image in details (such as appearance of the person, action, background, etc).

Bibliography

- [1] T. Ahonen, A. Hadid, and M. Pietikainen. Face recognition with local binary patterns. In *Eur. Conf. on Comput. Vision*, pages 469–481, 2004.
- [2] S. Baluja and H. A. Rowley. Boosting sex identification performance. *Intl. J. of Comput. Vision*, 71(1):111–119, 2007.
- [3] V. Bruce, A. Burton, E. Hanna, P. Healey, and O. Mason. Sex discrimination: How do we tell the difference between male and female faces? *Perception*, 22:131–152, 1993.
- [4] R. Brunelli and T. Poggio. Hyperbf networks for gender classification. In *Proc. DARPA Image Understanding Workshop*, pages 311–314, 1992.
- [5] L. Cao, M. Dikmen, Y. Fu, and T. Huang. Gender recognition from body. In *ACM Multimedia*, 2008.
- [6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conf. on Comput. Vision and Pattern Recognit.*, pages 886–893, 2005.
- [7] B. V. Dasarathy. *Decision Fusion*. IEEE Computer Society Press, 1994.
- [8] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari. Articulated human pose estimation and search in (almost) unconstrained still images. In *ETH Zurich, Technical Report No.272*, 2010.
- [9] M. Everingham, L. V. Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge 2008 (voc2008) results. 2008.
- [10] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *Int. J. Comput. Vision*, 61(1):55C79, 2005.

- [11] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multi-scale, deformable part model. In CVPR, 2008.
- [12] B. Golomb, D. Lawrence, and T. Sejnowski. Sexnet: A neural network identifies sex from human faces. In *Advances in Neural Information Processing Systems 3*, pages 572–577, 1991.
- [13] G.-D. Guo, C. Dyer, Y. Fu, and T. S. Huang. Is gender recognition affected by age? In *IEEE International Workshop on Human-Computer Interaction*, pages 2032–2039, October 2009.
- [14] G.-D. Guo and G. Mu. Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 657– 664, 2011.
- [15] G.-D. Guo, G. Mu, and Y. Fu. Gender from body: A biologicallyinspired approach with manifold learning. In *the Ninth Asian Conference on Computer Vision*, pages 236–245, September 2009.
- [16] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.
- [17] B. Moghaddam and M.-H. Yang. Learning gender with support faces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):707–711, 2002.
- [18] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio. Pedestrian detection using wavelet templates. In *IEEE Conf. on Comput. Vision and Pattern Recognit.*, pages 193–199, 1997.
- [19] R. Rosipal and N. Kramer. Overview and recent advances in partial least squares. In C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor, editors, *Subspace, Latent Structure and Feature Selection Techniques*, pages 34–51. Springer, 2006.
- [20] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Trans. on Graphics*, 23(3):309–314, 2004.
- [21] W. Schwartz, H. Guo, and L. Davis. A robust and scalable approach to face identification. In *ECCV*, 2010.

- [22] W. Schwartz, A. Kembhavi, D. Harwood, and L. Davis. Human detection using partial least squares analysis. In ICCV, 2009.
- [23] V. N. Vapnik. Statistical Learning Theory. John Wiley, New York, 1998.
- [24] H. A. Wild, S. E. Barrett, M. J. Spence, A. J. O'Toole, Y. D. Cheng, and J. Brooke. Recognition and sex categorization of adults' and children's faces: examining performance in the absence of sex-stereotyped cues. *J. of Exp. Child Psychology*, 77:269–291, 2000.
- [25] H. Wold. Path models with latent variables: The nipals approach. In H. M. Blalock and et al, editors, *Quantitative Sociology: International perspectives on mathematical and statistical model building*, pages 307– 357. Academic Press, 1975.
- [26] H. Wold. Partial least squares. In S. Kotz and N. Johnson, editors, *Encyclopedia of Statistical Sciences*, volume 6, pages 581–591. Wiley, New York, 1985.
- [27] M. K. Yamaguchi, T. Hirukawa, and S. Kanazawa. Judgment of sex through facial parts. *Perception*, 24:563–575, 1995.
- [28] T. Ojala, M. Pietikäinen, and D. Harwood (1994), "Performance evaluation of texture measures with classification based on Kullback discrimination of distributions", *Proceedings of the 12th IAPR International Conference on Pattern Recognition (ICPR 1994)*, vol. 1, pp. 582 - 585.
- [29] Auteurs Van De Weijer Joost, Schmid Cordelia, Verbeek Jakob url, Larlus Diane Détail. Learning Color Names for Real-World Applications. *IEEE Transactions on Image Processing*, IEEE Signal Processing Society, 2009, 18 (7), pp. 1512-1523.
- [30] Corinna Cortes and V. Vapnik, "Support-Vector Networks", *Machine Learning*, 20, 1995.
- [31] B. Berlin and P. Kay, *Basic color terms: their universality and evolution*. Berkeley: University of California, 1969.
- [32] A. Mojsilovic, "A computational model for color naming and describing color composition of images," *IEEE Transactions on Image Processing*, vol. 14, no. 5, pp. 690–699, 2005.

- [33] R. Benavente, M. Vanrell, and R. Bladrich, “A data set for fuzzy colour naming,” *COLOR research and application*, vol. 31, no. 1, pp. 48–56, 2006.
- [34] G. Menegaz, A. L. Troter, J. Sequeira, and J. M. Boi, “A discrete model for color naming,” *EURASIP Journal on Advances in Signal Processing*, vol. 2007, 2007.
- [35] J. van de Weijer, Cordelia Schmid, Jakob Verbeek, Diane Larlus, Learning Color Names for Real-World Applications. *IEEE Transaction in Image Processing (TIP)*, vol 18 (7):1512-1524, July 2009.
- [36] http://www.robots.ox.ac.uk/~vgg/data/stickmen/buffy_stickmen_v3.0.tgz
- [37] http://www.vision.ee.ethz.ch/~calvin/ethz_pascal_stickmen/downloads/ETHZ_PASCAL_Stickmen_v1.1.tgz