[UCI]I804:24011-200000242050

2011년 8월
석사학위 논문

Facial Pose and Expression Recognition Using Limited Feature Points Based on a Dynamic Bayesian Network

조선대학교 대학원

컴 퓨 터 공 학 과

Wei Zhao

Facial Pose and Expression Recognition Using Limited Feature Points Based on a Dynamic Bayesian Network

제한된 특징점들을 이용한 동적 베이지안 네트워크 기반 얼굴 방향 및 표정 인식

2011년 8월 25일

조선대학교 대학원

컴 퓨 터 공 학 과

Wei Zhao

Facial Pose and Expression Recognition Using Limited Feature Points Based on a Dynamic Bayesian Network

지도교수 이 상 웅

이 논문을 공학 석사학위신청 논문으로 제출함

2011년 4월

조선대학교 대학원

컴 퓨 터 공 학 과

Wei Zhao

Wei Zhao 의 석사학위논문을 인준함

위원장 조선대학교 교 수 김 판 구 (인)

위 원 조선대학교 조교수 양 희 덕 (인)

위 원 조선대학교 조교수 이 상 웅 (인)

2011 년 5 월

조선대학교 대학원

ABSTRACT

Facial Pose and Expression Recognition Using Limited Feature Points Based on a Dynamic Bayesian Network

Wei Zhao
Advisor: Prof. Sang-Woong Lee, Ph.D.
Department of Computer Engineering
Graduate School of Chosun University

In daily life, language is an important tool during the communications between people. Except the language, facial actions can also provide a lot of information. Therefore, facial actions recognition becomes a popular research topic in Human-Computer Interaction (HCI) field.

However, facial actions recognition is a quite challenging task because of its complexity. In a literal sense, there are thousands of facial muscular movements many of which have very subtle differences. Moreover, muscular movements always occur spontaneously when the pose is changed.

To address this problem, we build fully automatic facial points detection system firstly based on local Gabor filter bank and Principal Component Analysis (PCA). Then a Dynamic Bayesian networks (DBNs) is proposed to perform facial actions recognition using junction tree algorithm over a limited number of feature points.

In order to evaluate the proposed method, we have used the Korean face database for model training, and CUbiC FacePix, FEED, and our own database for testing. Experiment results clearly demonstrate the feasibility of the proposed approach.

Contents

List of Tables

List of Figures

Section I

Introduction

Facial actions can provide information not only about affective state, but also about cognitive activity, psychopathology and so on. In our lives, facial actions always express a mount of information beyond words. In Human-Computer Interaction (HCI), facial actions recognition is an important tool for communication between a human being and a computer. However, this is always a tough task because of the essence of facial actions. Thousands of distinct nonrigid facial muscular movements have been observed and most of them only differ in a few features. For example, spontaneous facial actions are usually in the term of slight appearance changes. What is more, different facial actions can happen simultaneously. All of these make the recognition difficult. Many methods have been proposed by researchers in order to solve this problem. The Facial Action Coding System (FACS) [1] is one of the most popular methods to analyze the facial actions. In FACS system, nonrigid facial nuscular movement is described by a set of facial action units (AUs).

In this paper, we focus on the methods based on Bayesian Network (BN) and then extend to Dynamic Bayesian Network (DBN). In [2], a unified probabilistic framework is built to recognize the spontaneous facial actions based on a DBN. The authors declare that there are coherent interactions among rigid and nonrigid facial motions. According to this idea, facial feature points can be organized into two categories: global feature points and local feature points. By separating the 28 feature points into two groups, they realize the interactions between pose and expression variables. In this paper, the pose is considered in only pan angle and divided into three states: left, frontal and right. The expression is analyzed by FACS. Another paper [3] uses BN for face identification. The author takes the salient facial areas as facial features. These facial features are combined together to generate a human face. In [4], a probabilistic measure of similarity is used instead of standard Euclidean nearest-neighbor eigenface matching. The advantage of this improved method is demonstrated by the experiments. Some other researchers use hierarchical Bayesian network for human interactions [5]. In [5], a method to recognize the interactions between two persons is presented using a hierarchical Bayesian network. In our case, we use BN and DBNs to handle the pose and expression recognition using only 21 feature points on human face. We assume that pose and expression can only be considered as a kind of distribution of the feature points. The experiment results show that this hypothesis is reasonable.

Section II

Previous Works

Facial pose and expression are two kinds of different facial actions. Over the decades, there has been extensive researches in computer vision on recognizing facial actions.

For head pose estimation, many methods have been proposed during these 20 years. There are many approaches to cluster all of these methods. We apply the fundamental approach that underlies its implementation [6]. This evolutionary taxonomy consists of the following eight categories that describe the conceptual approaches that have been used to estimate head pose:

- **Appearance template methods** compare a new image of a head to a set of exemplars (each labeled with a discrete pose) in order to find the most similar view.

- **Detector array methods** train a series of head detectors each attuned to a specific pose and assign a discrete pose to the detector with the greatest support.

- **Nonlinear regression methods** use nonlinear regression tools to develop a functional mapping from the image or feature data to a head pose measurement.

- **Manifold embedding methods** seek low-dimensional manifolds that model the continuous variation in head pose. New images can be embedded into these manifolds and then used for embedded template matching or regression.

- **Flexible models** fit a nonrigid model to the facial structure of each individual in the image plane. Head pose is estimated from feature-level comparisons or from the instantiation of the model parameters.

- **Geometric methods** use the location of features such as the eyes, mouth, and nose tip to determine pose from their relative configuration

- **Tracking methods** recover the global pose change of the head from the observed movement between video frames.

- **Hybrid methods** combine one or more of these aforementioned methods to overcome the limitations inherent in any single approach..

Facial expressions convey non-verbal cues, which play a key role in both interpersonal communication and human-machine interfaces. Many researchers have endeavored to find the normative qualities of facial expression [7], [8], [9]. Most of them have been based on FACS. Recently, research has been focused on improving facial action recognition combining spontaneous facial expressions with varying head pose. The work on facial action analysis can be classified as mentioned in [2]. This classifying is based on how these methods deal with the relationships between the head movement and nonrigid facial movements.

- **The first group** of methods [10], [11] explicitly represents and recognizes the facial expression on facial expression databases.

- **The second group** [12], [13], [14], [15] assumes that the 3D head pose is independent of nonrigid facial muscular movements and estimates the pose and nonrigid facial motions sequentially and separately.

- **The third group** [16], [17] explicitly models the coupling between rigid head movement and nonrigid facial motions for facial action analysis and simultaneously recovers the rigid and nonrigid facial motions.

Section III

Facial Feature Point Detection

Face detection and facial feature points extraction are two important steps in a whole facial actions recognition system. The results of the previous part of the system will affect the recognition rate directly. For feature points extraction, there are many popular methods such as Gabor filter-based method, Active Shape Model (ASM), Active Appearance Models (AAM) and so on. ASM performs well in experiments [18]. However, ASM is a statistical approach for shape modeling and feature extraction. In order to train the ASM model, a lot of training data is necessary. AAM [19] has the same problem as ASM. In order to realize the fully automatic feature point detection, several other methods appear. One of the most popular methods is to detect feature points using Gabor filter [20]. Gabor filter is a powerful tool in computer vision field. It is represented by frequencies and orientations. The representations are similar to those of human visual system. However, the drawback is that a lot of time is consumed when Gabor filter is used to detect features. In our paper, we apply a series of steps to simplify the Gabor filter and still obtain the good detection performance. The feature dimension in our paper is only one fifth of the dimension used in paper [20] while the system still performs well.

A Pupils and Nostrils Detection

In order to detect pupils and nostrils, first we should find the human face from the image.

There are many methods for detecting human face from an image. We can decide the human face region using skin color and shape information. Using skin color to find human face is one of the most popular algorithms. There are several advantages of this method. Color processing is much faster than other facial features and color information is independent of the orientation of the face. The disadvantage of skin color-based method is that when obstacle with similar color appears in the image, for example hands, mistakes will happen. The shape information can be added to solve this problem.

If more than one face appear in the image, other features are needed to work it out. Here we only handle the case with one human face. Our human face detecting system performs well in video. After detecting the human face region, we apply the method introduced in [21] to detect pupils and nostrils.

B Facial Feature Points Extraction Based on Local Gabor Filter Bank and PCA

We take the pupils and nostrils as the reference points and separate the face into several areas. In each small area, we build up the feature vectors for each point.

1 Gabor Filter Bank

Gabor filters with different frequencies and orientations can serve as excellent band-pass filters and are similar to human visual system. The Gabor filter function is then given by equation (III.1):

$$g(x, y, f, \theta) = \frac{1}{ab} exp[-\pi(\frac{x_r^2}{a^2} + \frac{y_r^2}{b^2})][exp(i2\pi f x_r)] \tag{III.1}$$

where

$$\begin{aligned} x_r &= x\cos\theta + y\sin\theta \\ y_r &= -x\sin\theta + y\cos\theta \end{aligned}$$

In paper [22], local Gabor filter bank is proposed and compared with traditional global Gabor filter bank. Both the theoretical analysis and the experiment results show that the local Gabor filter bank is effective. In our experiments, we choose 4 orientations and 3 frequencies from the original 8 orientations and 6 frequencies as in [22]. The 12 Gabor filters are combined to form a local Gabor filter bank as shown in Figure III.1.

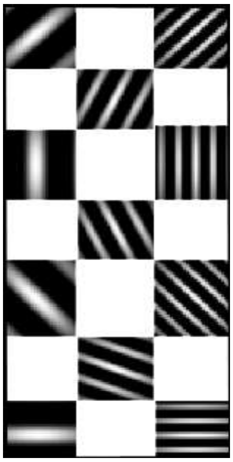

Figure III.1: Local Gabor filter bank with 4 orientations and 3 frequencies

The feature vector is a 11×11 image patch extracted from face image which centered on that point. The image patch is extracted from 12 filtered images and one original gray scale image. Thus 1573 ($11 \times 11 \times 13 = 1573$) dimensional vector is used to present one point. In order to describe the method better, we give an example of the feature vector. As shown in Figure III.2, there are 12 images from filtered human face images each of which is a 11×11 image patch. We can reshape these patches into a vector. This vector represents the point 6 as marked in Figure V.1. Figure III.3 is the feature vector of point 6 extracted from another person. Figure III.4 is the feature vector centered at another point near point 6. Calculating the Euclidean distances between these images, the distance between Figure III.2 and Figure III.3 is smaller than distance between Figure III.3 and Figure III.4. The Gabor filter used in this example is the global Gabor filter. We can also figure out the necessity of the local Gaobr filter because some patches are similar with each other in these three images. The patch in the second row and the third column provide more information than the others. So the Gabor filter for this patch is the most effective for detecting this feature point. This method is similar with the feature extraction method introduced in [20]. However, because we use the local Gabor filter here, the dimension of the feature vector (1573) is much lower than the dimension in [20] which is 8281.

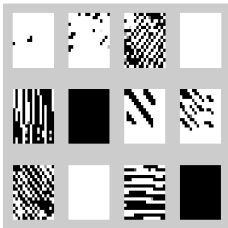

Figure III.2: 12 patches obtained from 12 filtered images. These patches represent the point 6 marked in Figure V.1

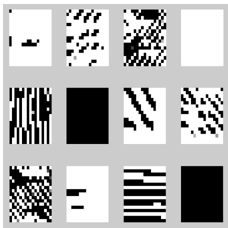

Figure III.3: 12 patches obtained from 12 filtered images. These patches represent the point 6 of a different person.

2 Principal Component Analysis (PCA)

In order to reduce the dimension further, Principal Component Analysis (PCA) is applied. Considering a set of N images $(x_1, x_2, ..., x_N)$, each image is represented

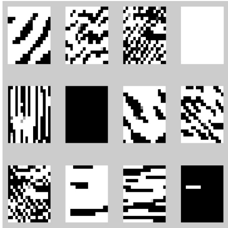

Figure III.4: 12 patches obtained from 12 filtered images. These patches represent the point near point 6.

by t-dimensional Gabor feature vector. The PCA [22], [23] can be used to transform the t-dimensional vector into a f-dimensional vector, where normally $f \ll t$. The new feature vector $y_i \in \Re^f$ are defined by equation (III.2)

$$y_i = W_{pca}^T x_i \quad (i = 1, 2, ..., N) \tag{III.2}$$

where W_{pca}^T is the linear transformations matrix, i is the number of sample images.

In our case, each feature point is represented by 1573-dimensional feature vector. Then a transformations matrix can be found to project the 1573-dimensional vectors to a lower dimensional space.

We take point 6 as an example again to explain our method. Assuming there are five points as the candidates and each candidate is presented by a vector whos dimension is lower than 1573. After calculating the Euclidean distances among these five points and the trained points, we can decide which point is the feature point.

Section IV

Modeling the Pose and Expression with Dynamic Bayesian Network

We simplify the poses and expressions as a position distributions of the feature points. Each pose and expression has a certain position and distance distribution. For example, in Figure IV.5, the feature points of frontal human face is symmetrical. When the head turns to left, the relative position of points are changed to asymmetric. The degree of the asymmetry indicates the poses angle. When smiling, the relative positions of four points of the mouth will change: the corner points of the mouth will move up. We are going to verify that the method is reasonable through the experiments.

A Proposed Bayesian Network

1 A Brief Introduction of Bayesian Network

A Bayesian network was firstly proposed by Pearl[29]. The BN represents the joint probability distribution over a set of random variables in a directed acyclic graph (DAG). The links between these variables represent the causality relationships.

More formally, we can define $Pa(X_i)$ as the parents of variable X_i and the joint probability $P(x)$ over the variables is given by the equation (IV.1):

$$P(x) = \prod_{i=1}^{n} P(X_i|Pa(X_i)) \tag{IV.1}$$

where i is the index of variables and n is the total number of variables. From equation (IV.1), we can tell that a BN consists of two crucial respects: the structure and the parameters. So how to learn the structure and the parameter from a actual problem becomes an important issue in BN. We only consider the parameter learning issue in our case because we have already decided the BN structure. Parameter learning can be classified into 4 types, depending on the goal is to compute full posterior or just a point, and all the variables are observed or some of them are hidden. In our case, the goal is a point estimation and the hidden nodes exist. Hence, Expectation-Maximization (EM) algorithm is applied. We can indicate the two steps of this algorithm from its name: expectation and maximization. In fact, EM is an iterative optimization method to estimate some unknown parameters given measurement data. This algorithm is suitable for the situation that some data are missed. If θ denotes the parameters of the model, v is the observed variables and h is the hidden variables. The log-likelihood is given by the equation (IV.2) :

$$L(\theta, v) \quad = \quad log \sum_{h} p(v, h|\theta) \tag{IV.2}$$

Since maximizing equation (IV.2) directly is difficult, the variational approximation to the EM algorithm is used [25]:

$$\begin{aligned} L(\theta, v) &= log \sum_h p(v, h|\theta) \\ &= log \sum_h p(v, h|\theta) \\ &= log \sum_h \frac{q(h)p(v, h|\theta)}{q(h)} \\ &\geq \sum_h q(h) log \frac{q(v, h|\theta)}{q(h)} \\ &= \sum_h q(h) logp(v, h|\theta) - \sum_h q(h) logq(h) \end{aligned} \tag{IV.3}$$

It can be shown that the optimal setting for variational distribution q(h) is $p(h|v, \theta)$. We do not give prove this conclusion here and the detail can be found in [30]. Hence, we can replace q(h) with $p(h|v, \theta)$. The second term in the last line of the equation can be neglected because it dose not contain any θ. So the equation (IV.3) can be rewritten as following:

$$L(\theta, v) = \sum_h p(h|v, \theta) logp(v, h|\theta) \tag{IV.4}$$

To perform EM algorithm, firstly, infer the distribution of hidden variables h by initial parameter θ_0 and observed data (E-step). Then find the parameters that make the likelihood of equation (IV.2) maximal (M-step). These two steps can be expressed by equation (IV.5):

$$\begin{aligned} &E - step: \quad compute \qquad p(h|v, \theta_t) \\ &M - step: \quad \theta_{(t+1)} \leftarrow argmax_\theta \sum_h p(h|v, \theta_t) logp(v, h|\theta) \end{aligned} \tag{IV.5}$$

After parameter learning, the BN has been fixed. The next work is to infer the result from evidence. Inference is another important task in BN. In this paper, we use the classical Junction Tree algorithm as our inference engine. Junction Tree algorithm is a very popular algorithm and can perform exact inference in both directed and undirected graphical models. This Junction Tree algorithm first presents the original graph into another type, called the junction tree. Then use this junction tree to pass the information and infer the result.

2 The Introduction of the Proposed Bayesian Network

Figure IV.1 is the flow chart of our work.

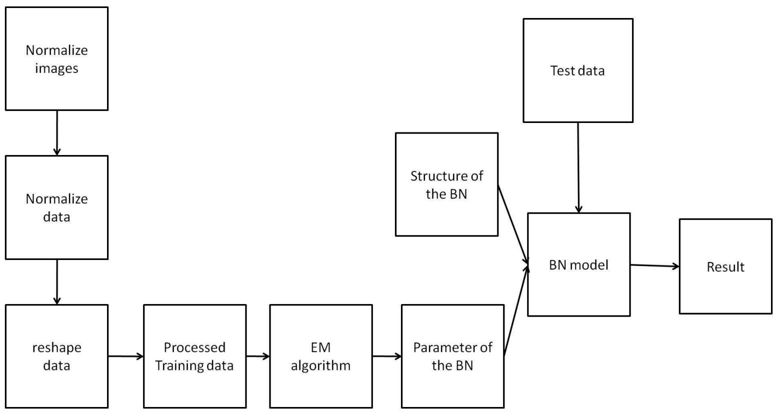

Figure IV.1: The flow chart of the proposed method.

Firstly, we build a two-layer Bayesian network as shown in Figure IV.2. The first layer contains two discrete variables: pose and smile.

Generally, three kinds of angles are used to represent the pose: pan,tilt and roll. In this paper, we only consider the pan angle. For human being's head, pan angle

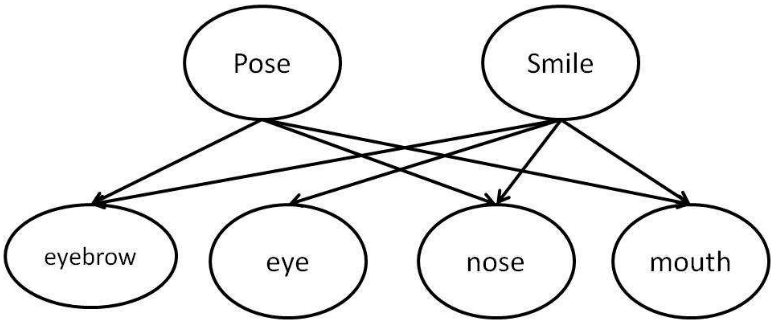

Figure IV.2: A two-layer Bayesian Network of the proposed BN.

means turn left or right. We separate the pan angle into 5 different views: frontal, left, more-left, right and more-right which correspond to five discrete states of node pose (Pan angle $\in \{1, 2, 3, 4, 5\}$) in the system. As described, we separate left- turn and right-turn angle into two groups. Usually, if the angle is around or larger than $45°$, it will be clustered to the more-left or more-right group. If the angle is between $15°$ and $45°$, it will be clustered to left or right group. Of course, the interval here is general and the boundary between groups are not exact.

The second layer consists of four continuous variables: eyebrow, eye, nose and mouth which are denoted by B, E, N, M. Each variable is presented by a vector with different length. The joint probability of the BN in Figure IV.2 is factored into conditional probabilities and prior probabilities as in equation (IV.6). For better expression, we denote P for node pose, S for smile.

$$
\begin{aligned}
P(B,E,N,M,P,S) &= P(B,E,N,M|P,S) \times P(P,S) \qquad \text{(IV.6)} \\
&= P(B|P,S)P(E|P,S)P(N|P,S)P(M|P,S)P(P)P(S)
\end{aligned}
$$

Our aim is to estimate the belief of pose and smile nodes given the evidences of the second layer:

$$
P(P,S|B,E,N,M) = \frac{P(B,E,N,M,P,S)}{\sum_{pose}\sum_{smile} P(P,S,B,E,N,M)} \qquad \text{(IV.7)}
$$

where the summation is over all possible configurations of the values on the node Pose and Smile.

Then the third layer is built. In this layer, the observations of the second-layer nodes are defined. According to this definition, the nodes in this layer are continuous too. The second and third layers reflet the uncertain relationship between the observations and the real values. The nodes on the third layer can be denoted by O_B, O_E, O_N, O_M as shown in Figure IV.3. The joint probability of the nodes from these two layers can be calculated by equation (IV.8):

$$
\begin{aligned}
P & \quad (B,E,N,M,O_B,O_E,O_N,O_M) \\
= & \quad P(B,O_B)P(O_B)P(E,O_E)P(O_E)P(N,O_N)P(O_N)P(M,O_M)P(O_M)
\end{aligned}
\qquad \text{(IV.8)}
$$

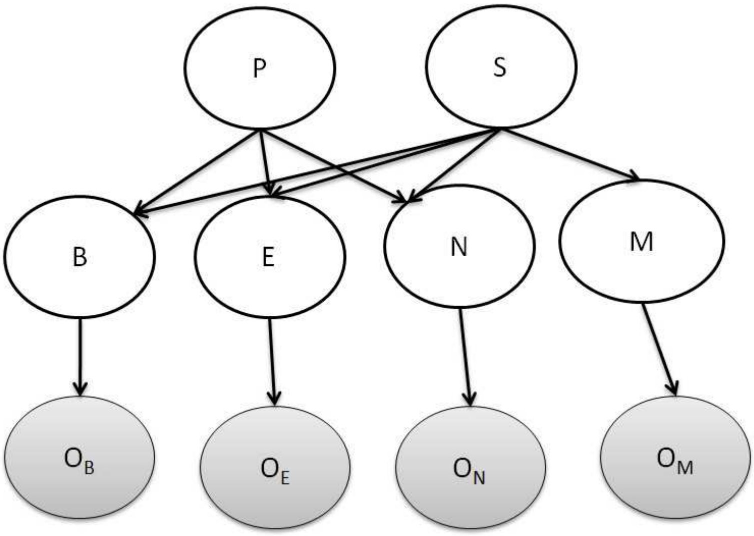

Figure IV.3: A simplified Bayesian network of proposed model.

In equation (IV.6), (IV.7), (IV.8), "P" is the probability distribution of the nodes. For the discrete nodes without parents, the probability distributions are given in the term of Conditional Probability Table (CPT). For the continuous nodes with discrete or continuous parents, the probability distributions are given in the term of Conditional Probability Distribution (CPD). In this work, the CPD of continuous nodes are parameterized as a Gaussian distribution. For example, node S_B, given its parents pose and smile, the distribution can be defined as:

$$\begin{aligned} P & \quad (B = b|pose = k, smile = l) \\ = & \quad \frac{1}{(2\pi)^{\frac{n}{2}} |\sum_i|^{\frac{1}{2}}} exp(-\frac{1}{2}(b - \mu_i)^T \sum_i^{-1}(b - \mu_i)) \end{aligned} \tag{IV.9}$$

where n is the dimension of the feature vector. Mean μ_i and the covariance matrix $\sum_i$ are the parameters of the conditional gaussian distribution.

3 Parameter Learning of the Proposed Bayesian Network

We choose 50 people from Korea Face database for training. Each person contains 5 poses and 1 expression as shown in Figure IV.4.

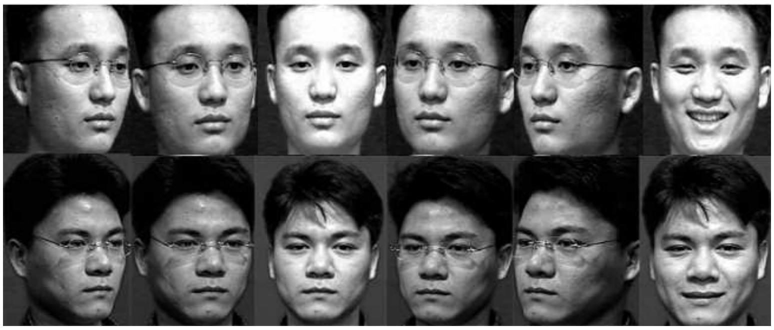

Figure IV.4: An example of the training data used in our model.

After getting the observations of the variables, we begin parameter learning step using the EM algorithm introduced before. We only choose 21 points on human face which are shown in Figure IV.5.

We can verify the parameter learning result by sampling some data from the BN and compare them with the training data. From Figure IV.6 and Figure IV.7, we can compare the data sampled from model before and after learning. After learning, the similarity of the two figures dramatically increase. This similarity implies the success of BN parameter learning.

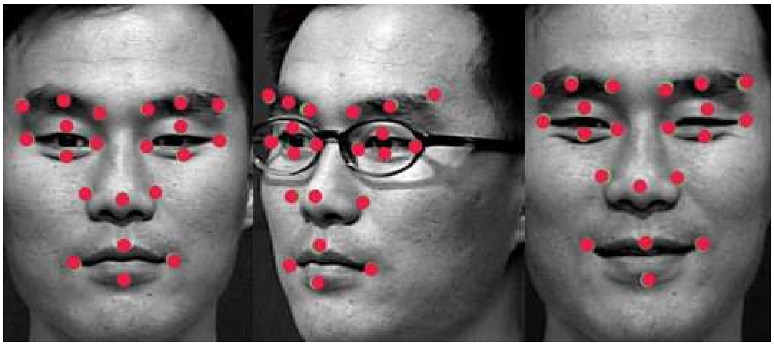

Figure IV.5: 21 feature points we applied in experiments.

Given this Bayesian network, based on the measurements of feature points, pose and expression are recognized by finding the optimal states of them:

$$Pose^*, Ex^* = argmax_{Pose,Ex} p(Pose, Ex|O_B, O_E, O_N, O_M). \tag{IV.10}$$

where Ex means expression.

B Extend the Proposed Bayesian Network to Dynamic Bayesian Network

1 A Brief Introduction of Dynamic Bayesian Network

Sequential data comes in two main forms: temporal (time-series) data, which is generated sequentially by some causal process, and sequence data. For modeling time-series data, it is natural to use directed graphical models, which can capture the fact that time flows forward. Arcs within a time-slice can be directed or undirected,

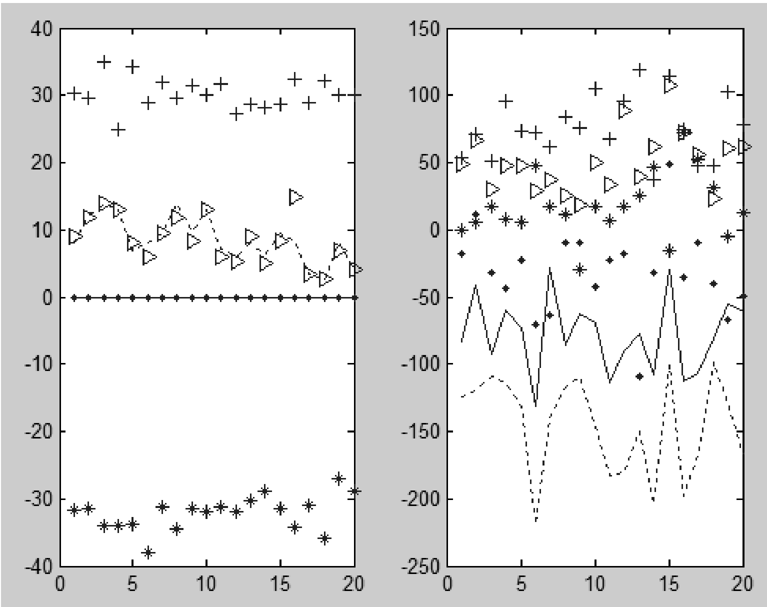

Figure IV.6: The distribution of node nose before parameter learning. The left image plots the training data of nose. Because the length of nose vector is 6, there are 6 lines in the figure. The right image is the data sampled from the BN before the parameters are learned.

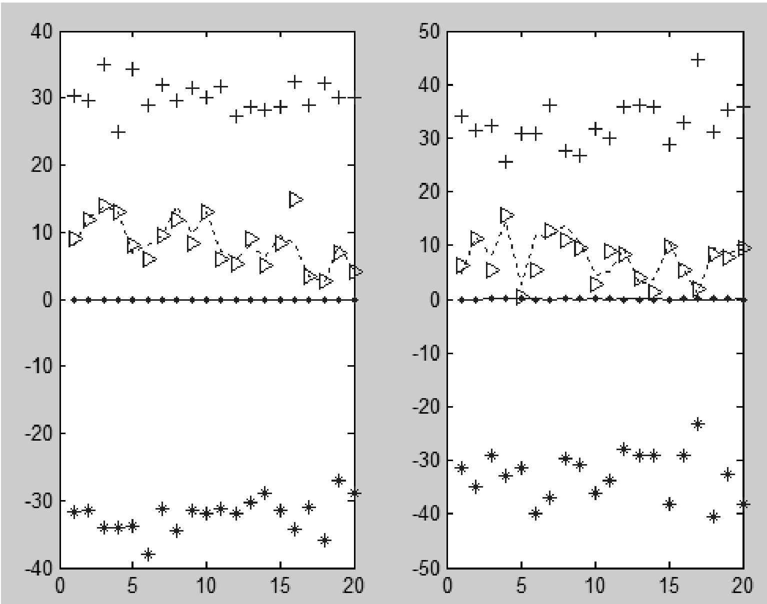

Figure IV.7: The distribution of node nose after parameter learning. The left image plots the training data of nose. The right image is the data sampled from BN after learning the parameters.

since they model instantaneous correlation. If all arcs are directed, both within and between slices, the model is called a DBN [37]. A DBN is defined as $B = (G, \Theta)$ where G is the model structure, and Θ represents the model parameters like the CPDs/CPTs for all nodes. There are two assumptions in the DBN model: First, the system is first-order Markovian, and second, the process is stationary which means that the transition probability $P(X^{t+1}|X^t)$ is the same for all t. Therefore, a DBN B can be also defined by two subnetwork: the static network B_0 and $B_{\rightarrow}$as shown in Figure IV.8. The static distribution $B_0 = (G_0, \Theta_0)$ captures the static distributions over all variables X^0. The transition network $B_{\rightarrow} = (G_{\rightarrow}, \Theta_{\rightarrow})$ specifies the transition probability for all t in finite time slices T.

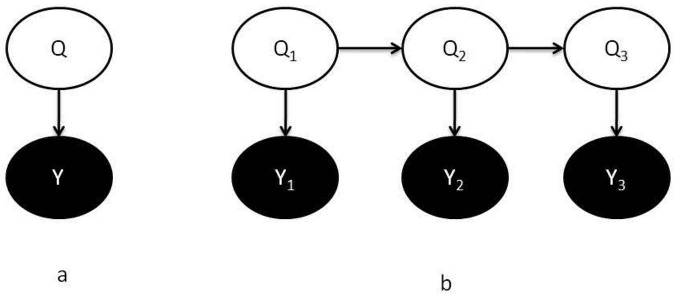

Figure IV.8: An example of DBN. The left image is a static network and the right image is the transition network

Given a DBN model, the joint probability over all variables can be factorized by unrolling the DBN into an extended static BN, whose joint probability is computed as equation (IV.11):

$$P(x^0, ..., x^T) = P_{B_0}(x^0) \prod_{t=0}^{T-1} P_{B_\rightarrow}(x^{t+1}|x^t) \tag{IV.11}$$

and transition network $B_\rightarrow$ can be decomposed as follows based on the conditional independencies encoded in the DBN:

$$P_{B_\rightarrow}(x^{t+1}|x^t) = P_{B_0}(x^0) \prod_{i=1}^{N} P_{B_\rightarrow}(x_i^{t+1}|pa(x_i^{t+1})) \tag{IV.12}$$

2 Parameter Learning of the Proposed DBN Network

According to the definition of DBNs, we unroll the original BN in finite time slices as shown in Figure IV.9 and Figure IV.10.

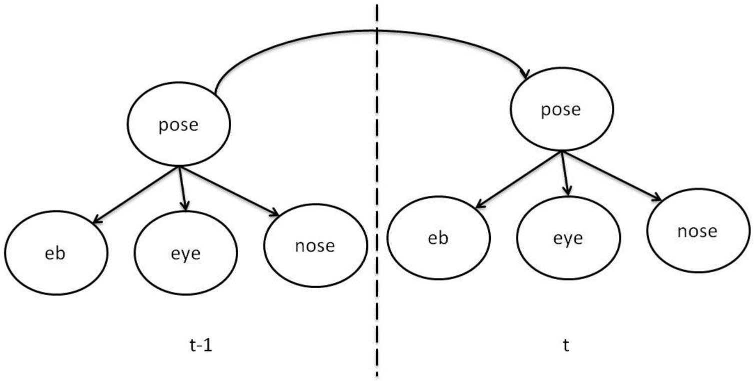

Figure IV.9: DBN of pose recognition.

Four cases have been distinguished in [38] for describing learning problems in DBNs. We summarize these four cases in Table IV.1.

Full observability means that the values of all the variables are known. Partial observability means that the value of some variables are unknown and these unknown

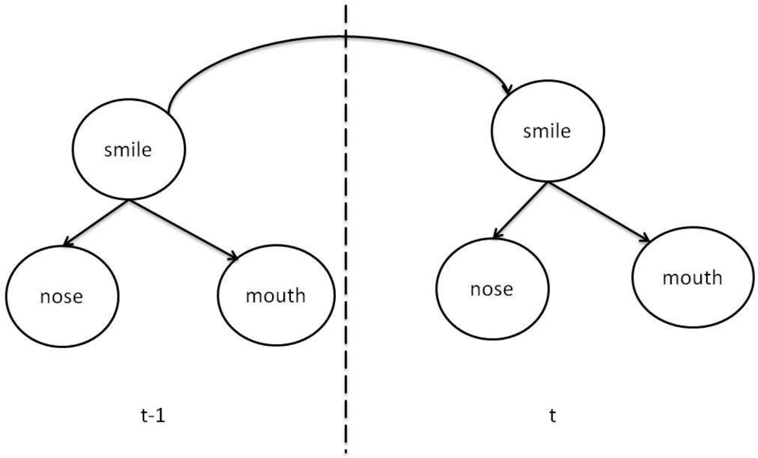

Figure IV.10: DBN of smile recognition.

Table IV.1: Four Cases for Creating DBNs Structure and Determining Their Parameters

Structure/Observability	Method
Known/Full(complete data)	Simple statistics
Known/Partial(incomplete data)	EM or gradient ascent
Unknown/Full(complete data)	Search through model space
Unknown/Partial(incomplete data)	Structural EM

variables are called hidden variables. In our paper, because the hidden variables exist and the structure is known, EM algorithm is applied. We have introduced EM algorithm for BN briefly in previous section. Here we explain the EM algorithm applied in DBNs through equation (IV.13) and equation (IV.14).

$$\log Pr(Q_0^{T-1}, Y_0^{T-1}|\theta) = \sum_{t=1}^{T-1} \log Pr(q_t|q_{t-1}) + \sum_{t=0}^{T-1} \log Pr(y_t|q_t) + \log Pr(q_0), \tag{IV.13}$$

where θ denotes the model parameter vector, Q is the hidden variables and the Y is the observations. The maximization step aims at finding parameters θ that satisfy the next condition:

$$\begin{aligned} \frac{\partial B(P, \hat{Q})}{\partial \theta} &= \sum_{t=1}^{T-1} \langle \frac{\partial \log Pr(x_t, x_{t-1})}{\partial \theta} \rangle + \sum_{t=0}^{T-1} \langle \frac{\partial \log Pr(y_t, x_t)}{\partial \theta} \rangle + \langle \frac{\partial \log Pr(x_0)}{\partial \theta} \rangle \\ &= 0, \end{aligned} \tag{IV.14}$$

We use the EM algorithm to train the DBNs and obtain the parameters which maximize the likelihood of the training data. Figure IV.11 gives some samples from the original DBNs before learning. We fetch 3 points from eyebrow and in Figure IV.11 we use three colors to present these 3 points. Figure IV.12 shows the distributions of the eyebrow points after parameters learning. We can figure out that feature points' distributions are characteristic for different head poses. The upper figure in Figure IV.12 is the distributions when the head turns right and the lower figure is the distributions when the head turns left. The upper figure in Figure IV.13 is the nose feature points' distributions when the head turns right and the lower figure is when the head turns left.

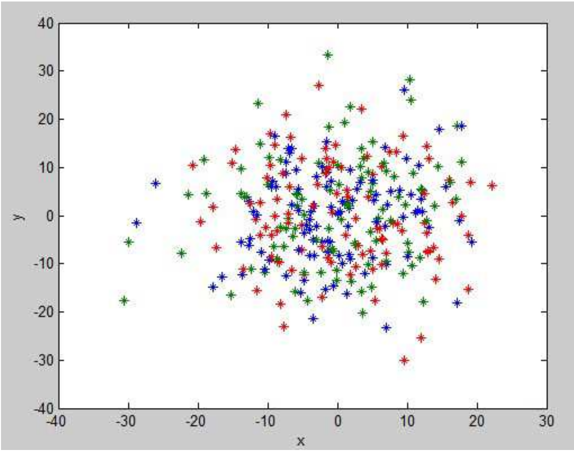

Figure IV.11: The eyebrow feature points' distributions sampled from the DBNs before parameter learning

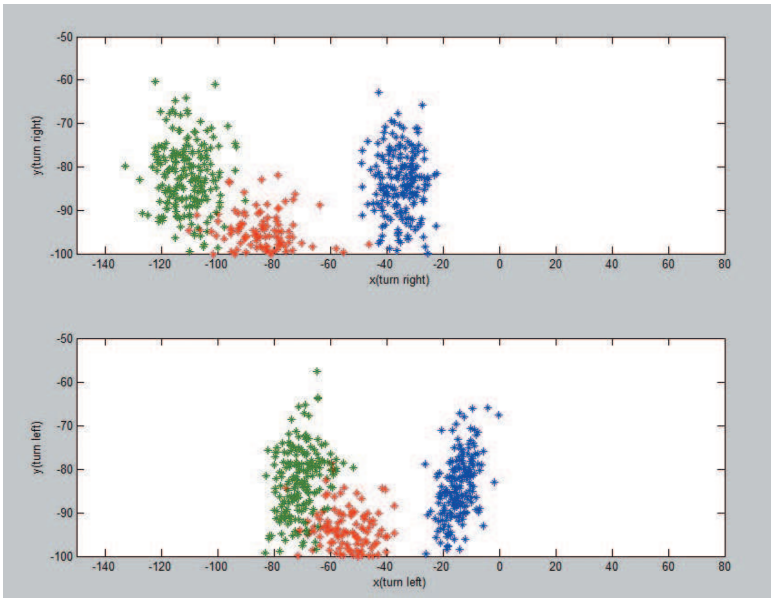

Figure IV.12: The eyebrow feature points' distributions sampled from the DBNs after parameter learning.

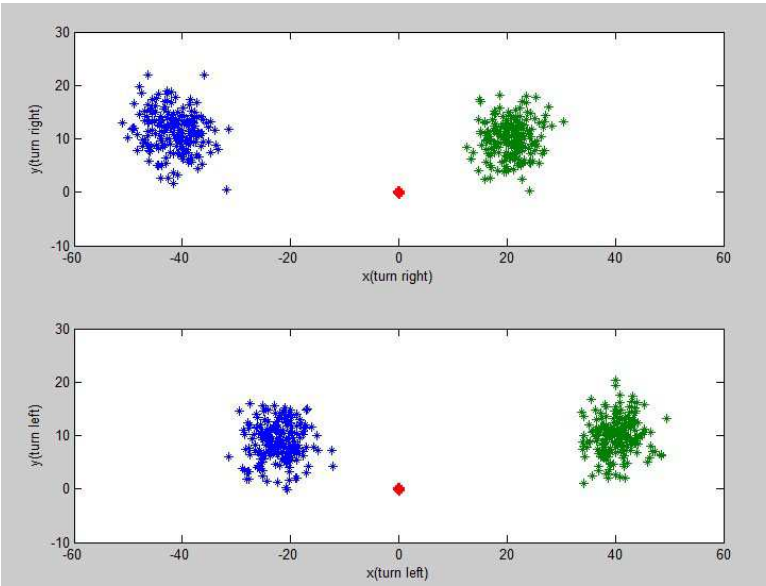

Figure IV.13: The nose feature points' distributions sampled from the DBNs after parameter learning.

Section V

Experiments Results and Analysis

A Experiment Environments and Databases Used in Experiments

The experiments are performed on the computer with $2.00GB$ RAM. The databases used in experiments are including Korean face database, CUbiC FacePix(30) database [27], [28], FEED Database (Facial Expressions and Emotion Database) [35], Japanese Female Facial Expression (JAFFE) database and a small database we create ourselves. The information of these databases can be found in the following list:

- **Korean Face Database** In Korea Face database, there are several different images for one person which are taken under different illuminations, gestures, obstacles or expressions. For the expression in Korean database, only smile is included.

- **CUbiC FacePix(30) Database** CUbiC FacePix(30) is a face image database [27], [28] created at the Center for Cognitive Ubiquitous Computing (CUbiC) at Arizona State University, and made available free of charge to the worldwide research community. It contains face images of 30 people. There are 3 sets of face images for each of these 30 people (each set consisting of a spectrum of 181 images) where each image corresponds to a rotational interval of 1 degree, across a spectrum of 180 degrees.

- **FEED Database** This Database with facial expressions and emotions from the Technical University Munich is an image database containing face images showing a number of subjects performing the six different emotions. The database has been developed in order to assist researchers who investigate the effects of different facial expressions.

- **JAFFE Database** The database contains 213 images of 7 facial expressions (6 basic facial expressions +1 neutral) posed by 10 Japanese female models. Each image has been rated on 6 emotion adjectives by 60 Japanese subjects. The database was planned and assembled by Miyuki Kamachi, Michael Lyons, and Jiro Gyoba and the photos were taken at the Psychology Department in Kyushu University.

- **Our own Database** In order to evaluate the proposed BN in spontaneous situation, we create a database ourselves. The database is comprised of videos which show spontaneous head gesture and expressions. In experiments, we extract images from these videos.

B Experiment Results for Feature Detection

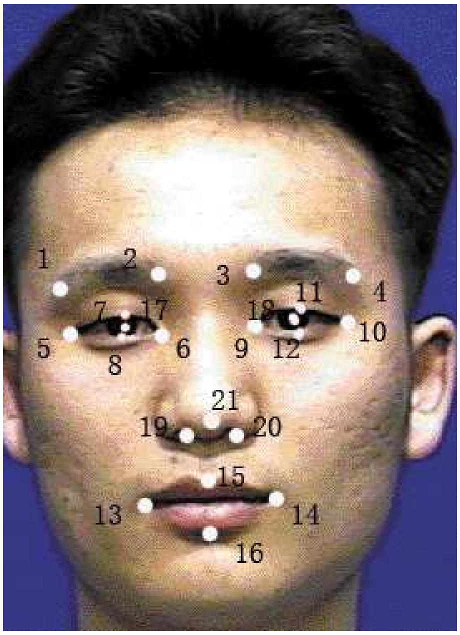

Figure V.1: An example of successfully detected points.

The facial feature points detection method proposed here is trained and tested on the Korea Face database. For our study, we use 60 frontal samples with nature expression. The 60 images are divided into two groups. 20 images are used for training and the 40 images are used for testing. To evaluate the performance of our system, the located facial points was compared to the true point which is got manually. If the distance between automatically detected point and the true point is less than 2 pixels, the detection is defined as a success. Table V.1 gives the results of 16 facial feature points' detection result based on Korea Face database. Table V.2 gives the results of Pupils and nostrils detection results using the method introduced in [21].

Figure V.2: Some usual mistakes appear during the experiments. The points 8, 15, 19 are missed in the left image and the points 14, 15, 16 are missed in the right image.

Table V.1: Experiment Results of Facial Feature Point Detection Based on Korea Face Database

16 **feature points detection results**	
Detected point	Accurate Rate
1: left corner of left eyebrow	95%
2: right corner of left eyebrow	93%
3: left corner of right eyebrow	90%
4: right corner of right eyebrow	90%
5: left corner of the left eye	98%
6: right corner of the left eye	98%
7: top of the left eye	90%
8: bottom of the left eye	86%
9: left corner of the right eye	93%
10: right corner of the right eye	95%
11: top of the right eye	93%
12: bottom of the right eye	90%
13: left corner of mouth	93%
14: right corner of mouth	90%
15: top of the mouth	86%
16: bottom of the mouth	80%
Average accurate rate of first 16 points	91.25%

Table V.2: Pupils and Nostril Detection Results Using the Method Introduced in [21]

Pupils and Nostril Detection	
Detected point	Accurate Rate
17: left pupil	98%
18: right pupil	98%
19: left nostril	92%
20: right nostril	90%
Average accurate rate of first 16 points	94.5%
Average accurate rate of all points	92.87%

The accurate detection of pupils and nostrils is pretty crucial in the experiments. The regions of interest(ROI) are defined based on the locations of pupils and nostrils. As shown in Table V.2, the accurate rate is good. Sometimes, even though some mistakes appear in detecting the pupils and nostrils, the distance between the true point and the detected point is not too far to affect the location of ROI. What is more, the corner points are more easier to be detected while the bottom points like point 8 and point 16 are missed more frequently. This is because the characters of corner points are obvious and stable. The feature points described in Table V.1 and Table V.2 are shown in Figure V.1. Figure V.2 shows some general mistakes in experiments.

C Experiment Results of Head Pose Recognition Based on CUbiC FacePix(30) Database and Our Own Database

1 Normalization Based on Korean Face Database

Before training and testing, we normalize the figures as introduced here. We use the two points as shown in Figure V.3. If we denote the left and right points a_1 and a_2 respectively, the rotation angle is calculated as the equation (V.1):

$$\theta = \frac{y_2 - y_1}{x_2 - x_1} \tag{V.1}$$

where x_1 and y_1 is the coordinate value of a_1 and x_2 and y_2 is the coordinate value of a_2. After cropping and rotation, a normalized image from Korean Face database is given in Figure V.3. Secondly, extract the feature points information from ground

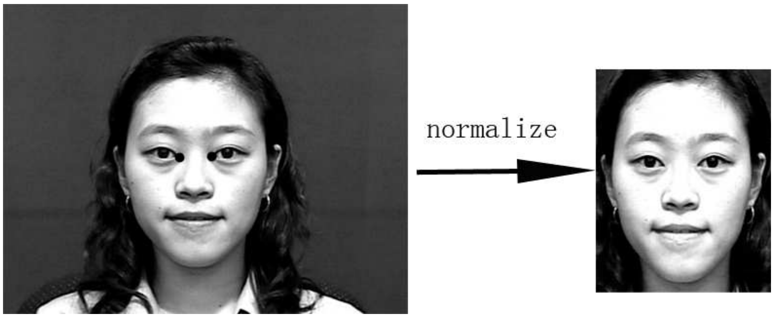

Figure V.3: A normalized image from Korean Face database.

truth data files which are contained in the database. This information is the coordinate values of original images' feature points. Thirdly, translate the original coordinate values to the new coordinate. The translate formulas are different depending on the value of θ. If θ is larger than zero, the new coordinate values can be calculated as in equation (V.2) and equation (V.3):

$$
\begin{aligned}
x_{new} &= \frac{x}{\cos\left(\frac{\theta\pi}{180}\right)} + \left(y - x \cdot \tan\left(\frac{\theta\pi}{180}\right)\right) \cdot \sin\left(\frac{\theta\pi}{180}\right) && \text{(V.2)} \\
y_{new} &= \frac{y}{\cos\left(\frac{\theta\pi}{180}\right)} + \left(640 - \left(x + y \cdot \tan\left(\frac{\theta\pi}{180}\right)\right)\right) \cdot \sin\left(\frac{\theta\pi}{180}\right) && \text{(V.3)}
\end{aligned}
$$

If the θ is less than zero, the newo coordinate values should be calculated as in equation (V.4) and equation (V.5):

$$
\begin{aligned}
x_{new} &= \frac{x}{\cos\left(\left|\frac{\theta\pi}{180}\right|\right)} + \left(480 - \left(y + x \cdot \tan\left(\left|\frac{\theta\pi}{180}\right|\right)\right)\right) \cdot \sin\left(\left|\frac{\theta\pi}{180}\right|\right) && \text{(V.4)} \\
y_{new} &= \frac{y}{\cos\left(\left|\frac{\theta\pi}{180}\right|\right)} + \left(x - y \cdot \tan\left(\left|\frac{\theta\pi}{180}\right|\right)\right) \cdot \sin\left(\left|\frac{\theta\pi}{180}\right|\right) && \text{(V.5)}
\end{aligned}
$$

At last, separate the feature points into different groups: eyebrow, eye, nose and mouth.

After training the Bayesian Network, the next step is to evaluate this network using different database. When evaluating, we choose three other different databases.

2 Experiment Results

Figure V.4 shows some examples from CUbiC FacePix(30) database. We evaluate pose recognition of our system on this database. Firstly, 45 images are picked from database randomly. These 45 images include 6 different angles: right turn (15°, 30°, 45°) and left turn (15°, 30°, 45°). In these images, the pose with 15° and 30° angles belong to the same cluster. The pose with 45° is classified to another cluster. Table

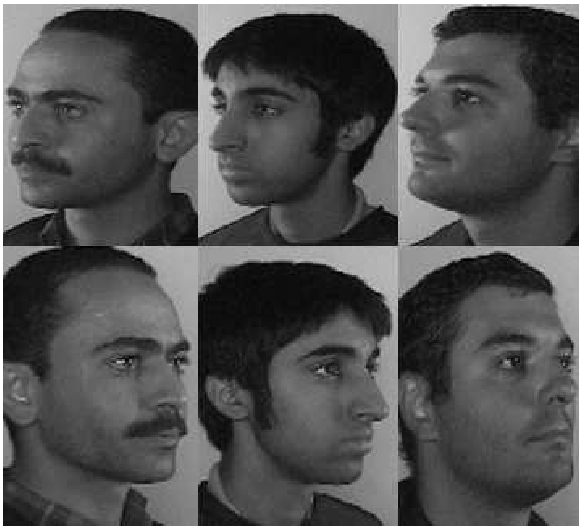

Figure V.4: An example of the test data from CUbiC FacePix(30) database [27], [28] we used in Bayesian Network. The first row is left–turn pose (45°). The second row is right–turn pose (45°).

V.3 gives the result of all kinds of pose recognition.

The accurate rate for left-turn 15° images is lower than the others. Usually, these images may be misclassified to the frontal images. As shown in Figure V.5, some misclassified images are given. The first image is left-turn (15°) and the right image is frontal. In our experiments, the left image is misclassified to frontal image.

We also give the conclusion from the experiments that not all the feature points are useful for the pose recognition. The feature points from pose-invariable components are more informative while the feature points from components like mouth will even decrease the accurate rate. That is because their shapes deform easily and

Table V.3: Pose Recognition with Five Levels

Pose Recognition		
Pose	The group this pose belong to	Accurate Rate
Right(15°)	right	100%
Right(30°)	right	100%
Right(45°)	more-right	100%
Left(15°)	left	93%
Left(30°)	left	100%
Left(45°)	more-left	100%
Images from video	uncertained	95%

Figure V.5: Misclassified images

Table V.4: Pose Recognition with Node Mouth and without Node Mouth

Pose	Without node $'$mouth$'$	With node $'$mouth$'$
Right	100%	86%
Left	97%	67%

obviously. When applying the evidence to the network, we can ignore the feature points from mouth and the accurate will increase. Table V.4 compares the accurate rate with and without the node $'$mouth$'$.

After evaluating the network based on CUbiC FacePix(30) database, we generate a small database ourselves. This database is extracted from some short videos in order to evaluate the pose recognition accurate rate in spontaneous situation. The result can be found in the last row of Table V.3.

D Experiment Results of Expression Recognition Based on FEED Database

In order to evaluate the expression recognition of our system, we use the FEED Database [35]. We use the images with happy expressions as shown in Figure V.6.

Figure V.6: The images with happy expression from FEED Database [35].

Table V.5: Smile Recognition Based on the FEED Database

Smile recognition	
Expression	Smile(front)
Accurate Rate	100%

The experiment result based on FEED Database is shown in Table V.5.

E Experiment Results of DBNs

We will give the results of pose recognition, smile recognition and surprise recognition separately here. For each recognition, we define some simple action elements as shown in Table V.6, V.7, V.8.

We choose 50 persons from Korea Face database for training the DBNs. Each person's images contain variety of head poses. We use other 30 persons to test the head pose recognition. Table V.6 also gives the results of recognizing these four actions.

For evaluating the smile recognition, we apply the FEED Database which has introduced previously. We choose 30 persons to train the DBNs and 20 persons to test. The results are shown in Table V.7 too.

For surprise recognition, we apply the JAFFE database. Some examples are

Table V.6: The Definition of Action Elements

$T = t - 1$	$T = t$	The name of action elements	Accurate rate
left	left	keep left	92%
left	right	turn right	94%
right	right	keep right	100%
right	left	turn left	92%

Table V.7: The Definition of Smile Actions

$T = t - 1$	$T = t$	The name of action elements	Accurate rate
smile	smile	keep smile	75%
smile	normal	finish smile	90%
normal	smile	smile	95%
normal	normal	keep normal	90%

shown in Figure.V.7. We define four kinds of basic motions too. TableV.8 shows the definition and the accurate rate of these motions.

Figure V.7: Some examples with surprise expression from JAFFE database.

F Experiment Results Analysis and Comparisons with Other Work

The method we proposed here takes the different head poses and expressions as different feature points' distributions. There are some conditions for using this

Table V.8: The Definition of Surprise Actions

$T = t - 1$	$T = t$	The name of action elements	Accurate rate
normal	surprise	surprise	80%
surprise	surprise	keep surprise	86%
surprise	normal	finish surprise	80%
normal	normal	keep normal	75%

method. Firstly, the poses and expressions must have clear and unique feature points' distributions. Secondly, the distribution can reflect the main and universal characters of the poses and expressions we are going to recognize. A typical example is the recognition of head poses. We pick 21 feature points from human face. For each head poses, the distributions are clear and unique as shown in previous sections which makes the recognition easy and fast. When come to the expressions, some posed expressions can be recognized by this method because their feature points distributions are clear such as the smile recognition experiments. However, in real-life, spontaneous expressions appear frequently. Most of the spontaneous facial expressions are activated without significant facial appearance changes, which means the amplitudes of the spontaneous facial expressions are smaller than those of the posed facial expressions. For spontaneous expressions, more researches should be done about the universal characters of these expressions.

Dimension decrease is always an important issue in Computer Vision. For BN and DBN, this is especially important because when calculating the covariances in BN and DBN, the dimension is the product of dimensions of the node and its parent nodes. If the dimension of the feature is large, then after product, the dimension will

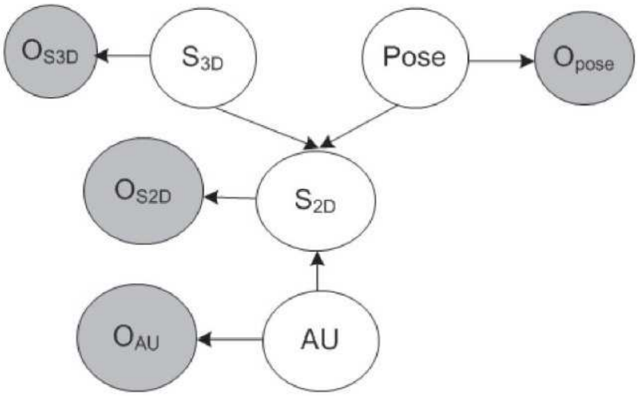

Figure V.8: A graphical model to represent the causal relationships in [2]

increase dramatically and the speed of the system will decrease too. So we should control the dimension of the features we choose. Here we choose 21 points as the features.

In [2], BN method are applied here. The goal is to recognize the spontaneous facial action. There are 28 feature points used here and the head pose are separated into 3 groups while we use 21 points and separate the head pose into 5 groups. In figure V.8, a graphical model in [2] is shown. The gray nodes is the observations. We can find that the state of head pose has already been got by other methods and BN just is used as the last judging tool. In this case, the dimension in BN will be much smaller. Our proposed method applies the BN directly without introducing other recognition methods and get the same results in head pose recognition.

Section VI

Conclusions and Future Work

In this paper, we propose a whole system for facial actions recognition. For feature detection, we propose a fully automatically facial feature points detection method using local Gabor filter bank and PCA. The experiment results have suggested several conclusions. Firstly, local Gabor filter bank outperforms global Gabor filter bank by reducing the dimension of features and obtaining same performance. Secondly, PCA can significantly reduce the dimensionality of the feature with most information kept. The PCA method can be affected by different illuminations. So illumination normalization can improve the results too. Because the dimension of the feature vector is reduced dramatically, the efficiency of the system is improved.

This automatically feature points detection system can be improved in several ways: Firstly, we plan to improve the classifier part to get a better accurate rate. What is more, we will perform the fully automatic feature points detection in the case of expressions and poses exist.

Secondly, we introduce the proposed BN and DBNs for facial actions recognition without 3D information. In BN, The different states of pose and expression are presented by different distributions of feature points. Some researchers reconstruct 3 dimensional face from 2 dimensional images and then project the feature points to a 2 dimensional plane. However, sometimes it is difficult to get 3D information. Through these experiments, we find that 2 dimensional information can be applied directly and the results are satisfied. Also from the experiments, we can find that not all the feature points are effective for different kinds of facial actions recognition. For example, for pose recognition, the feature points from mouth will decrease the result but in expression recognition, feature points from mouth contain a lot of information and become very important. So we can consider to separate many feature points into different groups and for each kind of facial actions, apply the most effective one.

Extending the BN into DBNs, a set of action elements are defined and recognized in experiments. These actions recognized here are defined between two adjacent time slices. In practice, some actions can appear instantaneously and disappear suddenly. In order to catch these actions, the interval between two time slices must be tiny. This requires a very fast feature tracking system. What is more, in order to guarantee the performance, more other features should be added.

Future research directions are realtime face tracking and more other kinds of facial actions recognition. Finally, we want to realize the realtime communication between human and computer.

Bibliography

[1] Gianluca Donato, Marian Stewart Bartlett, Joseph C. Hager,Paul Ekman, and Terrence J. Sejnowski, "Clasifying Facial Actions," IEEE Transactions on Pattern Aanlysis And Machine Intelligence, Vol. 21, No. 10, October 1999

[2] Yan Tong, Jixu Chen and Qiang Ji, "A Unified Probabilistic Framework for Spontaneous Facial Action Modeling and Understanding," IEEE Transactions on Pattern Aanlysis And Machine Intelligence, Vol.32, No.2, February 2010.

[3] Guillaume Heusch, "Bayesian Networks as Generative Models for Face Recognition," Idiap Research Institute

[4] Baback Moghaddam, Tony Jebara and Alex Pentland, "Bayesian face recognition," Pattern Recognition 33 (2000) 1771-1782

[5] Sangho Park, J.K.Aggarwal, "A hierarchical Bayesian network for event recognition of human actions and interactions," Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712, USA

[6] Erik Murphy-Chutorian, Student Member, IEEE, and Mohan Manubhai Trivedi, Fellow, IEEE, "Head Pose Estimation in Computer Vision: A Survey," IEEE Transactions on pattern analysis and machine intelligence, VOL. 31, NO. 4, APRIL 2009

[7] Lyons, M.J., Akamatsu, S., Kamachi, M., Gyoba, J." Coding Facial Expressions with Gabor Wavelets," Proc. of FG98 (1998) 200-205

[8] Chibelushi, C.C., Bourel, F. " Facial Expression Recognition: A Brief Tutorial Overview," CVonline: On-Line Compendium of Computer Vision 2003

[9] Krinidis, S., Buciu, I., Pitas, I., " Facial Expression Analysis and Synthesis: A Survey," Proc. of HCI03, Vol. 4 pp.1432-1436, 2003

[10] J. Wang, L. Yin, X. Wei, and Y. Sun, "3D Facial Expression Recognition Based on Primitive Surface Feature Distribution," Proc. IEEE Intl Conf. Computer Vision and Pattern Recognition, vol. 2, pp. 1399-1406, 2006.

[11] Y. Chang, M. Vieira, M. Turk, and L. Velho, "Automatic 3D Facial Expression Analysis in Videos," Proc. Analysis and Modelling of Faces and Gestures, pp. 293-307, 2005.

[12] B. Braathen, M.S. Bartlett, G.C. Littlewort, E. Smith, and J.R. Movellan, "An Approach to Automatic Recognition of Spontaneous Facial Actions," Proc. Fifth IEEE Intl Conf. Automatic Face and Gesture Recognition, pp. 345-350, 2002.

[13] M.S. Bartlett, G.C. Littlewort, M.G. Frank, C. Lainscsek, I.R. Fasel, and J.R. Movellan, "Automatic Recognition of Facial Actions in Spontaneous Expressions," J. Multimedia, vol. 1, no. 6, pp. 22-35, Sept. 2006.

[14] F. Dornaika and F. Davoine, "Simultaneous Facial Action Tracking and Expression Recognition Using a Particle Filter," Proc. Intl Conf. Computer Vision, vol. 2, pp. 1733-1738, 2005.

[15] Y. Chang, M. Vieira, M. Turk, and L. Velho, "Automatic 3D Facial Expression Analysis in Videos," Proc. Analysis and Modelling of Faces and Gestures, pp. 293-307, 2005.

[16] B. Bascle and A. Blake, "Separability of Pose and Expression in Facial Tracking and Animation," Proc. Intl Conf. Computer Vision, pp. 323-328, 1998.

[17] Z. Zhu and Q. Ji, "Robust Real-Time Face Pose and Facial Expression Recovery," Proc. IEEE Intl Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 681-688, 2006.

[18] Li Dang, Fanrang Kong, "Facial Feature Point Extraction Using A New Improved Active Shape Model," 3rd International Congress on Image and Signal Processing (CISP2010), 2010

[19] Ching-Ting Tu and Jenn-Jier James Lien, "Automatic Location of Facial Feature Points and Synthesis of Facial Sketches Using Direct Combined Model," IEEE

transactions on systems, man, and cyberneticspart b: cybernetics, Vol. 40, No. 4, August 2010

[20] Danijela Vukadinovic, Maja Pantic, ”Fully Automatic Facial Feature Point Detection Using Gabor Feature Based Boosted Classifiers,” IEEE International Conference on Systems, Man and Cybernetics Waikoloa, Hawaii October 10-12, 2005.

[21] Anima Majumder, L. Behera and Venkatesh K Subramanian, ”Automatic and Robust Detection of Facial Features in Frontal Face Images,” 2011 UKSim 13th International Conference on Modelling and Simulation

[22] Hong-Bo Deng, Lian-Wen Jin, Li-Xin Zhen, Jian-Cheng Huang, ”A New Facial Expression Recognition Method Based on Local Gabor Filter Bank and PCA plus LDA,” International Journal of Information Technology, Vol. 11, No. 11 2005

[23] R. O. Duda, P. E. Hart, D. G. Stork, Pattern Classification. Wiley, New York (2001)

[24] G. Cowell, P. Dawid, L. Lauritzen and J. Spiegelhalter, ”Probabilistic Networks and Expert Systems,” Springer, 1999.

[25] M.I. Jordan, Z. Ghahramani, T.S. Jaakkolla and L.K. Saul, ”An Introduction to Vari- ational Methods for Graphical Models,” Machine Learning, vol. 37,

pp.183-233, 1999.

[26] Kevin P. Murphy, "The Bayes Net Toolbox for Matlab," Department of Computer Science, University of California, Berkeley Berkeley, CA, 94720-1776

[27] Black J, Gargesha M, Kahol K, Kuchi P, Panchanathan S, "A framework for performance evaluation of face recognition algorithms," ITCOM, Internet Multimedia Systems II, Boston, July 2002.

[28] Little G, Krishna S, Black J, Panchanathan S, "A methodology for evaluating robustness of face recognition algorithms with respect to changes in pose and illumination angle, " ICASSP 2005, Philadelphia, March 2005.

[29] J. Pearl, "Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference," Morgan Kaufmann, 1988.

[30] Kevin P. Murphy, "Inference and Learning in Hybrid Bayesian Networks," 2008 Technical Report CSD-98-990, Dept. of Computer Science, Univ. of California Berkeley, 1998.

[31] Sridhar Mahadevan,"The Junction Tree Algorithm," University of Massachusetts.

[32] David Kahle,Terrance Savitsky,Volkan Cevher and Stephen Schnelle, "Junction Tree Algorithm," Rice University.

[33] S. L. Lauritzen and D. J. Speigelhalter, "Local Computations With Probabilities On Graphical Structures And Their Application To Expert Systems," The Journal of Royal Statistical Society, vol. 50, pp.157-224, 1988.

[34] G. Cowell, P. Dawid, L. Lauritzen and J. Spiegelhalter, "Probabilistic Networks and Expert Systems," Springer, 1999.

[35] Frank Wallhoff, Facial Expressions and Emotion Database http://www.mmk.ei.tum.de/waf/fgnet/feedtum.html, Technische University Mnchen 2006

[36] T. Dean and K. Kanazawa, "Probabilistic Temporal Reasoning," Proc. Seventh Natl Conf. Artificial Intelligence, pp. 524-528, 1988.

[37] Kevin P. Murphy, "Dynamic Bayesian Networks," www.ai.mit.edu/ murphyk 12 November 2002

[38] K.Murphy, S.Mian, "Modelling Gene Expression Data using Dynamic Bayesian Networks," Technical Report, Computer Science Division, University of California, Berkeley, CA, 1999.

Acknowledgement

It is a pleasure to thank those who made this thesis possible. First of all, I would like to express my sincere gratitude to my supervisor, Prof. Sang-Woong Lee for his guidance, encouragement and continuous support throughout the course of this work. Without his instruction, impressive kindness and patience, I could not have completed my thesis.

I also owe my deepest gratitude to the committee members, Prof. Pan-Koo Kim and Prof. Hee-Deok Yang, for their invaluable supports, detailed reviews and excellent advices during the preparation of this thesis.

My colleagues in the computer vision and multimedia lab have made available their support in a number of ways during the two years.

Last but not least, I would like to thank my family for their constantly unconditional support and encouragement and also to my dearest friends for their generous help all the time.