CHAPTER 1
INTRODUCTION AND SCOPE OF THE WORK

1.1 INTRODUCTION

**Thesis area:** The research work has focused on segmentation techniques for the faces present in photos. Photos of a face in two orientations are taken for constructing a three-dimensional (3D) face model.

**Application:** Online e-greeting cards, Avatars, entertainment, advertising, facial surgery, and teleconferencing.

**Market scenario:** Cloud Computing and the Software-as-a-Service (SaaS) models have become familiar with web services. A*Star’s 3D Face Modeling technology is very familiar with web services. This system can be applied to the following surveillance scenarios: a) Prevention of gambling at casinos is detected, b) In correctional facilities to screen visitors, and c) In airports to identify aircrews or to detect criminals.

An intelligent photo sorting application could use face recognition technology to tag photos automatically and sort them. A user would tag a photo with a particular person once, and the application could automatically tag all other photos featuring that person.

Other areas where face recognition is useful include a) Document control (photos on passports or national ID cards), b) Computer security (access user verification), and c) Time and attendance (entry/exit verification).

The application of the research is supportive in 3D facial modeling. Microsoft implements two-dimensional (2D)-to-3D face reconstruction. The software finds 83 key feature points such as face contour points, eye centers and nose shape that characterize the face under normal illumination and neutral expression. The key feature points are used to compute the 3D face model. From the input image the face texture is extracted and mapped to the 3D face model.

**The impact of the existing technology:** The current technology used by Microsoft uses features obtained from one photo containing a face.

**Emerging opportunities:** Developed 3D films. With the emerging trend of internet technology, much software for phone calls was developed. This software provides smileys and other animations.

**Software:** Some of the software used for 3D modeling is SIMTech Active Tag and System, Dance! 3D, Digital Forensic Evidence and File Recovery System, Intelligent Gesture-based Information Kiosk.
**Problem statement:** Microsoft claims that still, the 3D facial modeling is not complete, and it requires lots of developments. Microsoft uses only one facial orientation whereas other software developing 3D modeling uses more than one orientation. The particular problem is a complete reconstruction of the actual 3D appearance of a face.

**Issue proposed:** Segmentation of the face and transferring the segmented profile into a standard template (“Ruth’s face” from Rutgers University Talking Head).

Face detection has become popular due to increase in communication and security. The various postures of the face, lighting conditions, false presentation of some one’s face for breach of safety provides a challenge for human face detection. Most of the face detection methods use databases consisting templates of features. In many face detection technologies, segmentation plays an important role, and more research is carried out to improve the accuracy of face detection.

Many factors influence the success of human face detection and segmentation. Those factors include complex color background, the condition of illumination, change of position and expression, rotation of the head, and the distance between camera and subject. The human face provides different appearances at different instances. Hence, exact identification requires additional efforts.

Development of more efficient methods for human-computer interaction (HCI) are developed which do not rely on traditional devices such as keyboards, mice, and displays. Many commercial applications were developed. Faces were located in the images. Detecting a face is difficult in a single image due to change in scale of the face, its position, its upright or rotated orientation, and its frontal or profile pose. The facial expression, the presence of other objects that occlude the face, and change in the brightness of the lighting conditions result in a change in the facial textures.

Segmentation helps in extracting facial features which leads to the development of 3D face model. Face segmentation is an essential step of face recognition system since most face classification techniques tend only to work with face images. Therefore, face segmentation has to extract exactly the face part of given large image. Segmentation output becomes inaccurate when there is a change in the facial textures due to unexpected conditions.

Three-dimensional facial modeling helps in the virtual presentation of 3D data on the computer screen and utilizes it for various applications like 1) human-machine interface, 2) advanced multimedia, 3) augmented reality, 4) immersive virtual reality, 5) simulation of human-behavior with virtual human and 6) medical application.
There are two techniques used for obtaining 3D human models: 1) based on the accuracy and precision shapes and 2) relies on the virtual reality applications.

There are several approaches to the reconstruction of either a face or a body from photographs. Creation of smooth 3D face from the 2D images have been done using rendering. Representation of the 3D face appearance using feature points extracted from 2D images is a difficult process.

This thesis research work has proposed the creation of one to one locations in the 3D face using the information obtained from the 2D images. The thesis focuses on extracting the 3D model of the face by using contextual clustering segmentation and estimating the location of points on the face by using artificial neural network (ANN) methods for projecting the features.

1.2 OBJECTIVES OF THE THESIS WORK

1. To extract characteristics of a face using contextual clustering segmentation method, and
2. To create a face model from the available features using artificial neural network method.

1.3 SCOPE OF THE PRESENT WORK

Development of systematic approach was adopted for getting snapshots of the face without any unique environment and existing facial databases are used. The proposed segmentation method and point estimation by artificial neural networks for 3D modeling is presented. The algorithms used in this research work are as follows:

a). Contextual clustering segmentation,

b). Estimation of points to be projected on the reference face using Echo state neural network, and

c). Estimation of points to be projected on the reference face using combined back propagation algorithm and radial basis function.

1.3.1 Schematic Flow of the 3D Face Modeling

Figure 1.1 present the sequence in which 3D facial modeling has to be carried out. Photos are taken in two orientations. First orientation should be in front of the face. The second orientation should be focusing on the cheek on the right hand of left-hand side of the face.
Take two images of face at 90 degree orientations

Identify the corresponding locations of features on the two orientations

Segment the images using contextual clustering

Use Ruth's face model to project the segmented points

**Fig.1.1 Flow chart for 3D reconstruction of face**

### 1.3.2 Implemented Architecture for 3D Facial Modeling

- **Input two orientations of a face**
- **Extract features using contextual clustering**
  - Identify and extract features: summed value, mean value, contextual value
  - **Segment by CC**
  - **Train BPA / RBF / ESNN using three features**
  - Store final weights obtained for BPA / RBF / ESNN obtained from training
  - **Testing: Segment facial images using BPA / RBF / ESNN by using final weights**
  - **Comparison of segmentation Performance of CC / BPA / RBF / ESNN**
  - **Projection of points of two segmented images onto Ruth's face**
  - **Comparison of 3D face modeling Performance of CC / BPA / RBF / ESNN**

**Fig.1.2 Architecture for 3D facial modeling**
Working Details of implemented Architecture (Figure 1.2)

**Step 1:** Images in two orientations of a face will be considered.

**Step 2:** Images are segmented to get the face and feature profiles which are extracted from the images.

**Step 3:** Image Segmentation is performed by using the features obtained from CC. Subsequently, ANN algorithms are trained by BPA/ RBF / ESNN final weights are stored in a database.

**Step 4:** Process the extracted features with final weights of the BPA/ RBF / ESNN, to get an output in the output layer of the BPA/ RBF/ ESNN as segmented images, during the testing process.

**Step 5:** The segmented points are projected on Ruth’s face to obtain the 3D model.

1.4 EXISTING METHODS FOR EXTRACTING THE FACIAL INFORMATION

Table 1.1 presents different methods Knowledge-based, Feature invariant, Template matching and Appearance-based methods for modeling of a face.

1) **Knowledge-based methods:** Uses human knowledge for describing features of a face.

2) **Feature invariant approaches:** Algorithms are used to find structural features to locate a face.

3) **Template matching methods:** Numerical values are generated to form a template, based on the output of the algorithm.

4) **Appearance-based methods:** Using the facial images for training an algorithm for further recognition purposes.

1.5 FACE DEFINITION AND ANIMATION IN MOVING PICTURES EXPERT GROUP (MPEG-4)

The Face and Body Animation Ad Hoc Group (FBA) defines in detail the parameters for both the definition and animation of human faces and bodies. Definition parameters allow a detailed definition of body/face shape, size, and texture. Animation parameters permit the definition of facial expressions and body postures. The animation parameters are precisely defined to allow an accurate implementation of any facial/body model. Discussion on facial definitions and animations based on a set of feature points located at morphological places are presented.
<table>
<thead>
<tr>
<th>Approach</th>
<th>Representative works</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge-based</td>
<td>Multi-resolution rules-based method</td>
</tr>
<tr>
<td>Feature invariant</td>
<td></td>
</tr>
<tr>
<td>Facial features</td>
<td>Grouping of edges</td>
</tr>
<tr>
<td>Texture</td>
<td>Space gray level dependence matrix (SGLD) of face pattern</td>
</tr>
<tr>
<td>Skin color</td>
<td>Mixture of Gaussian</td>
</tr>
<tr>
<td>Multiple features</td>
<td>Integration of skin color, size and shape</td>
</tr>
<tr>
<td>Template matching</td>
<td></td>
</tr>
<tr>
<td>Predefined face templates</td>
<td>Shape template</td>
</tr>
<tr>
<td>Deformable templates</td>
<td>Active shape model</td>
</tr>
<tr>
<td>Appearance-based method</td>
<td></td>
</tr>
<tr>
<td>Eigenface</td>
<td>Eigenvector decomposition and clustering</td>
</tr>
<tr>
<td>Distribution based</td>
<td>Gaussian distribution and multilayer perceptron</td>
</tr>
<tr>
<td>Neutral network</td>
<td>Ensemble of neural network and arbitration schemes</td>
</tr>
</tbody>
</table>
1.5.1 Facial Animation Parameter Set

The FAP are encoded for low-bandwidth transmission in broadcast (one-to-many) or dedicated interactive (point-to-point) communications. FAPs manipulate key feature control points on a mesh model of the face to produce animated visemes (visual counterpart of phonemes) for the mouth (lips, tongue, teeth), as well as animation of the head and facial features like the eyes or eyebrows. All the FAP parameters (Figure 1.3) involving translational movement are expressed regarding Facial Animation Parameter Units (FAPU). These units are defined to allow the interpretation of FAPs on any facial model in a consistent way, producing reasonable results regarding expression and speech pronunciation.

![Fig.1.3 Facial animation parameter units](image)

1.5.2 Facial Definition Parameter (FDP) Set

The FDPs are used to personalize the generic face model to a particular face. The FDPs are usually transmitted once per session, followed by a stream of compressed FAPs. The FDP fields are as follows:

- **FeaturePointsCoord** – specifies feature points for the calibration of the proprietary face.
- **TextureCoords** – specifies texture coordinates for the feature points.
- **TextureType** – contains a hint to the decoder on the type of texture image, to allow better interpolation of texture coordinates for the vertices that are not feature points.
- **FaceDefTables** - this field describes the behavior of FAPs for the face in the FaceSceneGraph.
FaceSceneGraph - this node can be used as a container of the texture image as explained above or the grouping node for the face model rendered in the compositor and, therefore, it has to contain the face model.

1.5.3 FDP Cloning From Two Orthogonal Photographs

Using MPEG-4 parameters modeling method used laser scanner range data as input instead of easy and a cheap device like a camera. Also, it assumes that the FDP points are provided. The orthogonal photos (frontal and side views) are used as inputs while reconstructing 3D facial models. Extracting FDPs on photos in a user-friendly interface initializes a face and deforming a generic model. FDPs contain features of lips, chin, tongue, eyelids, eyes, eyebrow, nose, cheek, ear, tongue, head rotation point, teeth and some of face and hair outlines as shown in Figure 1.4 (a).

1.5.4 Face Cloning

A preprocessing is done to make

1) A generic shape of the face,
2) Generic animation structure,
3) Definition of feature points and feature lines that show the connection between feature points,
4) Definition of control points in the generic model,
5) Coordinate relation calculation for spline is used for shape modification and
6) Expression database.

Whenever there is input for a new face data, the predefined feature points and lines are positioned correctly. One process normalization and a semi-automatic feature positioning starting from a few key features to guide automatic feature detection.

1.5.4.1 Feature Detection

Feature detection from 2D image data is one of the main steps. It means to catch the 2D or 3D positioning data of most visible points on a face such as an eyebrow, eye outlines, nose, and lips, the boundaries between hair and face, and chin lines. Some parts such as forehead and cheeks are not always easy to locate exactly on 2D photographs, and they are called non-feature points.

Photographing- Taking an orthogonal photograph is a very difficult task in practice. There are some instructions for photography which are described as follows:

1) The ideal weather and place: It is best if the weather is cloudy since the skin color appears natural and no shadow casts on the face. The photographs are taken indoors, the lights and place, are selected that best emulate the cloudy weather condition. In this case, at least, two lights are recommended to minimize the strong reflection of the skin from a single source.

2) The ideal position of the camera: If the camera shot is taken up too close to the subject, the resulting virtual human will have an overall, deformed shape. Used maximum zoom of the camera positioned to minimize deformations. If space is limited, the best of the given condition will be completed. The camera calibration step avoids that limits input equipment by specific camera parameters for the calibration calculation.

3) The best hairstyle and face arrangement: Tucked the hair behind the person's ears so that the ears are visible, and the forehead is visible. The person has to remove eyeglasses and project a neutral face (no smile). The person has to sit up straight with the eyes open and the mouths closed. The teeth should not be visible.

4) For the side view, the face is made straight to see the profile properly.
5) The size of an image is not a problem, but it should not be too small or too big. The recommended size is 480 x 640.

**Normalization** (Figure 1.5) – Preparation is done for 3D-generic model and 2D-feature point frames for normalization and feature detection. Normalization is used to bring image data into feature point’s space.

![Diagram of Normalization Process](image)

**Fig.1.5 Normalization brings the input images into feature point’s space**

- a) First, prepare two 2D-frames composed of feature points and feature lines with predefined relation for frontal and side views.
- b) Design the feature lines in the frames to an initial position for the piecewise affine transformation and snake method, which will be used later.
- c) To make the head size in the frontal view and the side view the same, measure in two views the lengths of a face.
- d) Then choose one point from each view to match them with the corresponding points in the prepared frame.
- e) Then use two transformations (scaling and translation) to bring each photograph to the feature frame coordinates, overlaying frames on photos. Figure 1.5 considers two pictures of different size and position.

**Structured Feature Detection** - This method helps either interactively or fully automatic for detecting feature points by using special background information, predefined threshold, or image segmentation. Many times it would be difficult to locate the boundaries. Parameters such as the threshold for edge strength are too sensitive depending on each facial image.

Providing a semi-automatic feature point extraction has two main steps. When considering few key features interactive positioning is used. To adjust remaining features, automatic method is used. Figure 1.6 describes the prominent key feature points. After
applying the piecewise affine mapping, bring other feature points in relatively close to face features. Snake model is used to get correspondence between points from photographs and points on a generic model, which has a fixed number. Above the conventional snake, some more functions called structure snake are added, which is useful to make correspondences between points on the frontal view and ones on the side.

![Fig.1.6 (a) Normalization (b) Feature points frames on the frontal and side views](image)

Features are detected in all the orientations of the images. Get \((x, y)\) from the frontal view and \((y, z)\) from the side view which provide some automatic adjustment to make \(y\) coordinates in the frontal and side views share the same or, at least, similar value.

![Fig.1.7 Piecewise affine mapping for curves](image)

**Piecewise Affine Mapping** - In this procedure, combine the free form deformation with affine mappings (Figure 1.7). Implementation of matrix manipulation and transformations is done in affine mapping. This helps in translating points chosen to transform points. An object profile considers the various control points.

**Shape reconstruction** - In this method, a structured feature is identified. A 3D modification using 3D control points makes the condition for perfect orthogonal image less
critical than in 2D modifications. A generic model using 3D features is modified. Spline deformation is used to change the shape of the generic model by taking 3D feature points as control points for deformation. The massive calculation to get coefficients of vertex movement related to control points is done only once with the generic model and the saved coefficients for vertices applied for each new shapes. Here, approximate results are obtained.

**Modification in 2D or 3D** - It has a certain set of 3D feature points. The question is how to modify a generic model, which has more than a thousand points to make an individualized smooth surface. First, choose modification in 2D and then combine two sets of 2D to make 3D or do modification in 3D directly. This is a critical issue for the data handling for error tolerance.

An algorithm is applied to detected 2D features to make into 3D features to decrease the dependency of orthogonality of photographs and then take modification in 3D. Since feature point is known from the structure of photographs. There are possibilities to handle them in a proper way by filtering using the structure of feature points.

**Asymmetric 3D features from two sets of the 2D feature** - The input is two views of a face. One frontal view and one side view are the inputs. In general, more information is present in frontal view. Two sets of feature points on frames have structure, which means every feature point has its name. Some points which are visible on images have position values on both frontal and side views while others have values only on the frontal view or on the side (back part of a head is not visible in the frontal view). From a given set of 2D points the problem is how to make \((x, y, z)\). Since the perfect orthogonal pair photographs are not easy to be realized, it may result in unexpected 3D shape if the average of \(y_s\) and \(y_f\) for y coordinate (subscripts s and f mean side and frontal view) is taken. The criterion is to combine two sets of 2D features into 3D features, say \((x, y_f)\) and \((z, y_s)\).

![Fig.1.8 Feature points names are organized in a special way to indicate if the feature belongs To _R_, _L_ or centerline to make the algorithm create 3D points](image-url)
Where
FV means the feature point has value \((x,y)\) on the frontal view.
SV means the feature point has value \((z, y)\) on the side view.
_R_ means the right side region indicated as _R_.
_L_ means the left side region indicated as _L_.
_C_ means neither _R_ nor _L_.

Figure 1.8 shows the name convention used.
1. _R_, FV, !SV : Use a predefined relation from a typical face to get z, for instance, the depth value z of an inner corner point of the eye is calculated from middle and outer corner points of the eye. FV, SV: points have \(x, y, z\), and take \(y\) for y value.
2. _R_, !FV, SV : Use a predefined relation from a typical face to get z, for instance, the x value, for instance, the x of neck point invisible is calculated with visible neck points of the frontal view.
3. _L_, FV, !SV, _L_ : Take z value from the corresponding point with _R_ that has the counter reflection part by a vertical line, take the depth value z from the reflection part., for example, the left corner of the left side eye is corresponding to the right corner of the right side eye.
4. _L_, !FV, !SV : Use a predefined relation from a typical face to get x and take y and z values from the corresponding point with _R_.
5. !FV, SV, _C_ : Take x value on the line between two features with FV and _C_, for example, take the feature on top of the hair and middle chin feature or simply 0. Here, x as 0 is initialized, which will be used in the texture coordinate fitting method later.
6. Take \(y_s\) for y value in ‘ear’ region, which is useful for ‘ear’ texture for texture mapping.

**Modifying the generic model in 3D** - One of the easiest modification methods of a head is to use a spherical or cylindrical projection of head and use Delaunay triangulation and Barycentric coordinates. The more sophisticated solution is to use nonlinear deformations as opposed to applied linear interpolation. It can give a smooth resulting surface. For a deformation, it uses 3D feature points as a set of control points. Then the deformation of a surface can be seen as an interpolation of the displacements of the control points. Before applying deformation method, use global transformations (translation, and scaling) to bring obtained 3D feature points to generic model’s 3D space by selecting eight feature points on a generic model and corresponding feature points on detected feature points. In general, use the upper-most, right-most, leftmost and rightmost points for the translation and scaling.
Transform the feature point which defines generic model to adapt to the features detected in such a way as follows:

1) The design of the control point set on the generic model,
2) Calculation of the weights of a set of the control points set for each point on a surface of the generic model,
3) Deformation of the control point set to the new positions (here to the detected feature positions), and
4) Deformation of the object to the new positions using the new position of control points and the weights.

![Diagram showing the relation between feature points and control points](image)

**Fig.1.9 The relation between feature points and control points**

**Automatic texture mapping (Figure 1.9)** By applying real images, it utilizes texture mapping to increase photo-realistic looking of virtual objects. For virtual faces, the texture can add grain to the skin, including the color details for the nose, and lips. Texture mapping needs a texture image and texture coordinates. Each point on a head needs one or more coordinates. For texture mapping, the input images are not appropriately used.

**Texture Image Generation**- If two photographs for the frontal and side views are taken simultaneously in an environment with two cameras setting in the proper positions to get orthogonal photographs, perform texture mapping just by finding appropriate texture coordinates in two separate images. However, the restriction is removed as much as possible to get input images, so that two photographs taken with only one camera and the object rotate used, which changes the light condition and luminance. Image generation involves the following two steps.

1) Geometrical deformation to resist non-perfect orthogonal photographs as inputs, and,
2) Smoothing of boundaries between different images to resist non-coherent luminance.
**Geometrical Image deformation** - The frontal view is maintained due to importance than the side view. The side view is deformed and stuck to the frontal view. Two subsets of feature points are defined (one for left part and the other for the right part) on the frontal view, intending to keep an original (high) resolution for a major part of the frontal view, which lies between two feature lines. There is a corresponding feature line on the side image.

Use a feature named by Middle_top_HAIR, front_outline_R_HAIR1, face_R_HAIR3, outer_R_EYEBROW, R_CHIN5, R_CHIN4, R_CHIN3, middle_NECK for the right side feature lines for image deformation. For the left part, use dual-name policy such as face_R_HAIR3 is dual of face_L_HAIR3.

![Fig.1.10 The side view is deformed to stick to the frontal view](image)

**Fig.1.11 Input images**

Piecewise linear deformation is used for the side image to transform the feature lines to the corresponding one on the frontal view as shown in Figure 1.10, and Figure 1.11.
1.6 ORGANIZATION OF THE THESIS

Chapter 2 presents a detailed review of the literature on facial modeling.

Chapter 3 presents face database.

Chapter 4 presents the implementation of proposed algorithms for segmentation of faces.

Chapter 5 presents the results and discussion of segmented images and 3D modeling.

Chapter 6 gives the conclusions of the thesis work.

1.7 SUMMARY

This chapter has presented an introduction to facial modeling and also highlights the objectives and scope of the thesis work. This chapter also describes the flow of the proposed algorithms for the 3D facial modeling. Chapter 2 presents literature review.