International Journal of Electronics Signals and Systems

Volume 1 | Issue 1 Article 6

January 2011

Facial Feature Extraction Using a 4D Stereo Camera System

Soumya Kanti Datta
Network & Security,Institut Eurecom,Sophia Antipolis, France, datta.soumyakanti@gmail.com

Philip Morrow Dr.
School of Computing & Information Engineering,University of Ulster,Coleraine, UK, pj.morrow@ulster.ac.uk

Bryan Scotney Prof.
School of Computing & Information Engineering,University of Ulster, Coleraine, UK,
bw.scotney@ulster.ac.uk

Facial Feature Extraction Using a 4D Stereo Camera System

Soumya Kanti Datta
Network & Security

Institut Eurecom
Sophia Antipolis, France
datta.soumyakanti@gmail.com

Dr. Philip Morrow
School of Computing
& Information Engineering
University of Ulster
Coleraine, UK
pj.morrow@ulster.ac.uk

Prof. Bryan Scotney
School of Computing
& Information Engineering
University of Ulster
Coleraine, UK
bw.scotney@ulster.ac.uk

***Abstract* — Facial feature recognition has received much attention among the researchers in computer vision. This paper presents a new approach for facial feature extraction. The work can be broadly classified into two stages, face acquisition and feature extraction. Face acquisition is done by a 4D stereo camera system from Dimensional Imaging and the data is available in 'obj' files generated by the camera system. The second stage illustrates extraction of important facial features. The algorithm developed for this purpose is inspired from the natural biological shape and structure of human face. The accuracy of identifying the facial points has been shown using simulation results. The algorithm is able to identify the tip of the nose, the point where nose meets the forehead, and near corners of both the eyes from the faces acquired by the camera system.**

Keywords- Facial feature extraction, obj file format.

I. INTRODUCTION

Research in digital image processing has gained lots of momentum in the past two decades. Extraction of facial features is one of the related fields of research. Facial feature extraction is the first step in many important multimedia applications like face recognition [1], [2], [5] emotion recognition [3], and very low rate video coding. Face recognition is a widely deployed technique now-a-days and is used in airport security systems, verification in banking systems, video conferencing etc. Therefore the importance of correctly and accurately recognizing the facial features is becoming more important. Generally speaking, facial feature extraction technique includes the detection of the tip of the nose, lips, eyes and other important features of the face in face images.

Existing literature clusters the present techniques into four broad groups where the extractions are based on geometric feature [4], [6], [7], [8], template, color segmentation and appearance based approaches. Geometric feature based extraction is carried out by using relative positions and sizes of the important components of the face such as eyes, nose, mouth etc. Valley detection filters and analysis of horizontal and vertical edge integral projections are such examples. Template based approaches focus on a template function and energy function. The best match of a template in a facial image corresponds to minimum energy. For example, deformable templates [9], [10], [11], [20] and genetic algorithms. Color segmentation [12], [13] based feature extraction uses skin color to isolate the face. Any non skin color region within the face region is viewed as a candidate for "eyes" and / or "mouth" etc. Finally the appearance based scheme [14], [15], [16], [17], [18] aims to find basis vectors to represent the face using linear transformation and statistical methods.

This paper presents a novel approach for facial feature extraction. The following sections illustrate the entire work. Section II describes the face acquisition process by briefly introducing the 4D camera system of Dimensional Imaging. This section also gives an understanding of the 'obj' file format as the notion of face, node and vertex are used throughout the paper. Section III provides a detailed description of the developed algorithm for facial feature exaction. Section IV justifies the algorithm by simulation results using MATLAB and shows the simulation result of the features extracted from the facial images. Finally the paper discusses some future directions of the work and concludes.

II. FACE ACQUISITION

This is the first step in the facial feature extraction and the better acquisition is, the better is the feature extraction. The following subsection gives a brief overview of the camera system used and software used to generate the obj file that contains the facial information of the person.

A. The 4D Camera System and the Dimensional Imaging Software

The Dimensional Imaging [22] system is a system that creates 3D and 4D surface images. The system includes 3 video cameras and a software package consisting of a number of modules namely diVideoCapture, DI3D, diView4, DI4Dtracker. The top and bottom video cameras are used to acquire images in grayscale mode and the middle camera captures color images of the subject. The

overall system also employs two desktop computers that are linked together and build the 4D models. The system incorporates time, the fourth dimension, by pictures taken of a person over time and hence it is a 4D system. For this work the camera system is used to capture the face of a person over time to build the 4D imaging model and then to generate obj files that carry information of the face of the subject. For better understanding, we provide a brief outline of the structure of obj file in the next subsections.

B. Obj File Format

An obj file is a powerful representation of the surface of a 3D object, composed of triangles and higher degree polygons. The position of the vertices, UV position of the texture co-ordinate vertices, normals and the faces that define each polygon are defined as a list of vertices and texture vertices in the obj file. The main advantages of the obj format can be listed as below:

- Contents are written in ASCII characters in most of the cases although a binary version is also available.
- No compression takes place while generating the obj file of an image.
- It represents the 3D geometry and color information is available from the textures.

The obj format is outlined as follows:

- **Comment line(s)**: Every obj comment line begins with a hash character (#). The characters that follow the hash are comments and are ignored. Generally the comment states something about the source or the software that is used to create the obj file. For example,
  # Produced by Dimensional Imaging OBJ exporter
- **Vertex**: Vertex command specifies a vertex by its three co-ordinates. For example, v 4.148257 87.417888 165.645966
- **Vertex Texture:** The vertex texture command specifies the UV (and sometimes W) mapping. These will be floating point values ranging between 0 and 1 which say how to map the texture when grouped with a vertex in the 'f' face command. For example, vt 0.183379 0.073270
- **Vertex Normal:** The vertex normal command specifies a normal vector, e.g. vn 0.822768 0.321408 -0.468774.
- **Face:** The face command specifies a polygon made from the indices of vertices that follows the command 'f'. 'f 1 24 73' means a face built from vertices having indices 1, 24 and 73 respectively. Among the many definitions available for face in obj format, this camera system deals with **face definition using vertex/texture/normal** where each vertex is accompanied with a texture, which describes how to map the texture at that point, and a normal specifying a normal at that point.

An obj file may require information about materials which are stored in an mtl file. If such is the case, the obj file must refer to the name of the mtl file by a command 'mtllib' in the file itself. For example,
#mtllib ./2009-10-08 S and D test_000_001.mtlg
In the mtl file also, a reference is there for the obj file which accesses the information stored in it as follows:
#newmtl 2009-06-24-model_002_007

III. FACIAL FEATURE EXTRACTION

After the obj file is obtained from the 4D camera system, it is read and the mesh structure is displayed using MATLAB. The mesh structure is initially displayed as shown in Fig 1. By clicking on the rotate 3D tab in MATLAB figure, the mesh structure of the face can be rotated to yield Fig 2 which gives the 3D representation. The algorithm developed in this work is inspired from the natural biological structure of the face. The algorithm first identifies the tip of the nose which in this case is the most vital feature to be extracted.

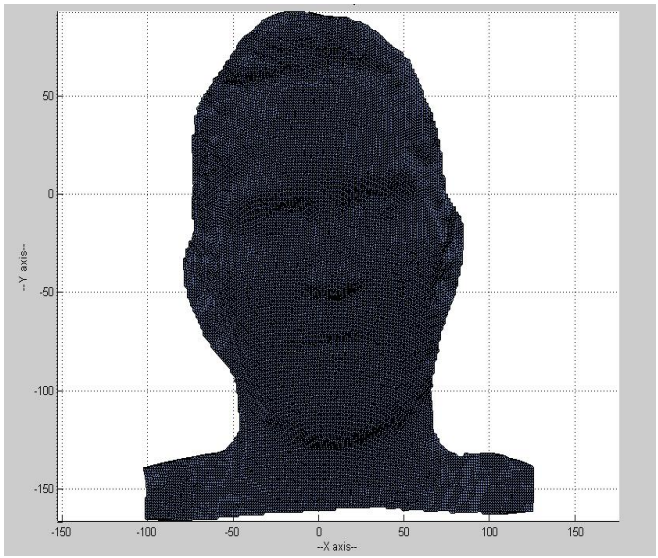

Figure 1. Mesh structure of the face as display by MATLAB.

A. Detection of 'Tip of the nose'

The correct extraction of the other features of the face depends on the accurate detection of the tip point of the nose which is pretty simple to identify when inspecting the mesh carefully. It is evident from Fig 2 that the top-most node of the nose i.e. the tip of the nose will have highest z-coordinate. The highest point is referred as the **peak point** throughout the work. The algorithm to detect the peak point is stated below:

1. Begin.
2. Read all the coordinates of nodes of a face from the corresponding obj file into an array.

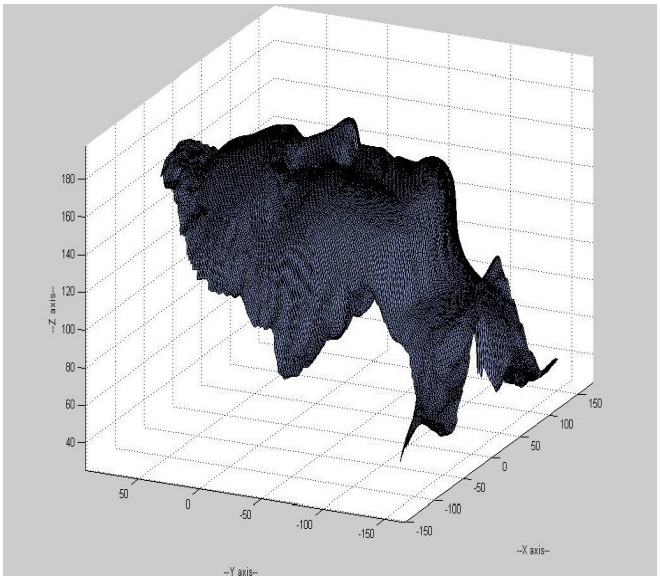

Figure 2. 3D view of the mesh structure generated by rotating Fig. 2 using the 'rotate 3D' tab in MATLAB figure.

3. Sort the z-coordinates to find out the highest value.
4. Store the coordinates (that contain the highest z-coordinate) of the **peak point** and also the index of the node and call it **peak index**.
5. End.

B. Detection of the point where nose meets the forehead

Once the peak point is determined, the nodes, or the faces, that go up in the direction of the forehead can be detected. When Fig 2 is inspected carefully again, it is noticed that from the peak point, when traversed to the direction of the forehead, the value of z-coordinates decrease gradually with the minimum at the point where the nose meets the forehead, then there is an increase in the z-coordinates. It should be also noted that, y-coordinates increase as the traversal is done from the peak point to the forehead. The following procedure has been developed to locate the point where the nose meets the forehead.

1. Begin.
2. Read the co-ordinates of the peak point and the peak index.
3. Locate the adjacent faces that have the peak point as one of the nodes building the faces.
4. Store the indices and the co-ordinates of the nodes that belong to the adjacent faces.
5. Repeat for all the new adjacent nodes
    a. Extract the y-coordinate of the new node
    b. If the y-coordinate is greater than that of peak point, store the index and co-ordinates of the nodes separately in arrays namely store_index_temp and store_node_temp.
    c. Else ignore the new node.
    d. End If.
    e. End Loop.
6. Calculate the distance between the x-coordinates of the nodes stored in store_index_temp and the peak point.
7. Sort the distances to get the node with the minimum distance.
8. The node with minimum distance becomes the new **peak point**. Store the coordinates of the new peak point and its index.
9. Repeat steps 2 to 9 while recording the z-coordinates until a gradual rise in z-coordinate is observed.
10. Plot the z-coordinates in y-axis and the nodes in x-axis.
11. Obtain the faces from the peak point that lead to the point at which node meets the forehead.
12. End

The above procedure gives rise to Fig 3 and Fig 4 in MATLAB. This procedure identifies the point where nose meets the forehead accurately. This point further leads to the near corners of the two eyes. In Fig. 4 and Fig. 5, the required point is the node with minimum value of z-coordinate.

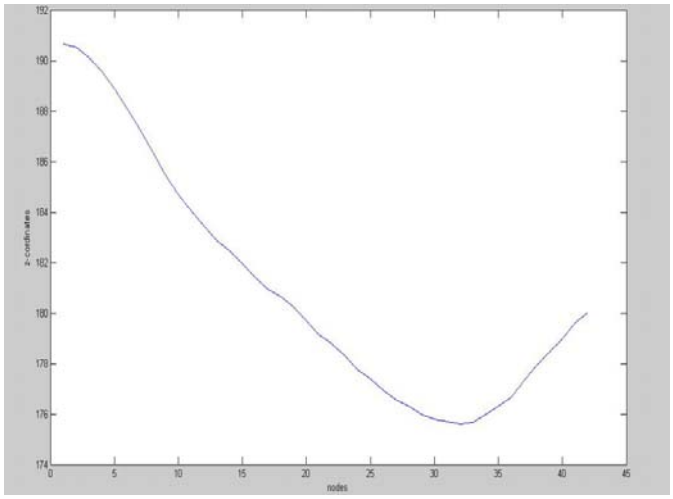

Figure 3. Plot of the z-coordinates of the nodes leading to the forehead from the peak point.

C. *Detection of near corners of the two eyes*

The detection of near corners of the eyes is now possible. The point identified in previous step is located in Fig. 3. When observed, it is revealed that from that point, if traversed towards the left where x becomes more negative, initially z-coordinates fall. But there is a slight rise when the corner of the left eye is found. As human face is bilaterally symmetric, the same procedure is followed to detect the corner of the right eye. The only difference is, in this case the traversal has to move in the positive direction of the x-axis as comprehended from Fig 2.

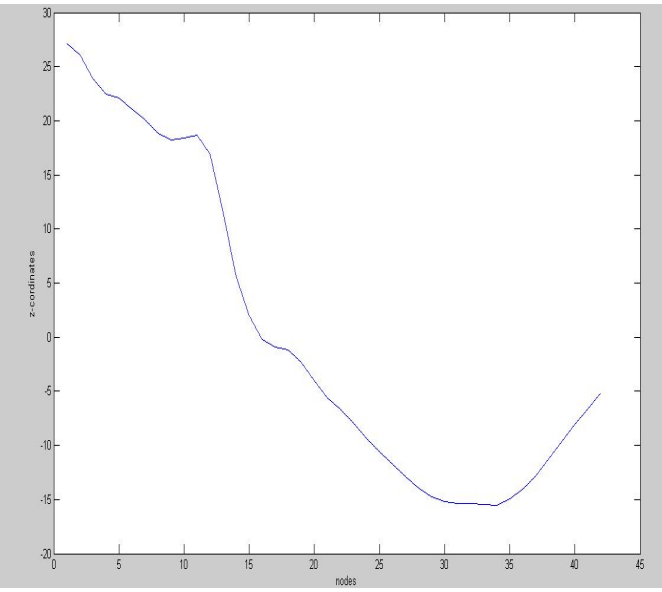

Figure 4. Another plot of the z-coordinates of the nodes leading to the forehead from the peak point.

The procedure to detect the near corners of the two eyes is as follows:

1. Begin.
2. Read the coordinates of the point where the nose meets the forehead.
3. Travel towards the negative x-axis and record the z-values.
4. Stop when a slow rise in the z-values is found.
5. Store z-value next to the minimum z-value.
6. The coordinates of the point in the previous step gives the corner of the left eye.
7. Travel towards the positive x-axis and record the z-values.
8. Stop at a slow rise found in z-values.
9. The coordinates of the point in the previous step gives the corner of the right eye.
10. End.

Thus the near corners of the two eyes are detected.

IV. RESULTS

The entire algorithm has been implemented in MATLAB. The simulation result for a particular facial image is given in Fig. 5. The tiny dots in Fig 5 point out the feature points extracted by the algorithm. The algorithm has been applied on several images captured by the 4D camera system and the algorithm is able to extract the features in every case.

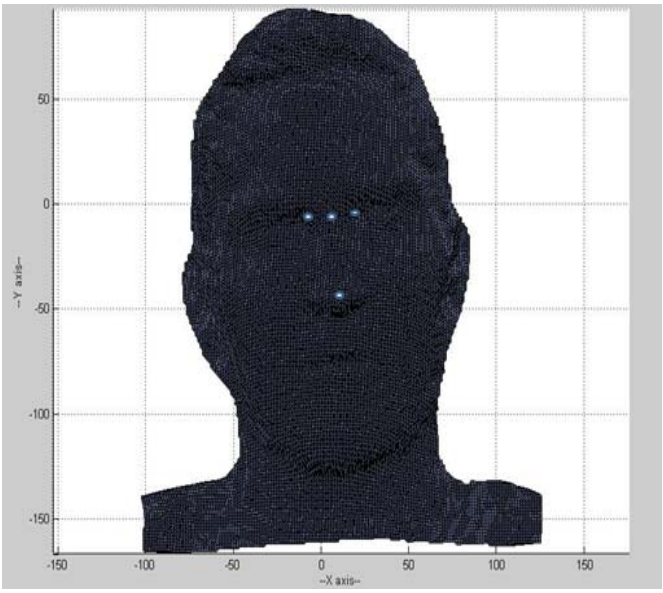

Figure 5. MATLAB simulation result of the developed algorithm.

V. FUTURE WORK

Future prospect of the work includes the identification of far corners of the eyes, the mouth and the chins. The facial feature extraction using the method stated above can be used in many applications. One novel application will be the use in medical imaging like medical diseases that affect the muscles of the chin. A system can be developed that will automatically locate the affected muscles in the face using feature extraction techniques and identify how the muscles are responding to treatment.

VI. DISCUSSION

The human face is a complex structure and the diversity of the features is well known. The proposed algorithm is simple and makes use of the natural structure of the face, and works in most of the cases. Simulation results show that the procedure is able to identify the features mentioned above correctly. Currently investigation is being done to extract other important features of the face like the extraction of the entire eyes, eyebrows, mouth etc. Once most of the facial features are identified, the developed algorithm can pave way for face recognition in a novel way. Emotion recognition [21] is also a very important research now-a-days. Humans can identify the emotion by looking at a face but recognizing emotion using computer vision heavily

depends on facial feature extraction. As the relative positions of eyebrows, chin, opening of lips helps in identifying emotion, correct extraction of facial features is of utmost importance. Therefore it can be concluded that this algorithm is quite useful and will find its way into many important applications.

ACKNOWLEDGMENT

The authors would like to thank Mr. Stuart Henry, Miss Deborah Rankins and Dr. Cathryn Peoples for their participation in this work to build the facial database and make the work successful.

REFERENCES

[1] S.Kar, S.Hiremath, D.G.Joshi, V.K.Chadda, A.Bajpai, "A Multi-Algorithmic Face Recognition System". International Conference on Advanced Computing and Communications, 2006, pp. 321 - 326.

[2] S.Jahanbim, Hyohoon Choi, R.Jahanbin, A.C.Bovik, "Automated facial feature detection and face recognition using Gabor features on range and portrait images". 15th IEEE International Conference on Image Processing, 2008, PP. 2768 - 2771.

[3] M.Ghosh, A.Chakrabarty, A.Konar, A.Nagar, "Prediction of the Interactive Dynamics of Stimulated Emotions: Chaos, Limit Cycles and Stability" Second UKSIM European Symposium on Computer Modeling and Simulation, 2008, pp. 105 – 110.

[4] T.Kawaguchi, D.Hidaka, and M.Rizon, "Detection of eyes from human faces by Hough transform and separability filter", IEEE International Conference on Image Processing, Vol 1, pp. 49 – 52, 2000.

[5] Viola and Michael Jones, "Robust real-time face detection", International Journal of Computer Vision, pp. 137-154, 2004.

[6] M. Jones and P. Viola, "Face Recognition Using Boosted Local Features", IEEE International Conference on Computer Vision, 2003.

[7] Shu Liao, Wei Fan, Albert C. S. Chung and Dit-Yan Yeung, "Facial Expression Recognition Using Advanced Local Binary Patterns, Tsallis Entropies And Global Appearance Features", IEEE International Conference on Image Processing, pp. 665-668, 2006.

[8] C. Liu and H. Wechsler, "Gabor Feature Based Classification Using the Enhanced Fisher Linear Discriminant Model for Face Recognition", IEEE Trans. Image Processing, vol. 11, no. 4, pp. 467- 476, 2002.

[9] Yuille, A.L.; Cohen, D.S.; Hallinan, P.W.. "Feature Extraction From Faces Using Deformable Templates", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp.104 – 109, 1989.

[10] Liang Zhang, "Estimation Of The Mouth Features Using Deformable Templates", IEEE International Conference on Image Processing, Volume 3, pp 328 – 33, Oct 1997.

[11] Paul Kuo, John Hannah, "An Improved Eye Feature Extraction Algorithm Based On Deformable Templates", IEEE International Conference on Image Processing, Volume 2, pp. 1206-1209, 2005.

[12] Son Lam Phung, Abdesselam Bouzerdoum, and Douglas Chai, "Skin Segmentation Using Color And Edge Information. International Conference on Signal Processing and Its Applications", Volume 1, pp. 525- 528 , July 2003.

[13] Teerayoot Sawangsri, Vorapoj Patanavijit, and Somchai Jitapunkul., "Face Segmentation Using Novel Skin-Color Map And Morphological Technique", Proceedings of World Academy of Science, Engineering and Technology volume 2 ISSN 1307-6884, jan 2005 .

[14] Thai Hoang Le, Tri Minh Nguyen, Hang Thuy Nguyen, "Proposal of a new method of feature extraction for face recognition", National Conference about Information Technology, DaLat city, 2006.

[15] M. Turk and A. Pentland (1991), "Face recognition using eigenfaces", IEEE Conference on Computer Vision and Pattern Recognition, pp. 586–591, 1991.

[16] Bruce A. Draper, Kyungim Baek, Marian Stewart Bartlett, J. Ross BeveRidge, "Recognizing Face with PCA and ICA", Computer Vision and Image Understanding 91, pp. 115–137, 2003

[17] P. Comon, "Independent component analysis—A new concept?", Signal Processing, vol. 36, pp. 287–314, 1994.

[18] M.S.Bartlett, J.R.Movellan, T.J.Sejnowski, "Face Recognition by Independent Component Analysis", IEEE Transactions on Neural Networks, Vol. 13, No. 6, Nov 2002.

[19] R.C. Gonzalez, R.E. Woods, *Digital Image Processing*, Addison-Wesley Pub. Co., Jan 2002.

[20] Z. Wang, F. K. Huangfu and J. W. Wan, "Human Face Feature Extraction Using Deformable Templates", *Journal of Computer Aided Design and Computer Graphics of China*, 2000, 12(5), pp 333-336.

[21] A.Chakraborty, A. Konar; U.K.Chakraborty, A.Chatterjee, "Emotion Recognition From Facial Expressions and Its Control Using Fuzzy Logic". IEEE Transactions on Systems, Man and Cybernetics, 2009, Vol 39, Issue 4, pp. 726 - 743.

[22] www.di3d.com