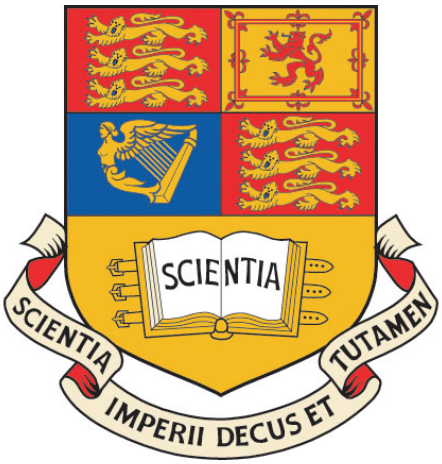
Imperial College London

Department of Computing



Detecting eating and talking from webcam videos

Wenyang Cai

Submitted in part fulfilment of the requirements for the degree of

Master of Science in Computing of Imperial College London, September 2014

Abstract

Automatic Speech Recognition(AVR), a integral part of Human Computer Interface(HCI), was extended to Audiovisual Automatic Speech Recognition(AV-AVR) in recent twenty years. Visual information was proven to have significant value for improving performance in AV-ASR. There is a large amount of research on improving performance of recognition speech. However, facial expression and non-linguistic vocalisation is also important part of human daily conversation. Detecting these action from speech could also improve performance of AV-ASR. This project try to integrate a series of action to extract proper features from relatively free recorded video. Then use different features to train a classifier and test and analyse the influence of using different features. In the process of face alignment, we tried two different face tracker for tracking facial points from video frames. The Intraface [16] and DRMF [2]. Intraface is more accurate and DRMF provide more point which could be helpful. In the process of accurate appearance feature vector, we tried to remove head-pose using deformable model and extract texture feature using block Local Binary Pattern(LBP). We try to train, test and analyse using different features train a Support Vector Machine Classifier to classify three classes of frames and sequence, normal face, eating and talking. This project evaluate some approaches to extract features and influence of different combination of features on classification performance.

Acknowledgements

I would like to grateful acknowledge Dr. Stavros Petrdis, who guide me through the whole project and gave me a lot of assistant in the project. And his friend who ever helped me. Without them, I am unable to finish the whole project.

I would like to appreciate my supervisor Maja Pantic for proposing this project.

My dear parents who support me financially and mentally.

Thank all my friends who also stayed in the computer room and fighting the project together, they gave me so many suggestion and help with a lot of technical details, especially acknowledge He Wei who helped me a lot on latex formatting. We being through a wonderful and memorable time.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Automatic Speech Recognition is a very important and popular technology in Human Computer Interactions. In recent years, device and software that support ASR system are increasing rapidly. Diversification of interacting with a system is increasing. People are not satisfied with just interacting with a computer through mouth and keyboards. Automatic Speech Recognition is a very good technology for HCI as human-human communication are mostly through speech. Audiovisual automatic speech recognition has shown significant improvement by including visual information. In human-human communication, people are able to understand others by reading the signal of facial expression, some non-linguistic vocalization, gestures. [10] Laughter are commonly used in human social conversation. However, the research in recognising the non-linguistic vocalization is rare. Moreover, people usually not just talking during conversation, they could also be smiling or eating while interaction with other people. This project intend to try to automatically distinguish video sequence of some different facial expressions, such as normal face, eating and talking.

1.2 Thesis

This project try to detect eating, talking and normal state of face from videos that are recorded using webcam. The process of project contains two main steps, first step is to extract suitable features that could accurately represent facial expression of each frame. Second step is to experiment use extracted feature to train classifier that could correctly classifier a video sequence.

1.3 Contribution

In this project, we integrate a sequence of process to extract video features with existing algorithms. The procedure contains tracking facial feature point using Intraface, warping, remove-head pose using deformable model, extract features using local binary pattern. In this project, we experiment using train a Support Vector Machine classifier with different features. We tried appearance feature, shape feature and combination of both. For appearance feature, we tried to extract 1-block uniform pattern feature and 3-block uniform pattern feature. We also explore the influence of normalize each feature vector by each video. By normalize, I mean substract its mean and divide by its standard deviation.

1.4 Outline of Report

Chapter 2: Present a brief introduction in Automatic Speech Recognition(ASR) and Audio-visual ASR, visual front end including some definition in this field, basic facial expressions, some application of application in audio-visual biometric.

Chapter 3: Present the main procedure of extracting appearance feature and shape feature. Compare two different tracker of tracking facial feature points. Describe the methodologies used in the procedure.

Chapter 4: Present the results of classifier and analysis from different aspects.

Chapter 5: Evaluate the process of extracting facial features and experiments and describe the limitation and future work of this project.

Chapter 2

Background

In human-human communication, human would be able to understand each other through variety feedbacks. We are able to understand others through conversation, facial expression, gesture, even intonation, mood. This phenomenon was introduced into human-computer interactions(HCI). Using computer to automatically analyse human face, voice and behaviours would be helpful in improving HCI. Many technologies are developed for computer to automatically recognize human actions and emotions such as automatic speech recognition [11], automatic recognize facial expression [16]. These technologies not only can be used for HCI, but also can be used in other area such as avoid impostor attacks [1]. Face and voice are important and personal biometric characteristics. Compare to traditional knowledge-based and token-based person recognition method, biometric recognition technology are more convenient and safer. With the increasing in computer computing power and development in computer vision, Automatic Speech Recognition(ASR) attracts more attention in recent years, from traditional audio-only ASR to audiovisual ASR, a huge progress was made in ASR. AV-ASR could be used on automatic generation of voice and facial animation from arbitrary speech.

2.1 Automatic Speech Recognition

The intuition of Automatic Speech Recognition is to convert a spoken sentence into to readable text in real time by computer and he ultimate goal of ASR is to let a computer 100% accurately recognise speech of any person under any environment. The beneficial of ASR is quite obvious, it can help deaf people listening by convert speech into readable text and help hard-reading people on reading by convert readable text into voice. The search engine may not be limited on text searching but also speech search. It has been studied for over 50 years, recent work has a great improvement in this area. The accuracy of recognition highly depended on robust information channel, background

environment, the training data base and the adaptation of speaker to database. With decreasing price of computing power, Speech recognition techniques are widely used on mobile devices, like Siri.

People can not only know what others says by listening to speaker, they can also know what others says by watching. Deaf people is able to communicate with each other by reading mouth movements. Visual modality was proved to have positive influence on reducing noise in ASR and the history can be quantified back to 1954 [11]. [11] gave three key reasons to include vision information in human speech recognition. Firstly it helps audio source localisation, visual information of tongue, teeth, and lips provide complementary information of articulation. Secondly, it is beneficial for distinguish confusable acoustics such as unvoiced consonants /p/ by providing information of facial muscle movements. Thirdly, facial muscle movements are robust information for ASR. This technique of using visual information to recognise speech is known as automatic lip-reading or speech-reading in ASR [11].

Audiovisual ASR uses both visual modality and audio modality in recognizing speech. There are two main challenge in AV-ASR from the original ASR. One problem is how to extract visual features, the other problem is how to combine it with audio features. Visual speech information mainly on speaker's face. Extracting visual feature requires techniques from other field, such as face detection, head pose estimation, face feature localisation, tracking, feature extraction and other techniques to extract useful features from image with a face. Those techniques are prerequisite for incorporating AV-ASR in HCI. The second problem is how to make the combination of both modality produce better performance than using single modality. There are three type of fusion: combine audio and visual information at the feature level; combine audio and visual classifier scores at decision level; and a combination of both. [11] named two areas, visual front-end design and audiovisual fusion.

2.2 Visual Front End

A major problem in audiovisual automatic speech recognition is extracting visual feature from images. The process is to extract visual speech features from videos or a sequence of images. Generally, visual speech feature can be classified as three types: appearance feature, shape feature, combination of both [11]. Appearance feature usually means the image feature of Region of Interest. Image of (ROI) used to be directly used for training and classification. However, image data contains many noises and influenced by the lighting condition. Some techniques in computer vision are used to extract image features from Region of Interest. A good image descriptor is required for improving classification performance. Local Binary Pattern is a very good texture descriptor. Shape feature

usually means contour of speaker's face, especially speaker's lips or including jaw and cheek. Shape feature usually means geometric-type features, there are many ways to describe shape feature of a face, such as statistical shape model or image moment descriptor of mouth, these model would be able to contain information of the height, width and other information of a mouth. Combined feature usually is the joint of both shape and appearance feature vectors or a model that include both features such as active appearance model.

In order to get appearance feature and shape feature or combined feature, there are some preprocessing steps should before extracting feature. Face detect, detecting the position of face, facial tracking, tracking facial feature points from frame to frame, face alignment, align facial feature points to a face, and ROI extracting technique extract features from ROI, these techniques are all required for extracting visual speech features from videos. There are many methods for lip contour extraction: Snake by Kass et al(1998), Deformable Template by Yuille et al (1989), Active Shape Models by Cootes et al(1995), Active Appearance Models(AAMs) by Cootes et al(2000). The later three models are all called Parameterized Models, also in recent years, some parameterized model are extended and developed. In order to remove head-pose, some deformable model of face are used to decompose the head points and remove head pose. If the head is not facing the front, appearance image of image need to be warped to frontal.

2.2.1 Face Detection

Face detection is to detect the location and size of single face or several faces in one digital image. There are several important aspects would influence face detection: background, head pose and lighting condition. There are two main approaches for face detection, one is non-statistical way, using traditional image processing techniques, the other is statistical approach, using statistical models. [11] uses traditional image processing techniques, such as colour segmentation, edge detection, image thresholding, template matching or motion information. Some using statistical modelling such as Fisher Discriminant detector, Distance From Face Space(DFFS), Gaussian Mixture Classifier(GMM)and neural networks such as Artificial Neural Networks(ANN). Once a face is detected, use face alignment techniques to estimate the location of several facial feature around the face.

2.2.2 Region of Interest

Usually the choice of Region of Interest depends on the purpose of the project. In AV-ASR it usually include large part of the lower face, such as the jaw, and cheeks or even the entire face [11], as when people speak, the lower face would show some movements. In my project, the ROI only

contains the gray-scale values of mouth region, which is scaled to $8 * 32$ size square region. [11] report that experiments shows that including jaw and cheeks was beneficial. As the tracker I use for tracking face does not include the jaw and cheek of the face, I have to just include the mouth.

2.2.3 Visual Feature and Postprocessing

Usually the image is not directly used for classification, because of the influence of noise and brightness. There are many descriptors can be used to represent an image and reduce the influence. In addition, the dimensionality of image vector is usually very large, it is not suitable for classification. The choice of visual feature is depends on the requirements of the project. The most popular descriptor for AV-ASR is texture feature descriptor. If the dimension of feature vector is too high, the common dimension reduction methods are traditional linear transforms. [11] gave some most commonly applied methods, Principle Component Analysis(PCA), discrete cosine transform(DCT), discrete wavelet transform, Hadamard and Haar transforms and a linear discriminant analysis based data projection. As the visual feature are extracted from variety of lighting condition and different face. These effects can be remedied by normalization, i.e. for each visual feature vector subtract it's mean and divide by the standard deviation. For different classification methods, some necessary steps may be needed before use feature vector to do classification.

2.3 Facial Expression

Facial expression could directly response to people's inner thoughts and feelings. Human face is major site for sensory input and outputs [8]. Generally, face would show four kinds of signals: static facial signal, slow facial signals, artificial signals and rapid facial signals [8]. Rapid facial signals underlie facial expressions. [8] indicate that rapid facial signals generally show five types of messages: affective attitudinal states and moods which means emotions, emblems, manipulators, illustrators, regulators. For example, smile belongs to regulators and chewing belongs to manipulators. Automatic analysis of facial signals such as rapid facial signals have potential applications in many areas. Layers, security, police could be use automatic analysis of facial signal system to monitoring and interpreting human facial signal and gain important information. For example, monitoring human reaction during inquisition, inquisitor would be able to tell whether a person is lie or not. Machine analysis of facial expression forms an important part of affective human-computer interface designs [8]. Research on machine analysis of facial expression mainly focuses on facial affect and facial muscle action detection [8]. Technologies used in this area are face detection, facial feature extraction, facial muscle action detection and emotion recognition [8].

2.3.1 Nonlinguistic Vocalization

Non-linguistic vocalization is defined as brief, discrete, non-verbal expressions of affect in both face and voice [9] such as laugh, sighs, sob. Nonlinguistic Vocalization can be used to detect speaker's affective state and facilitating affect-sensitive HCI [9]. Many non-linguistic vocalization are able to show speaker's true reaction. [9] shows there are 0.8 percent of time on laughing while talking.

2.4 Audio-Visual Biometrics

Biometrics recognition is using the utilization of physiological and behavioural characteristics for automatic person recognition [1]. Face and voice especially face are very personal biometric characteristics. Biometric Characteristics are often used for person identification and person verification. [1] indicates that traditional person recognition contains two types of method, one is knowledge-based, people using password, pin belong knowledge belong to that type, and the other is token-based, such as using a card with magnetic chip. Traditional methods are not good enough that they are either easily forgotten or stolen. Using biometric characteristics as personal identity could protect personal properties from being accessing by other person without worry about those problems. There are many biometric characteristics could be used as person identity such as Iris, fingerprint, hand, signature. Although the performance of using face and speech is not as accurate as using Iris and finger print, but the sensor for face and speech recognition cost much less than Iris sensor. Face and speech recognizer can not be high secure bank, they are suitable for some less secure place, such as entrance check.

Chapter 3

Processing and Methodologies

This chapter contains the procedure of extracting facial features, comparison of two tracker, example result of deform shape feature points and transform it to frontal, warp appearance feature point to frontal, extract uniform LBP features.

3.1 Processing Flow

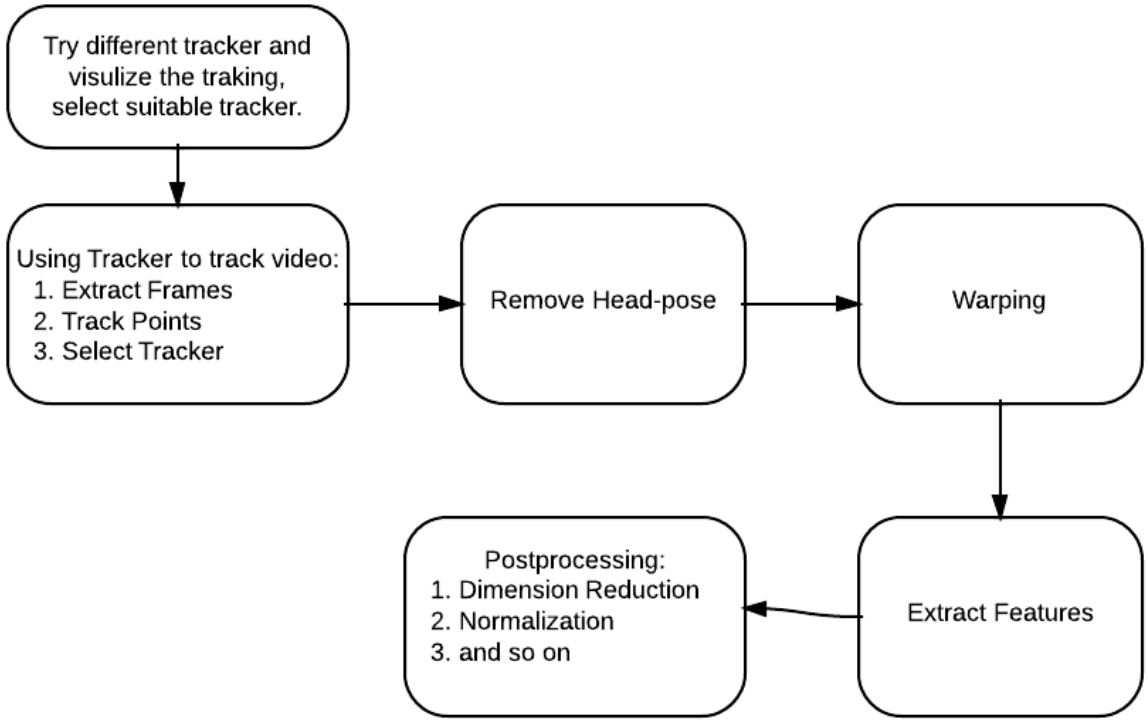


Figure 3.1: Main Procedure

Processing Chart Figure 3.1 shows main procedure of the whole project. At beginning, several trackers are tried to check the accuracy of tracking facial feature points. Intraface and DRMF are two trackers that were tried and test most. The two trackers are using different methods and also implemented in different languages. Intraface is programmed in c and matlab and it has great interface for matlab. There are two versions of DRMF, one of them was implemented by CUDA which uses parallel processing is quite fast. As the programme of DRMF doesn't integrate extract frames from videos. The images are extracted using external function, then tracked using DRMF. Feature that tracked using Intraface were used for experiment. Reason and comparison will be given in later section. Remove head-pose seems to be a very important part for this project, as subject's head moves frequently in many videos. Remove head pose means, transform facial feature points to frontal and also warp face image to frontal. After previous steps, each face image was scaled to same size grey image and the facial feature points are scale accordingly. Extracting features from the grey-image is to extract appearance feature of each face in the image. Post-processing is to standardize the feature vectors and label vectors, and prepare data for classification.

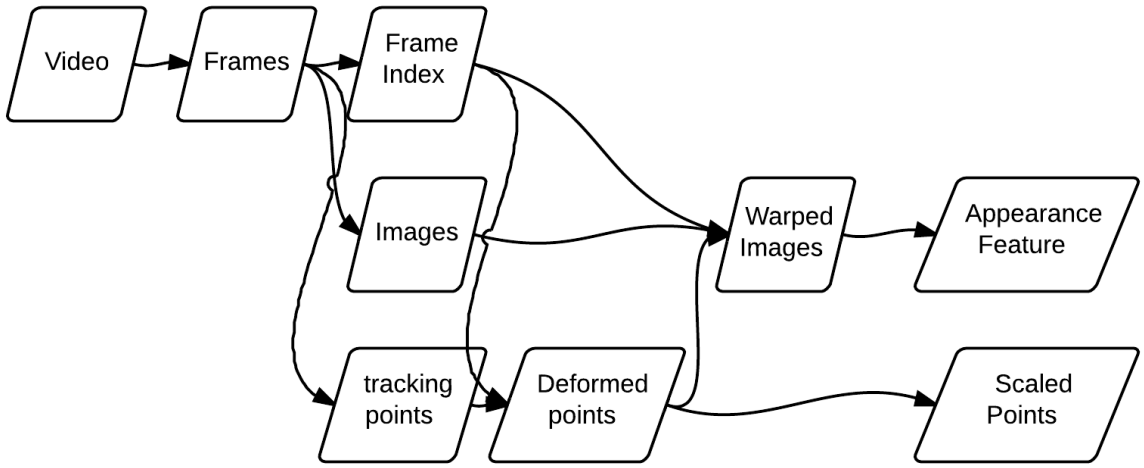


Figure 3.2: Data Flow

Data flow Chart Figure 3.2 shows how data flows from step to step. There are two types of encoded video, one is in format of flv and the other is avi. Extracting frames from videos is proceeded with Intraface and stored in formats of jpeg and mat which used for processing of matlab. Interface integrate the function of opencv and provide the interface for extracting frames from video. However, there are several situations that a track is unable to track a face in the image such as no subject in the image such as when the head is facing away from camera or face is partially not shown in the frame. Frame index points used to keep tracking those frame index that the tracker is able to tracker a face in a frame. Data 'Images' is the frame images stored in

mat format. Different tracker may track different number of characteristic facial points. Intraface tracks 49 facial points and DRMF tracks 66 points. Deformed points is the tracking point after the facial feature point were transformed to frontal face, as a deformable model was used, it is named as deformed points. Warped Images is the face after warping to frontal and usually it contains the part of face covered by mesh grid built by tracking points. Appearance feature is face feature extracted using local binary pattern (LBP). As the image size from image to warped images are changed, the points is rescaled from deformed point to scaled points.

3.2 Face Alignment

Face alignment is to align face in one image with respect to the same face in another image. Face alignment techniques are used to track characteristic facial points in image sequences. In this project, the aim of face alignment is to localise the feature points on face images. The points are usually around eyes, nose, mouth, and outline. Face alignment techniques are essential on face recognition, modelling and synthesis. There are three main different approaches Parametrized Appearance Models(PAMs), Discriminative approaches, Part-based deformable models. Parametrized appearance models contains many models such as active appearance models (AAMs), morphrable models, eigentrackings, and template tracking [16]. All these models are using Principle Component Analysis(PCA) method to parametrize a face. A face could approximately be decomposed as linear combination of shape basis and appearance basis. The problem of face alignment could be referred as minimising the difference between the constructed PAM and the face. Common approach is use Gauss-Newton methods [16]. Discriminative approaches are to learn the linear regression between the head move and appearance change. Part-based deformable model perform face alignment by maximising the posterior likelihood of part locations given image [16].

3.2.1 Active Appearance Model

Active Appearance Model (AAMs) is defined as a generative model of a certain visual phenomenon in [6]. AAMs are conceptually related to morphable models, constrained models and active blobs. In this project, it is refer to a model of face. As AAM is conceptually related to other parameterized appearance model and both Intraface and DRMF used parameterized models, so it is introduced as an example of parameterized appearance model for understanding purpose. According to [6], there are two types of AAMs, one refers as independent shape and appearance models, which model shape and appearance independently, and the other refers as combined shape and appearance models, which parameterized shape and appearance model with a single set of linear parameters [6]. Normally AAMs appears along with a fitting algorithm. However, in the following context, it only

refers to a model. [6] gives a well explaination about what is an AAM, most of following theory are from [6].

Shape Feature

Shape of a face s is defined by coordinates (x, y) of v vertices of face points and the mesh they built:

$$s = (x_1, y_1, x_2, y_2, ..., x_v, y_v)^T \tag{3.1}$$

s also can be expressed as a base shape s_0 plus linear combination of n shape vectors s_i:

$$s = s_i + \sum_{i=1}^{n} p_i s_i \tag{3.2}$$

Appearance Feature

For all pixels x in the mesh s_0, appearance $A(0)$ can be expressed by base appearance $A_0(x)$ and m appearance images $A_i(x)$.

$$A(x) = A_0(x) + \sum_{i=1}^{m} \lambda_i A_i(x) \qquad \forall x \in s_0 \tag{3.3}$$

AAMs are usually computed by applying Principle Component Analysis (PCA) to chosen images. The chosen images have a variety of shapes. The base shape s_0 is the mean shape and vector s_v is the eigenvector corresponding to the largest v eigenvalues. Base appearance A_0 and appearance A_i is computed by applying Principle Component Analysis to a set of shape normalised images.

Model $W(x : p)$ is the warp from s_0 to s. Then the model M set the appearance of $W(x : p)$ to $A(x)$.

$$M(W(x : p)) = A(x) \tag{3.4}$$

Combined AAMs

Combined AAMs just use parameter $c = (c_1, c_2, ...)^T$ to parametrize shape:

$$s = s_0 + \sum_{i=1}^{l} c_i s_i \tag{3.5}$$

and appearance:

$$A(x) = A_0(x) + \sum_{i=1}^{l} c_i A_i(x) \tag{3.6}$$

An example of AAM instantiation is clearly shown in figure 3.3.

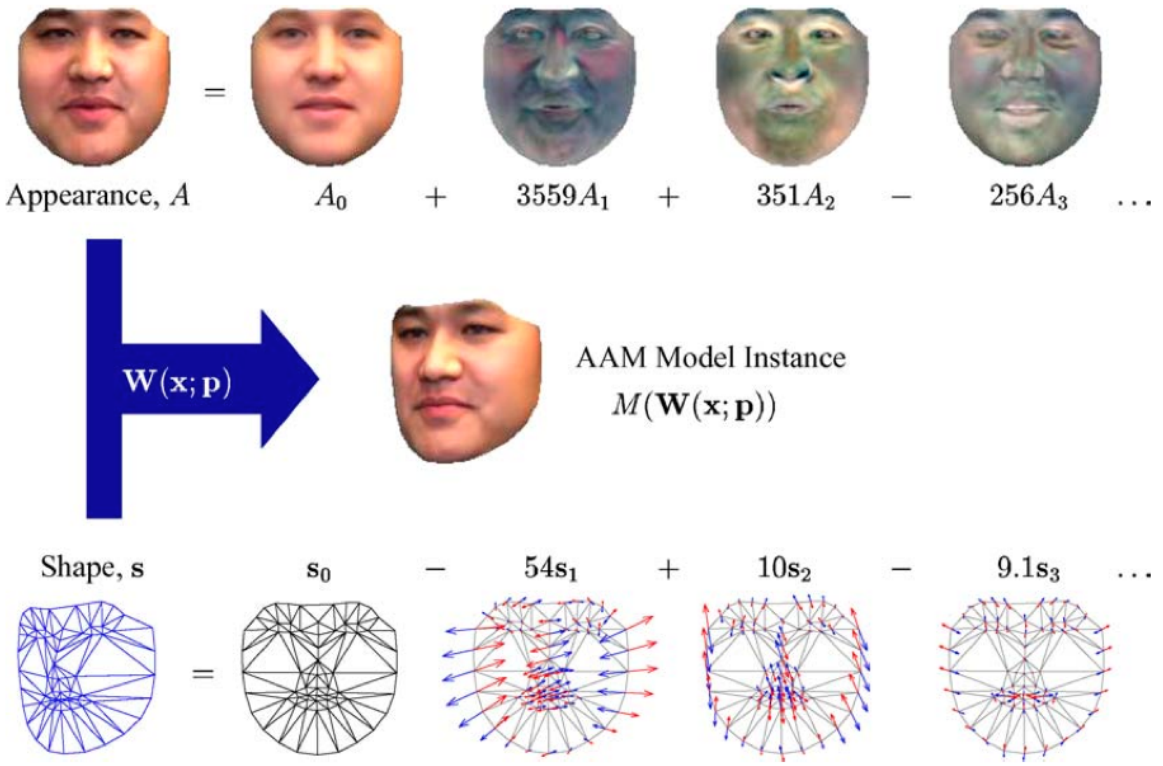


Figure 3.3: An AAM instantiation from [6]

3.2.2 Trackers

There are many trackers for tracking facial feature points. Different tracker may using different approaches, so they are suitable for different situations. I tried two main trackers for tracking characteristic facial points, one is Intraface [16] which use suppervised decent method, the other is DRMF [2] which use discriminative response map fitting. According to my using experiment, Intraface is very good at tracking motion face and DRMF is very good at fitting face model to a more standard face even with low resolutions. The number of facial points they track are also differnt.

Intraface

[16] implies image alignment can be posed as solving a nonlinear optimization problem. It uses Supervised Descent Method for minimising Non-linear Least Square(NLS) function, which avoids calculating the Hessian and the Jacobian that could be computationally expensive. For this reason, the running time of Intraface shows that the method is very effective and efficient.

Tracking Points Figure 3.4 shows the tracking points of Intraface. This tracker tracks 49 facial feature points. As you can see the eyes, nose, mouth, unfortunately the jaw and cheek may contain

visual information that may help classification.

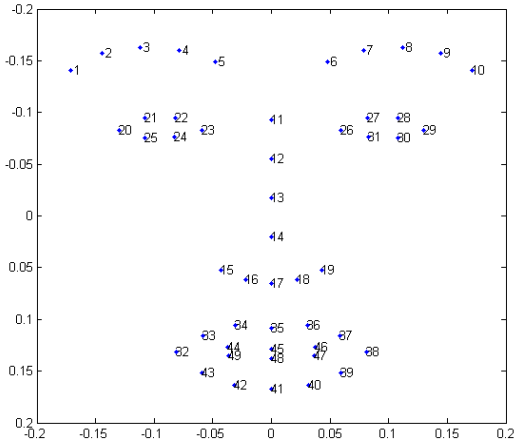


Figure 3.4: Intraface landmark points

Eating and Talking Sequence Figure 3.5 shows a sequence of image of eating tracked by Intraface. The point are aligned very precisely along the face. Figure 3.6 shows a talking sequences of image tracked by Intraface. The landmark points of mouth is very accurate.

Figure 3.5: Eating sequence tracked by Intraface

Figure 3.6: Talking sequence tracked by Intraface

DRMF

DRMF uses novel discriminative regression based on Constrained Local Models(CLMs) for face alignment [2]. The basic idea of DRMF is to fit a face for each frame of a video. After locating the position of a face, the tracker tries to fit a trained constrained local model to fit the face. Sometimes the fitting result is not very good and the landmark points of mouth region is not very accurate.

Tracking Points Figure 3.7 shows 66 facial feature points tracked by DRMF, the extra 17 points are the point around face bound. Other landmark points are at the same order as Intraface.

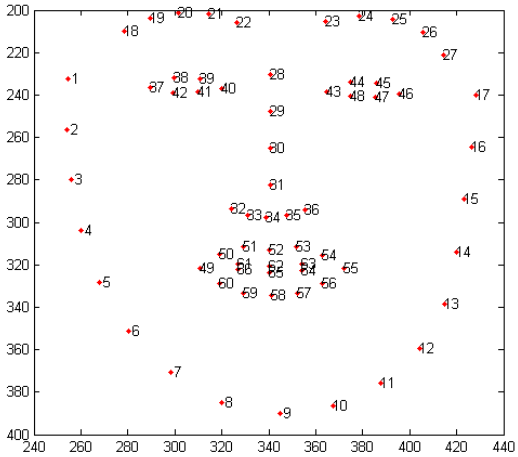


Figure 3.7: DRMF landmark points

Talking and Talking Sequence Figure 3.8 and 3.9 show image sequences of eating and talking tracked by DRMF. It is easy to see that the facial feature points are not aligned as good as Intraface. However, the advantage of DRMF is that with the extract bound points of face, we are able to extract the visual information about the jaw and cheek, which may be helpful for classification.

Figure 3.8: Eating sequence tracked by DRMF

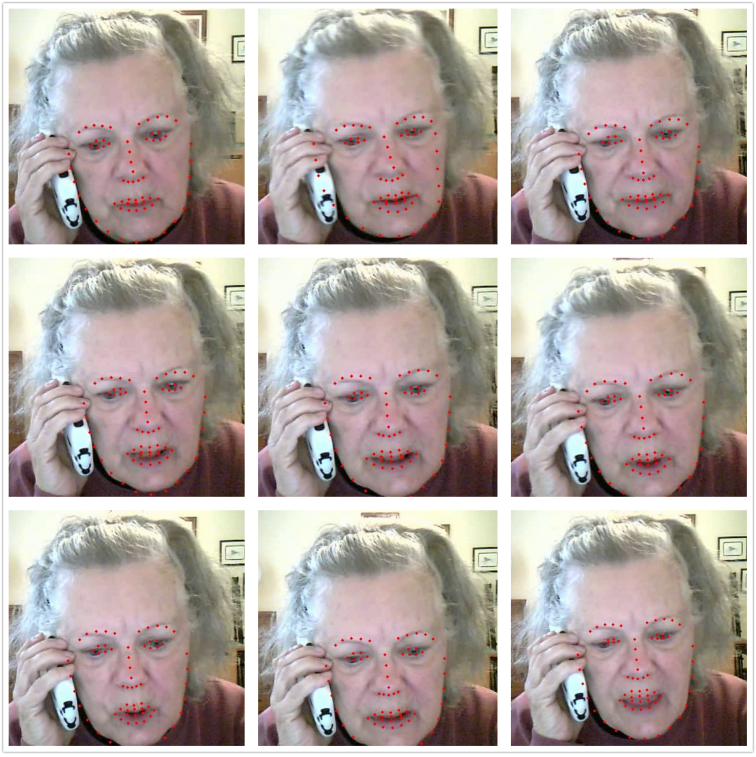


Figure 3.9: Talking sequence tracked by DRMF

3.2.3 Comparison

The following are some examples for comparing two trackers. Intraface is generally better than DRMF in accuracy and efficient. There are two versions of DRMF tracker one is implemented by CUDA language the other is by C language. Although the C version of DRMF is very slow and very easy to run out of memory, the version implemented by CUDA is very fast as CUDA is using parallel computing. However, for face points alignment, DRMF is not as accurate as Intraface. In some situations, DRMF try to fit a face and the fitting result is awful. The images with red points are tracked by Intraface, and the images with blue points is tracked by DRMF.

In figure 3.10, compare top left and top right images, we can see Intraface does not track the face of the smaller face, DRMF tracked the smaller face instead of the large face. This shows the advantage of DRMF on track face of low resolution face. We tried to use Intraface to track an image with multiple face, Intraface is able to track multiply faces in one image. So this means, Intraface is not good at tracking face on low resolution face. Compare the lower two images on the right hand side, the feature points of nose is not proper aligned, this is caused by imperfect of the fitting algorithm. The feature points of face bound are also not well aligned. Comparing the alignment of eye alignment on the left hand side of both tracker, Intraface is better than DRMF. The feature points of right eyebrows of the mid right image are also not aligned precisely.

In figure 3.11, in the top left image, DRMF is unable to fit a face to the face as the face is facing to the left. Of course, Intraface is unable to apply face alignment to an image doesn't show half of the face like this, it chooses to ignore this image. In lower two images on the left hand side shows two situations that DRMF fail to fitting face model to face because of the face is partial out of the image frame. The middle one shows that the fitting face points are force to stay in the frame and the mouth are moved up to close nose. The lower one image shows that as most of mouth region is out of frame, the model is fitted to the left eye forcefully. Middle and lower images on the right hand side shows that Intraface would ignore those points that are out of frame. It seems using Supervised Descent Method is very good at tracking face that moves. However, there is a bug of this tracker, while tracking the video, if the subject moves hand to cover the mouth, the points of mouth would be pushed upwards, and it will not come back even the hand is moved away, unless, the tracker loss the face and re-track the face.

Figure 3.10: Tracking result: images with red points on the left hand side are tracked by Intraface and images with blue points on the right hand side are tracked by DRMF

Figure 3.11: Tracking result, images with red points on the right hand side are tracked by Intraface and images with blue points on the left hand side are tracked by DRMF

3.3 Remove Head-pose

Remove head-pose means, to transform shape feature vector to frontal and warp appearance feature vector to frontal. Subjects were unrestricted while they were recording the video, their head pose are vary different. The coordinate of facial feature points are different for each video frame. In order to unite the head, we need to remove the head pose from the coordinates of facial feature points. If define the horizontal direction is x, the vertical direction is y and z is the direction, subject facing camera. It is very easy to remove head-pose in x-y direction, just by rotating and scaling the points would remove the head-pose. If the subject had a head-pose in x-z and y-z direction, it would be hard to find the correct transformation matrix for tracked facial points. The algorithm of removing head-pose from tracking points is [13]. Basically, it has a deformable 3-D Constrained Local Model(CLM), minimize the error of fitting the model with the 2D points and then remove head-pose and give the new 2D points. Figure 3.12 gave one example of original track points and deformed points. From figure 3.12, we can see this method could remove head-pose without losing or changing too much information of mouth shape.

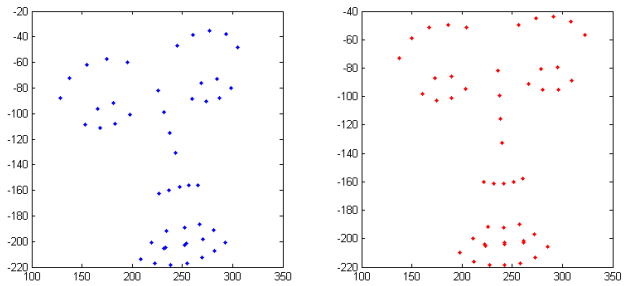


Figure 3.12: Traking points and Deformed Points Example

3.4 Warping

There are two types of features we need to extract, shape feature and appearance feature, the previous section just remove heap-pose of shape feature vector, in order to remove warp face image to frontal. Appearance of face on image is not correctly showing a face so it could not be directly used for classification, as the face is not facing to frontal. In order to have head-pose free face, we need to distort the image to frontal face that we need. In this step Piece-wise Affine [15] is used to warp the face image. According to the model in Active Appearance Model. We need to warping image from one to another with respect to the mapping of shape feature points set $p_1, p_2, ...p_n$ into another point set $p'_1, p'_2, ...p'_n$ which we calculated in the previous step. Each point is represented

as $p = [x, y]^T$. The mapping function would be:

$$f(p_i) = p'_i \quad \forall\ i = 1...n \tag{3.7}$$

Piece-wise Affine assume that f is locally linear. In 2-D framework like 2-D AAM, first find the mesh-grid constructed by all shape feature points. Delaunay triangulation is a good way to express it. In Delaunay triangulation, there are no points inside its circumcise. Then the problem is to find the mapping function f to map the triangle mesh of the first point set and the second point set. For each points on each triangle of the first point set I can be mapped to unique points on each triangles of the second point set I' by affine transformation, which is combination of translation, rotation and scaling. Assume p_1, p_2 and p_3 are vertices of a triangle in I, then the points inside the triangle can be written as:

$$\begin{aligned} p &= p_1 + \beta(p_2 - p_1) + \gamma(p_3 - p_1) \\ &= (1 - \beta - \gamma)p_1 + \beta p_2 + \gamma p_3 \qquad assume\ \alpha + \beta + \gamma = 1 \\ &= \alpha p_1 + \beta p_2 + \gamma p_3 \end{aligned} \tag{3.8}$$

As points p are in the triangle, so $\alpha \geq 0$,$\beta, \gamma \leq 1$. Then the corresponding points p' in image I' inside corresponding triangle would also satisfy the equation:

$$p' = f(p) = \alpha p'_1 + \beta p'_2 + \gamma p'_3 \tag{3.9}$$

We have the three points of a triangle, it is easy to determine the value of α, β and γ by solving two linear equation for a known point, $p = [x, y]^T$:

$$\begin{aligned} \alpha &= 1 - (\beta + \gamma) \\ \beta &= \frac{yx_3 - x_1y - x_3y_1 - y_3x + x_1y_3 + xy_1}{-x_2y_3 + x_2y_1 + x_1y_3 + x_3y_2 - x_3y_1 - x_1y_2} \\ \gamma &= \frac{yx_2 - xy_1 - x_1y_2 - x_2y + x_2y_1 + x_1y}{-x_2y_3 + x_2y_1 + x_1y_3 + x_3y_2 - x_3y_1 - x_1y_2} \end{aligned} \tag{3.10}$$

Equation 3.10 give the function to calculate α,β,γ, with the Equation 3.9, we can calculate all the points in image I'. Peudo-code of piece-wise affine warp would be:

1. For each point $p = [x, y]^T$ inside image I
2. Determine which triangle p is in
3. Use equation 3.10 to calculate α,β,γ
4. Use equation 3.9 to calculate the corresponding position of p'
5. set the value of p' in I to the value of p in I

6. end

Figure 3.13 is an example warping, the image was not highly distorted as the face is basically facing to the frontal. However, it is also to visual that the face is a little rotated to the right and there is a small change on the shape of mouth region.

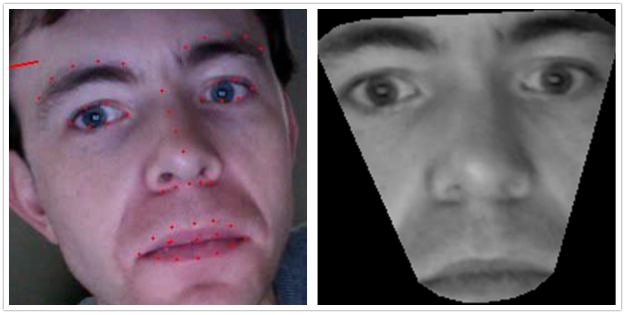


Figure 3.13: Talking sequence tracked by Intraface

3.5 Feature Extraction

Feature extraction is to extract suitable feature that is able to represent ROI according to the requirement of classification. The warped images are not directly used for classification, as the image value are sensitive to illumination, and noise. In addition, the dimensionality of image vector could be very large. An effective facial representation from original image is vital for classification. Experiments of [14] shows that Local Binary Pattern(LBP) features are effective and efficient for facial expression recognition. The best performance of [14] is obtained by combing Support Vector Machine and Boosted-LBP features. [14] also shows that LBP feature perform stably and robustly on low-resolution facial expression recognition. There are several reasons to use LBP: firstly LBP is robust to monotonic changes in illumination shown in figure 3.14; secondly, computational simplicity, the calculation only contain integer math and there is no need for preprocessing; thirdly, the time complex is $O(n)$.

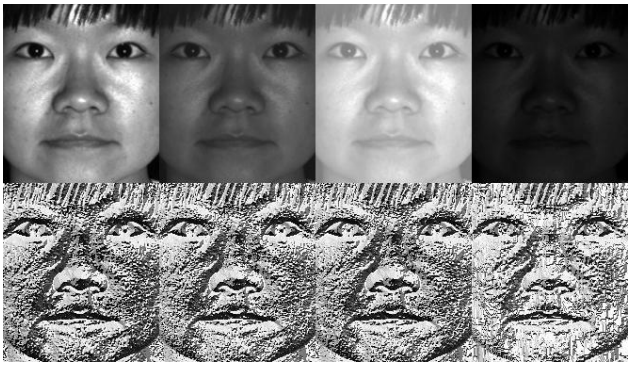


Figure 3.14: LBP Example[1]

3.5.1 Local Binary Pattern

Texture is an important characteristic of images and videos. 2-D surface can be characterized by two properties: pattern and contrast. In the warping step, the image was transformed to gray scale and we need a gray-scale rotation invariant pattern to measure. LBP is invariant to any monotonic gray level change and it is easy to compute. Ojala et al(1996) introduced the fist generation of LBP operator [14], it was proved to be a powerful texture descriptor. Figure 3.15 show how to calculate LBP value of a pixel and the contrast of the pixel. The binary number could be used

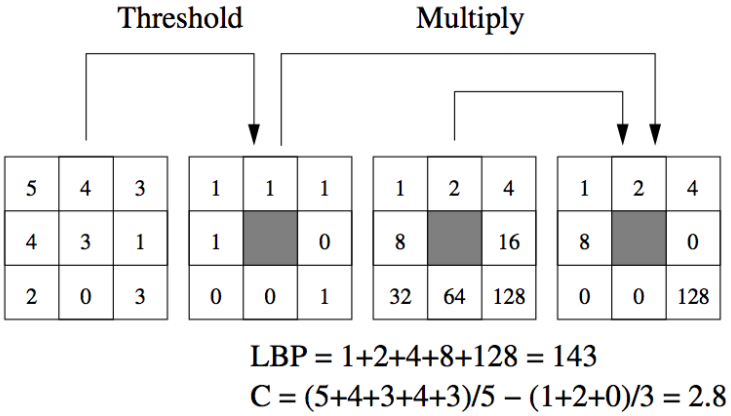


Figure 3.15: An Example of Calculating LBP code and a contrast measure of single pixel, image from [5]

for coding types of curved edges, spots, flat area, etc as shown in figure 3.16. Figure 3.16 is using circular pattern. In implementation, the on-grid points($g_0 - g_4$points in figure 3.17) directly use sample data for calculation, but the off-grid points(the points in the second grid($P = 8, R = 1$), not in grid($P = 4, R = 1$)) in figure 3.17) are calculating using interpolation. The 3X3 structure can not capture dominant feature for large scale structure [14]. So Ojala et al extend LBP to multiple scales as shown in figure 3.17.

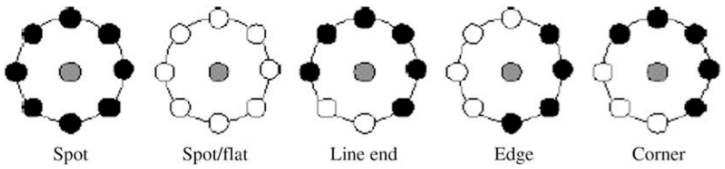


Figure 3.16: An Example of feature type coded by LBP and black circles present ones and white circle represent zeros, image from [14]

Define the operator as $LBP_{P\ R}$. Define texture T in local neighbourhood as the joint distribution

[1]LBP Example From `http://docs.opencv.org/modules/contrib/doc/facerec/facerec_tutorial.html#id22`

of gray level of P image pixels, R is the spatial resolution to the central pixel.

$$T = t(g_c, g_0, ...g_{P-1}) \tag{3.11}$$

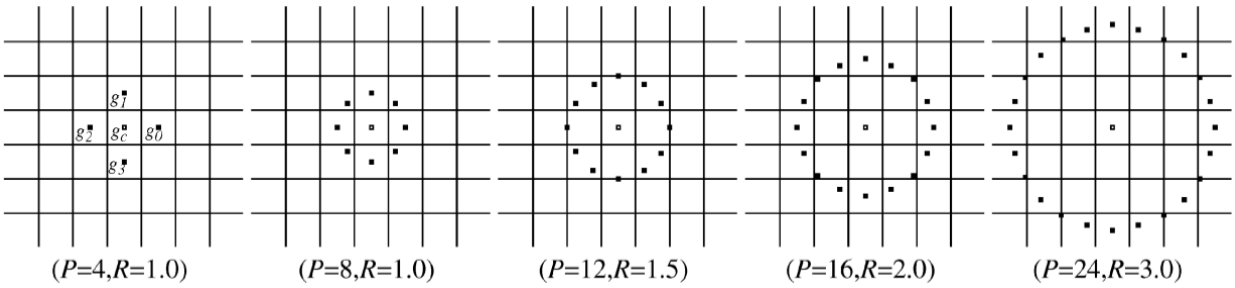


Figure 3.17: An example for multiscale circular pattern, image from [7]

Ojala et al proposed multi-scale LBP in [7], which could be used for arbitrary circular neighbourhoods and multiple scales; extend operator, uniform patterns, which can help reduce the feature vector. Uniform Local Binary Pattern is defined as patterns with at most two contiguous regions. The uniform pattern is as shown in figure 3.18. Uniform Patter have significant effect on dimension reduction as the 256(2^8) elements can be reduced to 59 elements, with 7*8 (uniform pattern from 1-7 in figure 3.18, as there are 8 directions) + 2 (uniform pattern 0 and 8)+ 1 (represent non-uniform patterns). In addition, most natural images are uniform LBP and uniform pattern is more robust than non-uniform pattern that it perform better result in many applications.

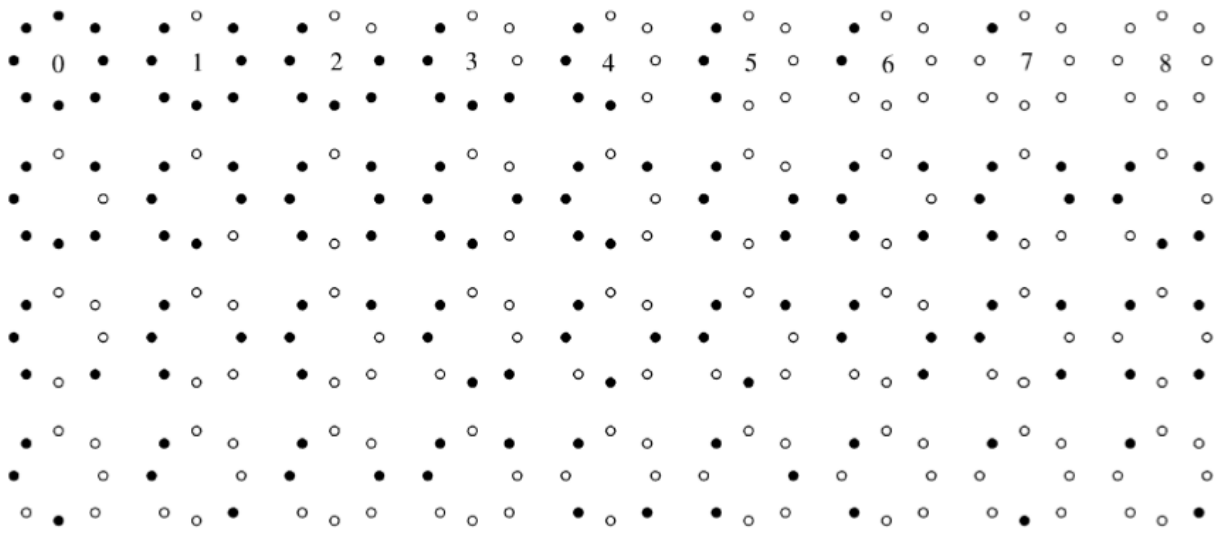


Figure 3.18: The first row are nine uniform patterns, Black represent 1, white represent 2, image from [7]

3.6 Postprocessing

As it is known large margin classifiers are sensitive to the way features are scaled, it's better to normalize either the data or the kernel function [3]. Feature of an image is represented by a vector, the number in the vector would influence the weight of feature in this dimension. In this experiment each dimension is treated equally, the number in the vector is scaled to$[0, 1]$.

Normalization It is proven that the performance of SVM is usually better if the data is normalized. There are two ways of applying normalization, standardizing the input features or normalizing the kernel function. In the training part, the data was standardising by subtracting its mean and divide by its standard deviation.

Scaling The range of appearance feature vector and shape feature vector is different. I would like to treat then as the same. So I scale all the vector into the range of $[0, 1]$, by subtract the minimum and divide by the maximum number of each dimension.

Chapter 4

Experiment and Results

In this chapter, section 4.1 describe component of database used in experiment, and section 4.2 introduce methodologies of classification and evaluation. Section 4.3 presents the result and give explanation of analysis from different aspects.

4.1 Data

There are 450 videos in two formats, avi and flv, 89 of them are flv videos, 361 of them are avi videos. The duration of videos are diverse from seconds to dozens of seconds. Videos were recorded using web-cam of different PCs. Video background are varied, as videos were recorded in places chosen by subjects. People were free to do anything while recording the vide. For example, a male disappear from the camera for half of the video sequence while recording. As a result, there were no faces in most of frames of that video. The Intraface tracker [16] is used to track those videos. It is able to track 439 videos. 1 video is tracked, but the tracking result does not match the label. 10 videos are tracked, but unable to identify the face in the video. The face in those untracked video can be clear identify by visual. One possible may be the resolution of the image. Maximum untracked video frame is $240 * 960$ pixels. Minimum tracked video frame size is $240 * 960$, the same as the maximum size of untracked video frame. It is reasonable to say tracker [16] may not be good at tracking low resolution videos. The observation mentioned in comparing tracker [16] and tracker [2] also support this hypothesis. The maximum frame tracked is $600 * 2400$. For each video there is a label file indicate the label of each frame, it contains the information of label of each frame and the millisecond of that frame in the video. frame number of each video is quite different, from around 200 to more than 900. The frame rate is around 30 fps. Table4.1 shows the tracking result of tracker [16].

There are six labels for each frame, normal face, eating, talking, looking away, occluded, other

Label / Title	Normal Face	Eating	Talking	Looking Away	Occluded	Other Problem
Total	35361	10409	5623	7730	21394	8422
Tracked	33776	9460	5196	5405	19014	3884
Rate	0.9552	0.9088	0.9241	0.6992	0.8888	0.4612

Table 4.1: Frame tracking result by tracker [16]

problem. A frame labelled as eating belong to an image sequence of eating. Label normal face, eating, talking and looking away are disjoint, but one frame can be labelled as one of them and occluded or other problem. It is very hard to track a face that facing camera with a big angle, so the tracking rate is very small for a face looking away. Only frame with label normal face, eating, talking in the experiment. Not all three labels are included in all videos, most videos miss one or two or even all three labels.

The three class: Normal Face, Eating, Talking are used for training and classification. The statistical data are show in table 4.2 and 4.3. As it is easy to find that the data is extremely imbalanced, there are a large amount of normal face.

	Normal Face	Eating	Talking
Tracked Frame Number	33776	9460	5196
Percentage of this class	0.6974	0.1953	0.1073
Percentage of not this class	0.3026	0.8047	0.8927

Table 4.2: Extracted Frames of Normal Face, Eating, Talking

	Normal Face	Eating	Talking
Tracked Sequence Number	871	114	207
Percentage of this class	0.7307	0.0956	0.1737
Percentage of not this class	0.2693	0.9044	0.8263

Table 4.3: Extracted Sequences of Normal Face, Eating, Talking

4.1.1 Feature

Each face is aligned with 49 facial feature points as shown in figure 3.4. As the tracker doesn't provide face bound point, so it's impossible to include jaw and cheek in ROI. Only the mouth region are used as Region of Interest. Then Local Binary Pattern feature is extracted from ROI. The size

appearance feature vector are different if image of ROI is divided into different number of blocks. 1 block and $1 * 3$ blocks are tried on dividing the image, the size of appearance feature vector is 95 and 177. Both 1-block appearance feature and 3-block appearance feature are experimented, in order to compare the influence of using different block features. As there are 49 shape feature points, so the size of shape feature points is 98.

4.2 Methodology

In the classification part, Support Vector Machine is used for training and classification. Two different type of appearance feature vectors are experimented, their dimensionality is 95 and 177, to see whether with more detailed appearance feature vector would be better for classification. The experiment focusing on finding answers to two question, would divide the image into more blocks while using LBP to extract features would influence the classification result, whether apply normalisation to each video would improve the classification result. In order to answer the first question, two group of features are examined. Both of them are extracted using blocked uniform pattern. However, one divides the image into 3 blocks, the other treats the image as one block. In order to find the answer to the second question, two different process are applied in normalising the features vector, one normalises both appearance feature and shape feature by each video, the other does not. One thing need mentions is that after put all feature vector together, feature vector of all groups are normalised. There are three types of feature vectors: shape feature vector, appearance feature vector and appearance+shape feature vector. As extracting feature using blocked uniform pattern only affect appearance feature vector, so in total, there are 10 groups of experiments.

SVM are firstly tested with linear kernel function and non-linear kernel, the Gaussian Kernel shows better result. Gaussian and polynomial kernels often leads to over-fitting in high dimensional database, while linear kernel is easier to tune because the only parameter that affects performance is the soft-margin constant [3]. The best result is using Gaussian Kernel, so Gaussian Kernel is used for classification. The most important parameters for Gaussian Kernel is penalty parameter c and γ in equation 4.1. Find the proper parameter could significantly increase classification result.

$$K(x, x') = e^{-\gamma||x-x'||^2} \tag{4.1}$$

4.2.1 Dealing with imbalanced data

We tried several actions to reduce the influence of imbalanced problem.

- Use 10-fold cross validation to all the classification result for all the entities.

- According to the number of training entities of each class, tune the training weight of each class.
- Or for each training, random select approximately the same number of entities in each class, usually .

4.2.2 libSVM

libSVM uses 'one-against-one' approach for multi-class classification [4]. If there are n different classes, it will generate $n(n-1)/2$ classifier, each classifier is trained with two classes. It also use a voting strategy: each entity is test with each classifier, the entity belong to the class with most number of votes.

4.2.3 Parameter Optimisation for SVM

A general way to find parameter c and γ is using cross-validation and grid-search. In n-fold cross-validation, first equally divide the data into n fold, leave out one fold of data as testing data and use other $n-1$ fold of data to train the classifier. Thus, all the data is predicted once and the cross-validation accuracy is the percentage of data are correctly classified.

Grid-search is try various pairs of c and γ and choose the one with the best cross-validation accuracy. Grid search approach is very simply and the computational time is no more than advanced method. To shorten the time of grid search, it is better to search with a coarse grid and then proceed with a more specific search in the identified grid.

4.2.4 Evaluation

In order to compare the different classification result, precision rate, recall rate and F measure to evaluate classification result of each group of data. For each class could form a table of 2x2 and 4 result, true position(TP), true negative(TN), false positive(FP), false negative(FN) as shown in table.

		Predicted Class	
		Class	Other
Actual Class	Class	TP	FN
	Other	FP	TN

Table 4.4: Confusion Matrix for two class

Recall rate is the percentage of actual entities that are correctly Predicted Positive [12]. Precision rate is the percentage of Predicted Positive entities that are correctly real positives [12]. TP represents the number of positives are correctly classified. FN represents the number of negatives are false classified. FP represents the number of positives are false classified. TN represents the number of negatives that are correctly classified. F measure evaluates both recall rate and precision rate. In this experiment, F1 measure are used for evaluation the result. The best score for F1 measure is 1 and the worst is 0, it can be interpreted as average weighted recall rate and precision rate.

$$\begin{aligned} recall &= \frac{TP}{TP+FN} \times 100\% \\ precision &= \frac{TP}{TP+FP} \times 100\% \\ F_\alpha &= (1+\alpha)\frac{precision * recall}{\alpha * precision + recall} \end{aligned} \tag{4.2}$$

4.3 Experiments

In this chapter, a detail description of results and explanations of analysis are presented. Figure 4.1, 4.5, 4.9, are frame classification result of Class Eating, Talking, Normal Face, using feature vectors that are normalised by each video. Figure 4.2, 4.6, 4.10, are frame classification result of Class Eating, Talking, Normal Face, using feature vectors that are NOT normalised by each video. Figure 4.3, 4.7, 4.11, are sequence classification result of Class Eating, Talking, Normal Face, using feature vectors that are normalised by each video. Figure 4.4, 4.8, 4.12, are sequence classification result of Class Eating, Talking, Normal Face, using feature vectors that are NOT normalised by each video. Figure 4.9, 4.15 are frame and sequence F1 measure of all three class using feature vectors that are normalised by each video. Figure 4.14, 4.16 are frame and sequence F1 measure of all three class using feature vectors that are NOT normalised by each video.

Most figure contains result of 5 groups, 1_A means the appearance feature is extracted by using 1 block uniform pattern and it is using appearance vector for classification. 3_AS means the appearance feature is extracted by using 3 block uniform pattern and it is using appearance and shape vector for classification. The same for 1_AS and 3_A, as using pure shape feature vector, it is not marked as normalised or not normalised. The identification of each group of classification is show in table 4.5.

4.3.1 Results and Analysis

Figure 4.1 shows precision rate, recall rate and F1 score of using appearance feature vector that are normalised by each video. The largest recall rate and F1 score are obtained by using shape

Identification	1_A	1_AS	3_A	3_AS	S	S_1	1_A_1	1_AS_1	3_A_1	3_AS_1
Normalisation (True if By Video)	T	T	T	T	T	F	F	F	F	F
Appearance Feature (1 or 3 Block)	1	1	3	3	1/3	1/3	1	1	3	3
Feature A: Appearance S: Shape	A	A+S	A	A+S	S	S	A	A+S	A	A+S

Table 4.5: Experiments and Identification in result figure

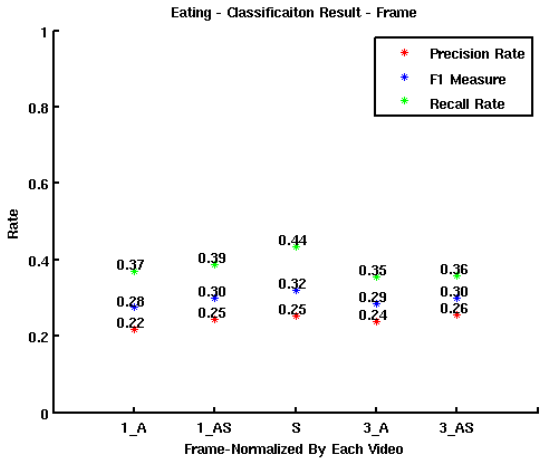


Figure 4.1: Class Eating - Classification Result of Frame - Frame normalised by each video

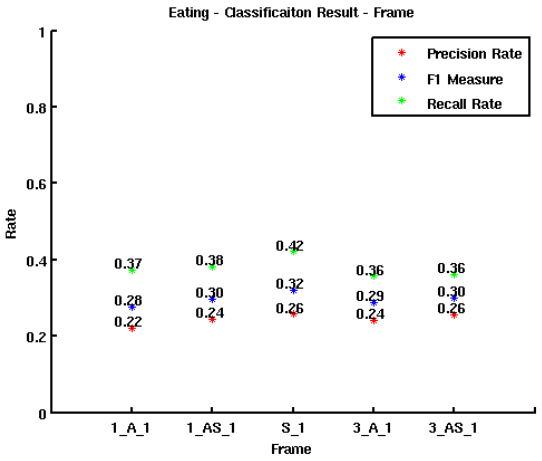


Figure 4.2: Class Eating - Classification Result of Frame - Frame NOT normalised by each video

feature vector in figure 4.1 and 4.2, which means shape feature is very good for classifying eating. The lowest F1 score is obtained by using 1-block appearance feature vector. For F1 score, 3_A is better than 1_A and 3_AS is better than 1_AS may because using 3-block LBP provides more information mouth region. The average F1 score of figure 4.1 is 0.3, almost the same figure 4.2, means normalise each feature vector of each video does not influence performance of classification eating. F1 score of both figure are in the range of 0.3±0.02, it is very close, it may means there is not much difference of using appearance feature or shape feature or both for classification. In addition, average score of classifying eating is 0.3, personally hypothesis is that distinguish eating from talking may be difficult.

Figure 4.3 and 4.2 shows the sequence classification result of eating. In figure 4.3 comparing to figure 4.1, there is a significant increase on recall rate of 1_AS, 1_S and 3_AS, and there is a significant drop of precision rate and F1 score, it may because as percentage of eating class number drop from approximate 0.2 of frames to 0.1 of sequences. The average of F1 score are almost the same for

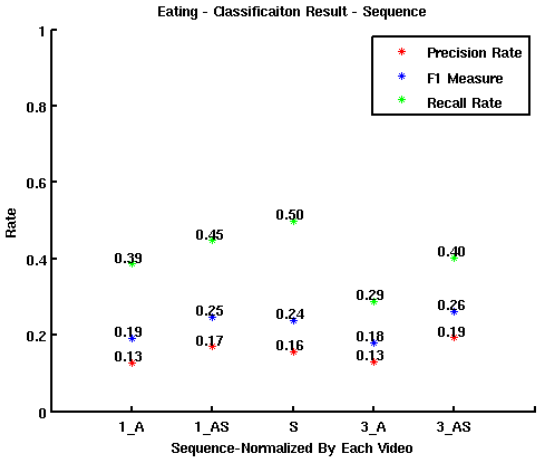


Figure 4.3: Class Eating - Classification Result of Sequence - Sequence normalised by each video

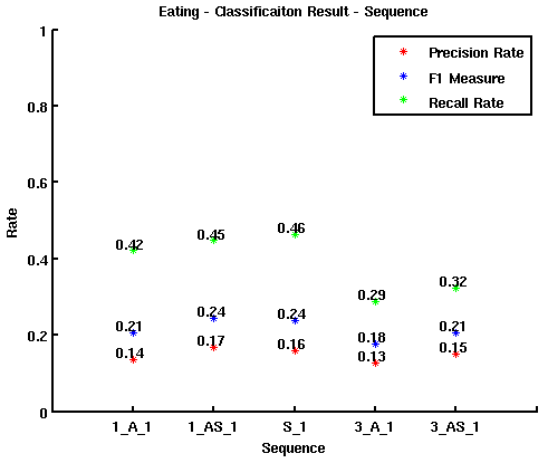


Figure 4.4: Class Eating - Classification Result of Sequence - Sequence NOT normalised by each video

figure 4.1 and 4.2 which the same situation as frame classification. However, the best F1 score is obtained by 3_AS if figure 4.1 and by S in figure 4.2. In figure 4.3 and 4.4 the average F1 scores are 0.224 and 0.216, standard deviation of F1 score in figure 4.1 is 0.0365 and standard deviation of F1 score in figure 4.2 is 0.0251, this means in classifying eating although not normalising data by video may lead to a little less in classification performance the result of using different feature vector could be more stable.

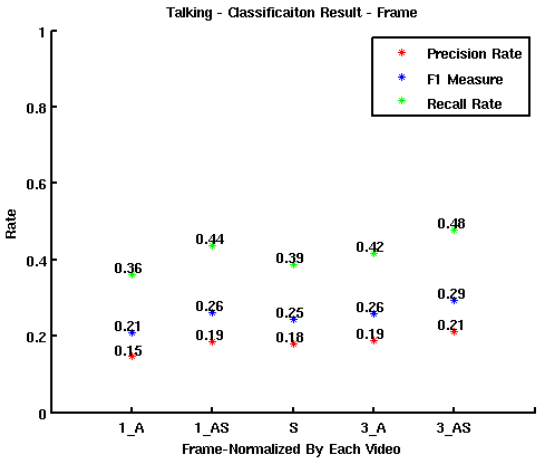


Figure 4.5: Class Talking - Classification Result of Frame - Frame normalised by each video

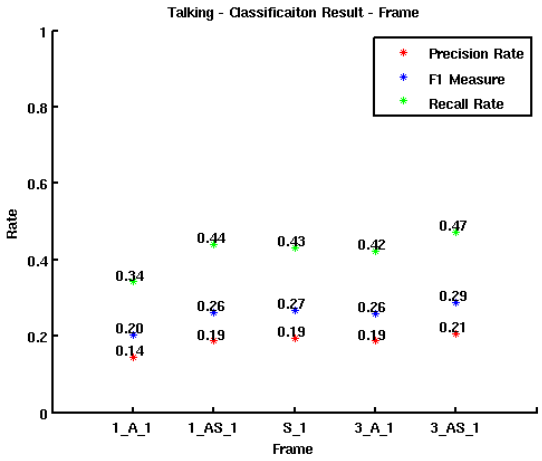


Figure 4.6: Class Talking - Classification Result of Frame - Frame NOT normalised by each video

In figure 4.5, the best F1 score is obtained by 3_AS, unlike figure 4.1, best F1 score is obtained by S. In addition, F1 score of 1_AS is better than S in figure 4.5, which is different as in figure 4.1 F1 score of S is better than 1_AS. What's more, F1 score of those classifications using both appearance and shape feature are larger than others, and it is the same result for figure 4.6. This

means that for talking frames, using both appearance and shape feature is better than using either one of them. Average of F1 score of talking is less than average F1 score of eating in both figure 4.5 and 4.6, it means the classification of eating is better than talking. Average recall rate of talking is higher than average recall rate of eating, this means proportion of false positive of talking is less than proportion of false positive of eating. Instead, average precision rate of talking is less than average precision rate of eating, it means proportion of false negative of talking is higher than proportion of false negative of eating. The standard deviation of F1 score in figure 4.5 and 4.6 are higher than the standard deviation of F1 score in figure 4.1 and 4.2, it means the influence of using different type of feature have more influence on talking than eating. F1 score of 3_A or 3_AS is higher or equal to 1_A or 1_AS in figure 4.1,4.2,4.5,4.6, they prove that using more information in appearance would increase the performance of classification. However, the high F1 score of S also means, using shape feature is better for classification than appearance feature.

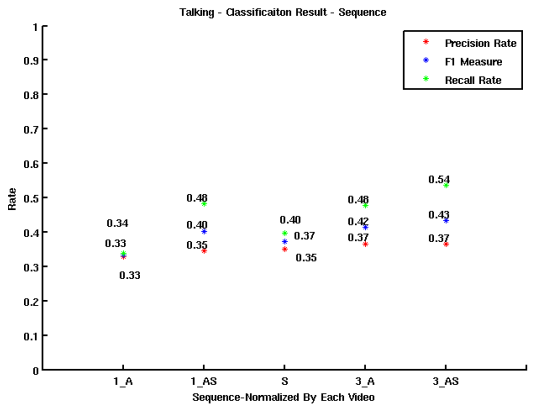


Figure 4.7: Class Talking - Classification Result of Sequence - Sequence normalised by each video

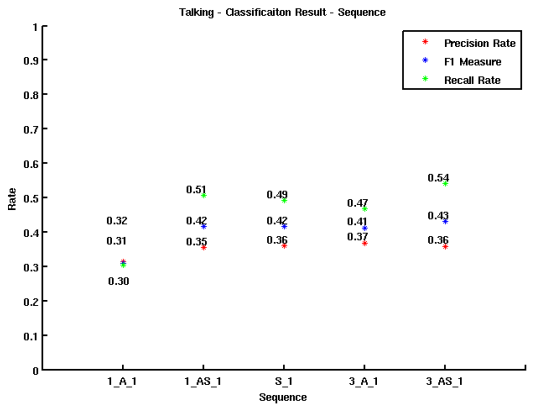


Figure 4.8: Class Talking - Classification Result of Sequence - Sequence NOT normalised by each video

Average F1 score in figure 4.7 is 0.39, which is much higher than 0.2240 in figure 4.3. Also, F1 score in figure 4.8 is 0.398, which is also better than 0.216 in figure 4.4. According to two above reasons, the classification of talking is better than eating. Compare figure 4.3 and 4.8, it easy to find that there is not much increase on average recall rate but average precision rate of talking is much better than talking. It means classifier is better recognise not taking than not eating. This is the main reason why classification of talking is better than eating. In classify talking, there is an obvious increase of using 3_A and there is a small increase on using 1_AS and 3_AS. From observation above, it seems appearance feature has a positive influence in classifier talking sequence. Compare figure 4.7 and 4.8, the F1 score in both figure is very similar, again it supports the hypothesis that there is not much influence whether normalise data by each video. However, F1 score of S_1 in figure 4.8 is better than F1 score of S in figure 4.7.

The best classification result is classifying normal face, it is much better than classify eating and

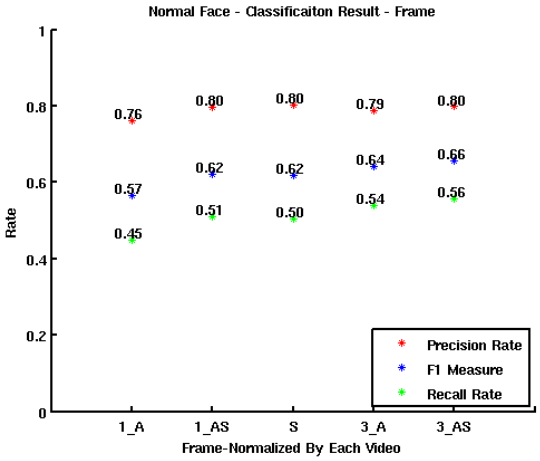


Figure 4.9: Class Normal Face - Classification Result of Frame - Frame normalised by each video

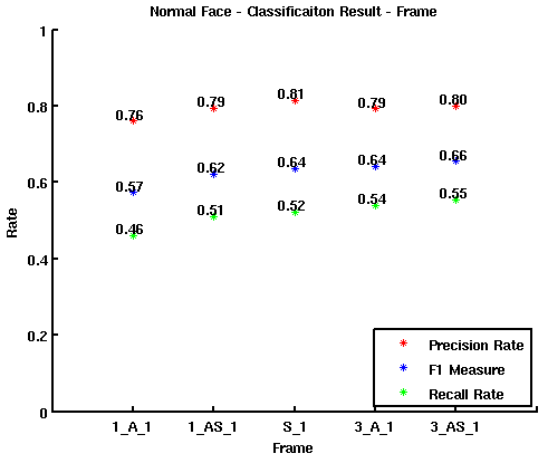


Figure 4.10: Class Normal Face - Classification Result of Frame - Frame NOT normalised by each video

talking. In both figure 4.9 and 4.10, the average recall rate is 0.51, precision rate is 0.79, and F1 score is 0.62. For normal face, the precision rate is higher than recall rate, which is different from classification result of eating and talking. From all frame classification result, figure 4.1, 4.2, 4.5, 4.6,4.9, 4.10, comparing F1 score, it easy to see that using shape and appearance feature is always better than using just appearance feature and using 3-block appearance feature is always better than 1-block appearance feature.

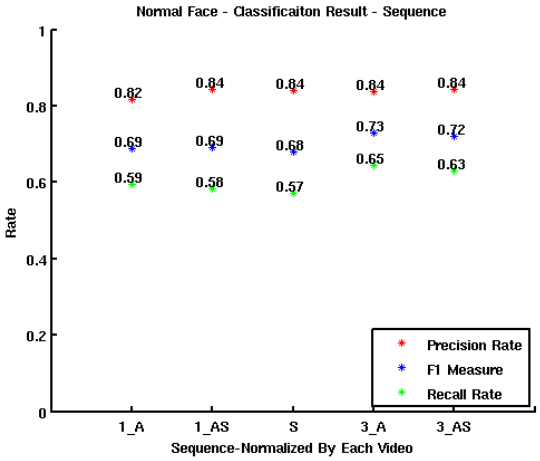


Figure 4.11: Class Normal Face - Classification Result of Sequence - Frame normalised by each video

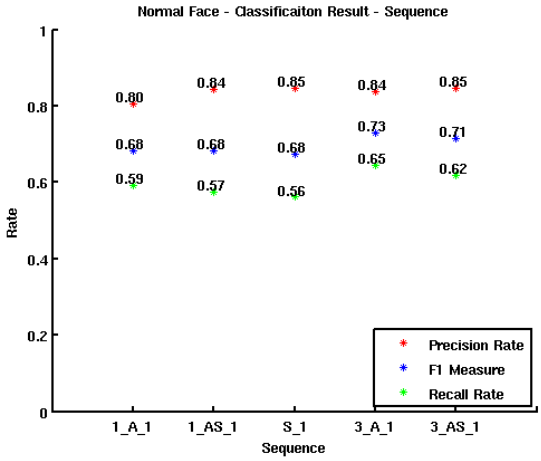


Figure 4.12: Class Normal Face - Classification Result of Sequence - Frame NOT normalised by each video

In figure 4.11 and figure 4.12, average precision rate is 0.84, F1 score is 0.7, and recall rate is 0.605. There is 0.1 increase in F1 score comparing to frame classification result of normal face. The performance of using S in figure 4.11 and 4.12 is the worst among using different features, which is different from classify eating and talking. The best F1 score is obtained by using 3-block

appearance feature and shape feature. It seems appearance feature is good for classifying normal face, but somehow had negative influence on classifying eating and talking. This may explain the reason why, using shape in classifying eating and talking performs relatively better than using appearance feature in classifying normal face.

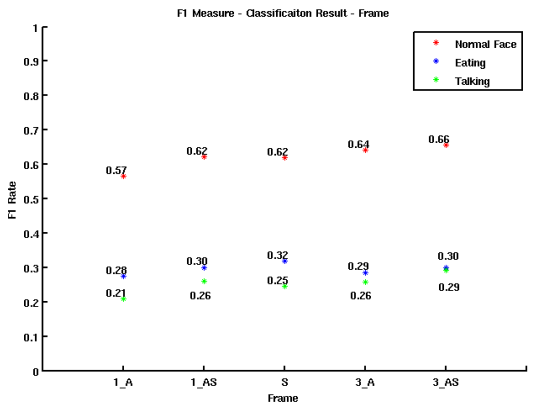


Figure 4.13: Three Class - Classification Result of frame - Frame normalised by each video

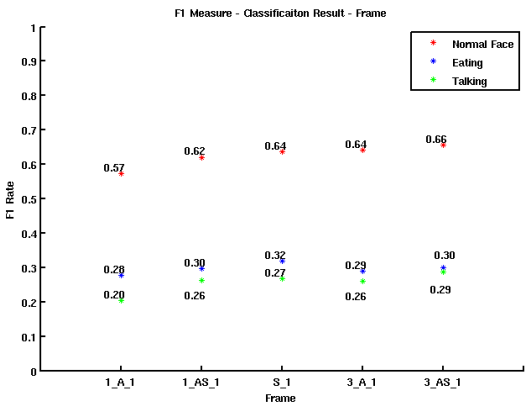


Figure 4.14: Three Class - Classification Result of frame - Frame NOT normalised by each video

The frame classification result of three classes are shown in figure 4.13 and figure 4.14. Classification result from best to worst is normal face, eating, talking, according to F1 score. The process of normalise feature vector by each video does not increase classification result. However, using 3-block uniform pattern appearance feature which provide more appearance information has positive influence on classification result. By calculating the total F1 score of 10-group features, the best classification result is gained by using 3-block uniform pattern feature with shape feature. The worst classification is gained by using just 1-block appearance feature.

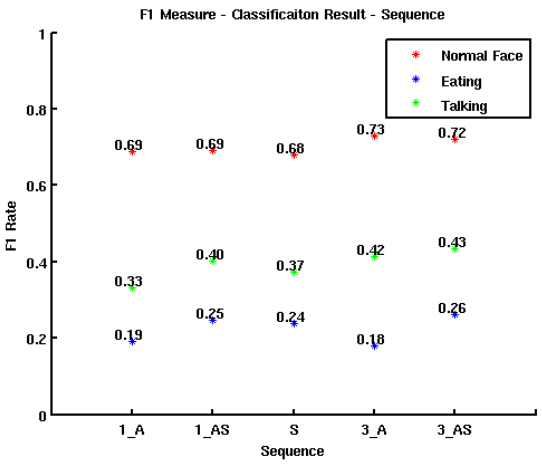


Figure 4.15: Three Class - Classification Result of sequence - Frame nomalised by each video

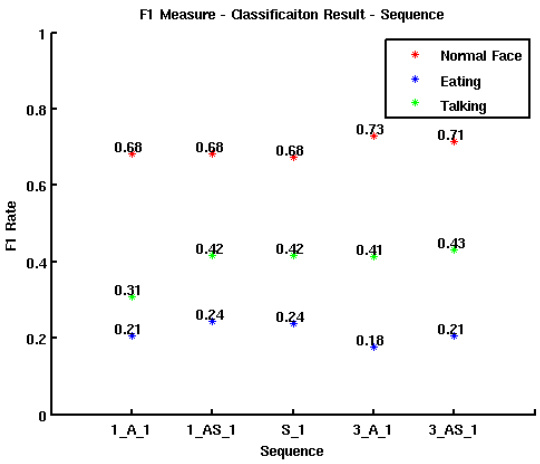


Figure 4.16: Three Class - Classification Result of sequence - Frame NOT nomalised by each video

For a sequence, the result of classify this sequence as a sequence of normal face, eating or talking is not directly classified by a classifier. It is calculated using majority voting from the classification result of frames. For example, the sequence is classified as a normal face if majority of frames are classified as normal face. Comparing result of classifying frame in figure 4.13 and sequence in figure 4.15, the F1 score of normal face has an increase of 0.1, eating has a decrease of 0.08, talking has an increase of 0.14. The result of comparing figure 4.14 and figure 4.16 is almost the same. According to average F1 score of 10-group experiments, the best F1 score is gained by using 3-block appearance feature with shape feature, which is the same as frame classification result.

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**18271 (39.06%)**	7880 (16.85%)	6791 (14.52%)
Eating	3156 (6.75%)	**3163 (6.76%)**	2425 (5.18%)
Talking	1387 (2.97%)	1304 (2.79%)	**2400 (5.13%)**

Table 4.6: Confusion Matrix-Frame-3_AS

Table 4.6 is an example of confusion matrix of frame using 3-block appearance feature with shape feature, more confusion matrix are in the appendix. From the table we can see, there are 48.78% of frames are classified as normal face, 26.40% are classified as Eating, 24.83% are classified as talking.

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**538 (45.13%)**	162 (13.59%)	171 (13.35%)
Eating	48 (4.03%)	**37 (3.1%)**	29 (2.43%)
Talking	50 (4.30%)	45 (3.78%)	**112 (9.40%)**

Table 4.7: Confusion Matrix-Sequence-3_AS

Table 4.7 is an example of confusion matrix of sequence using 3-block appearance feature with shape feature, more confusion matrix are in the appendix. The approximate percentage of sequences classified as normal face, eating, talking is 54%, 21%, 25%. Comparing to frames, an obvious

change is that there is increase in normal face and there is a decrease in eating. It seems, there are more sequence with low number of frames in normal face than eating. This hypothesis was proven and the evidence is in next subsection.

4.3.2 Data and Analysis

	Normal Face	Eating	Talking
1-10	322	32	90
11-20	122	17	45
21-30	117	18	25
30-40	91	8	16
41-50	35	3	7
50+	184	36	24

Table 4.8: The left column is frame number, first row is the class name. This table shows the number of sequences with frame number in the certain range.

In order to find out why the classification result is not good, and there is a big difference from classification result of frames and sequences. We statistical count the frame number of each sequence(shown in table 4.8). In table 4.8, the number of sequences with frame number between 1 to 10 is very large especially for normal face. Eating contains less low frame number sequence, this may be the reason there is a big difference in frame confusion matrix and sequence confusion matrix especially for normal face and eating. As the number of image sequence is so imbalance, majority voting strategy does not improve the result of sequence. Table B.1 in appendix gave a detail number of sequences with frame number from 1 to 10. It also shows the imbalance of testing data.

4.3.3 Examples of Wrong Classification

In this dataset, there is no smiling label and smiling is treated as normal face. Normal face do contain a large range of mouth movement, which make it easy to classified as normal face. In addition, when people are smiling the movement of mouth region is very obvious, it would be easy to classified to the wrong set. Such as the example shown in figure 4.17, the subject smiles and the action of mouth region could be classified into wrong class.

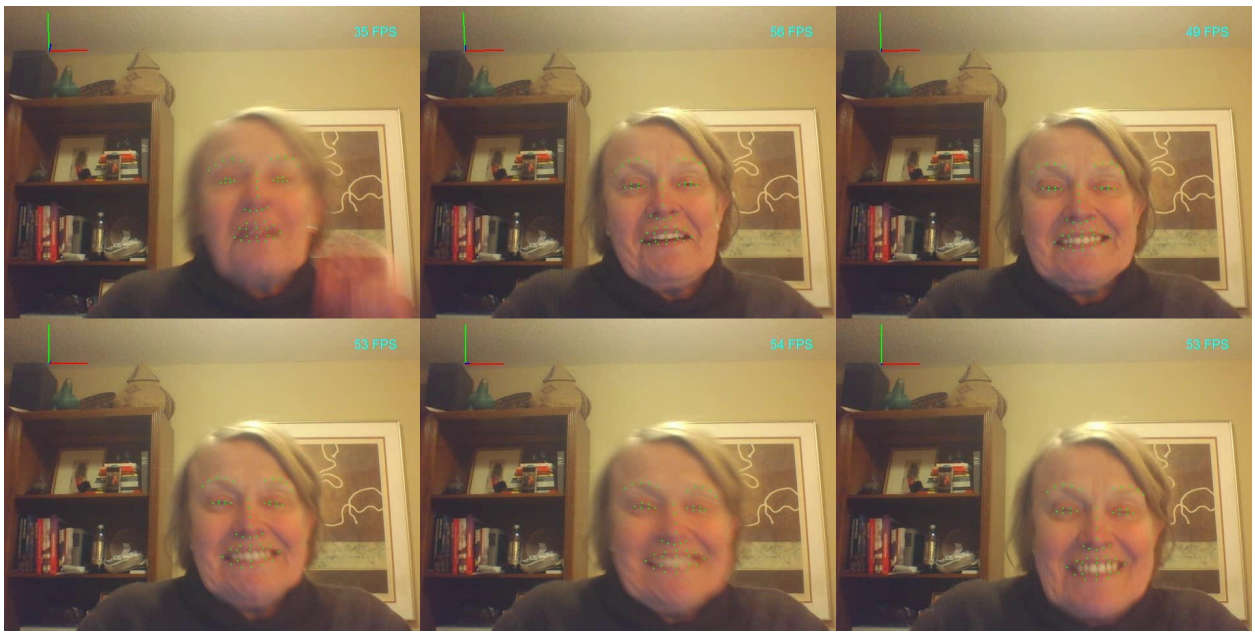


Figure 4.17: Wrong classification of normal face. Classification Result: [3 3 3; 3 3 1]

Figure 4.18: Wrong classification of normal face. Classification Result: [1 1 1; 3 1 2]

Figure 4.18 gave an example of wrong classification of eating. As you can see subject's mouth barely moves while chewing. And usually for a long video of eating sequence, it will contain many images of normal face as the subject would not keep eating and chewing without rest. The state of eating is always changing, it's more like a continuous state instead of a constant state of eating. SVM is good for modelling this type of state changing. Hidden Markov Model or long-term memory neural network may be better in modelling state changing. However, as database does not contain small sequence which just contain one complete eating. The time for doing this project is limited, it is impossible to label the videos by hands.

Figure 4.19: Wrong classification of Talking. Classification Result: [1 3 3; 1 1 1]

Figure 4.19 is a complete sequence of talking. Just by observing the image, it is hard to tell that the subject is talking. Four of the frames are classified as normal face, and these images looked very similar as normal face.

Conclusion Video set used in this project is recorded in uncontrolled environment, the condition of each video is different. The lighting condition, head moving, resolution of frames and the proportion of face inside the frame are very different. The number of each sequence is very different, it contains too many short sequence which should be filtered while using the data. In conclusion, more process should be done for filtering the sequences for classification step. Normal face frame may be not suitable for detection in this image as normal face contain many actions such as smiling, this would increase the difficulty of classifying normal face from other facial expressions. The model may be not good for expressing eating and talking.

Chapter 5

Conclusion and Future Work

Basically, all the procedure of detecting eating and talking from video recorded by webcam are finished. The difficulty of processing data is that the data usually is very diverse and there is no standard procedure, for different situation, different methodologies are used. For example, as there are many head movement in the videos, removing head pose from feature data seems to be very important. Data is vital to machine learning, basically data and model would directly decide the results. In this project, the importance of data and model are never should be underestimated. The result of classification is very unsatisfying. There are several reasons that would lead to this result. The process of remove head pose from shape feature vector, the transformation is not perfect, it will lose and change some information of shape feature. Secondly, the videos should be more carefully filtered, as videos are very diverse. Resolution, recording environment, lighting condition are different for each video. The most important and difficult problem may be remove head pose from face image and transform appearance image and facial feature points to frontal face. Thirdly, Support Vector Machine(SVM) may not very good at for this project. Since eating and talking contain state changing, using each frame as entity may not be a good idea. However, the number of frames in each videos is very different. A sequence may contains many section of eating, the video should be manually segmented if use SVM to train by sequences. In addition, frame number of each sequence is different, basically it is impossible to try to train an SVM using sequences.

As the time is restricted, we just carried on the experiment using SVM. There are much more selection for classification model, such as Hidden Markov Model, Long-short term memory neural network. There are many more other method could be tried, such as tracker for facial feature tracking, methodology of de-composite facial feature point and remove head pose, methodology of transform appearance image and facial feature points, extract different features. What's more, the video data could be categorised better and more filter and organisation could be done.

Bibliography

[1] Petar S Aleksic and Aggelos K Katsaggelos. Audio-visual biometrics. *Proceedings of the IEEE*, 94(11):2025–2044, 2006.

[2] Akshay Asthana, Stefanos Zafeiriou, Shiyang Cheng, and Maja Pantic. Robust discriminative response map fitting with constrained local models. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3444–3451. IEEE, 2013.

[3] Asa Ben-Hur and Jason Weston. A user's guide to support vector machines. In *Data mining techniques for the life sciences*, pages 223–239. Springer, 2010.

[4] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[5] Topi Mäenpää and Matti Pietikäinen. Texture analysis with local binary patterns. *Handbook of Pattern Recognition and Computer Vision*, 3:197–216, 2005.

[6] Iain Matthews and Simon Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60(2):135–164, 2004.

[7] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):971–987, 2002.

[8] Maja Pantic and Marian Stewart Bartlett. Machine analysis of facial expressions. 2007.

[9] Stavros Petridis and Maja Pantic. Audiovisual discrimination between speech and laughter: Why and when visual information might help. *Multimedia, IEEE Transactions on*, 13(2):216–234, 2011.

[10] Stavros Petridis, Maja Pantic, and Jeffrey F Cohn. Prediction-based classification for audiovisual discrimination between laughter and speech. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 619–626. IEEE, 2011.

[11] Gerasimos Potamianos, Chalapathy Neti, Guillaume Gravier, Ashutosh Garg, and Andrew W Senior. Recent advances in the automatic recognition of audiovisual speech. *Proceedings of the IEEE*, 91(9):1306–1326, 2003.

[12] David Martin Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. 2011.

[13] Jason M Saragih, Simon Lucey, and Jeffrey F Cohn. Deformable model fitting by regularized landmark mean-shift. *International Journal of Computer Vision*, 91(2):200–215, 2011.

[14] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803–816, 2009.

[15] Mikkel Bille Stegmann. Texture model formulation, 2000.

[16] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 532–539. IEEE, 2013.

Appendix A

Classification Result

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**14833**	9872	8237
Eating	3020	**3229**	2495
Talking	1618	1631	**1842**

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**518**	226	127
Eating	54	**44**	16
Talking	62	75	**70**

Table A.1: Classification Confusion Matrix: 1_A, left side is frame, right side is sequence

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**15161**	9797	7984
Eating	3129	**3256**	2359
Talking	1643	1704	**1744**

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**518**	230	125
Eating	54	**48**	12
Talking	71	73	**63**

Table A.2: Classification Confusion Matrix: 1_A_1, left side is frame, right side is sequence

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**16783**	8956	7203
Eating	2855	**3385**	2504
Talking	1413	1457	**2221**

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**509**	194	168
Eating	42	**51**	21
Talking	52	55	**63**

Table A.3: Classification Confusion Matrix: 1_AS, left side is frame, right side is sequence

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**16779**	8940	7223
Eating	2914	**3338**	2492
Talking	1433	1417	**2241**

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**500**	202	169
Eating	41	**51**	22
Talking	52	50	**100**

Table A.4: Classification Confusion Matrix: 1_AS_1, left side is frame, right side is sequence

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**16634**	9499	6809
Eating	2745	**3805**	2194
Talking	1358	1761	**1972**

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**499**	239	133
Eating	38	**57**	19
Talking	57	68	**82**

Table A.5: Classification Confusion Matrix: S, left side is frame, right side is sequence

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**17225**	8957	6760
Eating	2707	**3683**	2354
Talking	1229	1669	**2193**

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**489**	225	157
Eating	37	**53**	24
Talking	51	54	**102**

Table A.6: Classification Confusion Matrix: S_1, left side is frame, right side is sequence

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**17774**	8359	6809
Eating	3273	**3102**	2369
Talking	1472	1497	**2122**

Actual \ Predict	Normal Face	Eating	Talking
Normal Face	**562**	161	149
Eating	58	**33**	23
Talking	50	58	**99**

Table A.7: Classification Confusion Matrix: 3_A, left side is frame, right side is sequence

Predict / Actual	Normal Face	Eating	Talking
Normal Face	**17750**	8283	6909
Eating	3202	**3122**	2420
Talking	1448	1493	**2150**

Predict / Actual	Normal Face	Eating	Talking
Normal Face	**562**	168	141
Eating	56	**33**	25
Talking	52	58	**97**

Table A.8: Classification Confusion Matrix: 3_A_1, left side is frame, right side is sequence

Predict / Actual	Normal Face	Eating	Talking
Normal Face	**18354**	7887	6701
Eating	3201	**3141**	2402
Talking	1432	1219	**2440**

Predict / Actual	Normal Face	Eating	Talking
Normal Face	**549**	151	171
Eating	46	**46**	22
Talking	55	41	**111**

Table A.9: Classification Confusion Matrix: 3_AS, left side is frame, right side is sequence

Predict / Actual	Normal Face	Eating	Talking
Normal Face	**18271**	7880	6791
Eating	3156	**3163**	2425
Talking	1387	1304	**2400**

Predict / Actual	Normal Face	Eating	Talking
Normal Face	**538**	162	171
Eating	48	**37**	29
Talking	50	45	**112**

Table A.10: Classification Confusion Matrix: 3_AS, left side is frame, right side is sequence

Appendix B

Data Statistics

	Normal Face	Eating	Talking
1	59	3	6
2	67	1	8
3	40	4	10
4	40	6	8
5	21	3	12
6	24	3	10
7	22	5	11
8	21	1	6
9	15	3	8
10	13	3	11
10+	549	82	117

Table B.1: The left column is frame number,the first row is the different classes. This table is the number of sequences contain certain number of frame.