3D Face Identification and detection by using Adaboost Method

Er. Priyanka Kulkarni¹, Er.Kantilal P Rane², Er. Vijay Chaudhari³

¹PG Student, Department of E&TC, GF’s Godavari CoE, Jalgaon
²Asso. Prof., Department of E&TC, GF’s Godavari CoE, Jalgaon
³Assist. Prof., Department of E&TC, GF’s Godavari CoE, Jalgaon

ABSTRACT

The proposed system is developed for face detection which increases security in different areas viz. banks, military, industry, stadium etc. Face detection using adaboost algorithm highly improves the accuracy and reliability of the system. In certain methods used for face detection like eigen vector images are prepored before extraction of feature as they may be affected by light condition or change in facial expressions. The adaboost method overcomes all these drawbacks & removes the noisy regions in an image. This system detects mutual information and includes cross correlation. Experimental results show that the proposed system obtains the 3D view with more accuracy information.

Keywords: Face detection, adaboost algorithm, cross correlation, mutual information, position.

1. INTRODUCTION

Face detection is a very important concept in applications like face recognition, videosurveillance, human computer interface, face image database management, and querying image databases. There have been various approaches proposed for face detection viz. Template matching methods, Feature-based methods, Knowledge-based methods[1], and Machine learning methods. In template matching method depending upon features like scale, rotation the decision is made between the given template and the given image. Feature-based approaches extract salient features from the sensed and reference data sets and aim to find the transformation that minimizes the distance between corresponding features. The extracted features can be regions, edges, or interest points. Detect maximally stable extremal regions and set up the correspondence between pairs of images based on these stable regions. [3,4]. Knowledge-based methods[5] detects an isosceles triangle (for frontal view) or a right triangle (for side view). Machine learning methods use a lot of training samples to make the machine to be capable of judging face or non-face. Though machine learning method has gained tremendous success still its complexity and efficiency need to be improved. Among all these methods of face detection the area based methods is an important category. This paper proposes a system for 3D view information. The entire final image regions are obtained using adaboost algorithm. The mutual information and cross correlation is applied to detect the 3D view.

2. PREVIOUS WORK

Although there is a large amount of literature on face detection, certain papers are performed to verify the effectiveness of the face detection using colour images. Most of the commonly used databases for face detection, including the Carnegie Mellon University (CMU) database contain gray scale images only. Therefore, the systems has constructed the database for face detection from personal photo collections, these images contain multiple face images with variations in color, position, scale, orientation, and facial expression. Some sample images from Internet compose training set and testing set. The initial scheme integrates AdaBoost algorithm in skin color information to improve detection precision, meanwhile, to reduce computational cost. Results using the proposed method show that the new approach can detect face with high detection rate and low false acceptance rate and performance better than skin color detection and AdaBoost algorithm. But false alarms and misses are still present.

3. 3D ESTIMATION

3-D motion estimation has been studied for some time, but now, improved computation and memory resources make higher performance methods increasingly feasible. In 3-D registration, motion can occur along all three dimensions. Previous work on image registration can be broadly classified into feature-based and area-based methods. Feature-based approaches extract salient features from the sensed and reference data sets and aim to find the transformation that
minimizes the distance between corresponding features. The extracted features can be regions, edges, or interest points, detect maximally stable extremal regions and set up the correspondence between pairs of images based on these stable regions. Proposed a contour-based method, which uses region boundaries and other strong edges as matching primitives. Features from Accelerated Segment and perform descriptor matching to estimate the image transformation. In contrast to feature-based methods, area-based methods attempt to perform registration without extracting salient features. Common area-based approaches include cross correlation (CC) methods and mutual information methods.

The ability to represent or display a 3 dimensional object is fundamental to the understanding of the shape of that object. Furthermore, the ability to rotate, translate, and project views of that object is also, in many cases, fundamental to understand its shape. Manipulation, viewing, and construction of three-dimensional graphic images require the use of three-dimensional geometric and coordinate transformations. In geometric transformation, the coordinate system is fixed, and the desired transformation of the object is done with respect to the coordinate system. In coordinate transformation, the object is fixed and the desired transformation of the object is done on the coordinate system itself. These transformations are formed by composing the basic transformations of translation, scaling & rotation. Each of these transformations can be represented as a matrix transformation. This permits more complex transformations to be built up by use of matrix multiplication or concatenation. We can construct the complex objects/pictures, by instant transformations. In order to represent all these transformations, we need to use homogeneous coordinates.

Hence, if \( P(x,y,z) \) be any point in 3-D space, then in HCS, we add a fourth-coordinate to a point. That is instead of \((x,y,z)\), each point can be represented by a Quadruple \((x,y,z,H)\) such that \( H \neq 0 \); with the condition that \( x1/H1=x2/H2; y1/H1=y2/H2; z1/H1=z2/H2 \). For two points \((x_1, y_1, z_1, H_1) = (x_2, y_2, z_2, H_2)\) where \( H_1 \neq 0, H_2 \neq 0 \). Thus any point \((x,y,z)\) in Cartesian system can be represented by a four-dimensional geometric coordinate as \((x,y,z,1)\) in HCS. Similarly, if \((x,y,z,H)\) be any point in HCS then \((x/H,y/H,z/H)\) be the corresponding point in Cartesian system. Thus, a point in 3-dimensional space \((x,y,z)\) can be represented by a four-dimensional point as:

\[
(x',y',z',1) = (x, y, z, 1) \begin{bmatrix} x & y & z & 1 \end{bmatrix}, \text{ where } [T] \text{ is some transformation matrix and } (x',y',z',1) \text{ is a new coordinate of a given point } (x,y,z,1), \text{ after the transformation. The generalised 4X4 transformation matrix for 3D homogenous co-ordinate is:}
\]

\[
[T] = \begin{bmatrix} a & b & c & w \\ d & e & f & x \\ g & h & i & y \\ l & m & n & z \end{bmatrix}
\]

The upper left \((3x3)\) sub matrix produces scaling, shearing, rotation and reflection transformation. The lower left \((1x3)\) sub matrix produces translation, and the upper right \((3x1)\) sub matrix produces a perspective transformation, which we will study in the next unit. The final lower right-hand \((1x1)\) sub matrix produces overall scaling. The generalized 4x4 transformation matrix for three-dimensional homogeneous coordinates is:

\[
[T] = \begin{bmatrix} a & b & c & w \\ d & e & f & x \\ g & h & i & y \\ l & m & n & z \end{bmatrix}
\]

The upper left \((3x3)\) sub matrix produces scaling, shearing, rotation and reflection transformation. The lower left \((1x3)\) sub matrix produces translation, and the upper right \((3x1)\) sub matrix produces a perspective transformation, which we will study in the next unit. The final lower right-hand \((1x1)\) sub matrix produces overall scaling.

### 3.1 TRANSFORMATION for 3-D TRANSLATION

Let \( P \) be the point object with the coordinate \((x,y,z)\). We wish to translate this object point to the new position say, \( P'(x',y',z') \) by the translation Vector \( V = t_x \begin{bmatrix} 1 \end{bmatrix}, t_y \begin{bmatrix} 1 \end{bmatrix}, t_z \begin{bmatrix} 1 \end{bmatrix} \) where \( t_x, t_y, \text{ and } t_z \) are the translation factor in the x, y, and z directions respectively, as shown. That is, a point \((x,y,z)\) is moved to \((x+t_x, y+t_y, z+t_z)\). Thus the new coordinates of a point can be written as:

\[
x' = x + t_x, \quad y' = y + t_y, \quad z' = z + t_z
\]

The ability to represent or display a 3 dimensional object is fundamental to the understanding of the shape of that object. Furthermore, the ability to rotate, translate, and project views of that object is also, in many cases, fundamental to understand its shape. Manipulation, viewing, and construction of three-dimensional graphic images require the use of three-dimensional geometric and coordinate transformations. In geometric transformation, the coordinate system is fixed, and the desired transformation of the object is done with respect to the coordinate system. In coordinate transformation, the object is fixed and the desired transformation of the object is done on the coordinate system itself. These transformations are formed by composing the basic transformations of translation, scaling & rotation. Each of these transformations can be represented as a matrix transformation. This permits more complex transformations to be built up by use of matrix multiplication or concatenation. We can construct the complex objects/pictures, by instant transformations. In order to represent all these transformations, we need to use homogeneous coordinates.
3.2 TRANSFORMATION FOR 3-D ROTATION

Rotation in three dimensions is considerably more complex than rotation in two dimensions. In 2-D, a rotation is prescribed by an angle of rotation $\theta$ and a centre of rotation, say $P$. However, in 3-D rotations, we need to mention the angle of rotation and the axis of rotation. Since, we have now three axes, so the rotation can take place about any one of these axes. Thus, we have rotation about x-axis, y-axis, and z-axis respectively.

3.3 ROTATION ABOUT Z-AXIS

Rotation about z-axis is defined by the xy-plane. Let a 3-D point $P(x,y,z)$ be rotated to $P'(x',y',z')$ with angle of rotation $\theta$ see Figure 9. Since both $P$ and $P'$ lie on xy-plane i.e., $z=0$ plane their $z$ components remains the same, that is $z=z'=0$.

Thus, $P'(x',y',0)$ be the result of rotation of point $P(x,y,0)$ making a positive (anticlockwise) angle $\phi$ with respect to $z=0$ plane, as shown in fig.

From figure,

$P(x,y,0) = P(r\cos\phi, r\sin\phi,0)$

$P'(x',y',0)=P[r\cos(\phi+\theta), r\sin(\phi+\theta),0]$.

The coordinates of $P'$ are:

$x' = r\cos(\theta+\phi) = r(\cos\theta\cos\phi - \sin\theta\sin\phi)$

$y' = r\sin(\theta+\phi) = r(\sin\theta\cos\phi + \cos\theta\sin\phi)$

Similarly,

$x' = x\cos\theta - y\sin\theta$ (where $x=rcos\phi$ and $y=rsin\phi$)

Thus,

$x' = x\cos\theta - y\sin\theta$

$[R_z]_{h} = [y'] = x\sin\theta + y\cos\theta$

$z' = z$

In matrix form,

$(x',y',z') = (x,y,z)$

$[\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}]$ (a)

In terms of HCS, above equation (a) becomes

$(x',z',1) = (x,y,z,1)$

$[\begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}]$

That is, $P'_{h} = P_{h} [R_z]_{h} --------(b)$
Rotation about the x-axis can be obtained by cyclic interchange of x\(\rightarrow\)y\(\rightarrow\)z\(\rightarrow\)x in equation of the z-axis rotation i.e.,
\[x' = x \cos \theta - y \sin \theta\]
\[y' = x \sin \theta + y \cos \theta\]
\[z' = z\]
After cyclic interchange of x\(\rightarrow\)y\(\rightarrow\)z\(\rightarrow\)x
\[x' = x \cos \theta - y \sin \theta\]
\[y' = x \sin \theta + y \cos \theta\]
\[z' = z\]
So, the corresponding transformation matrix in homogeneous coordinates becomes
\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & \cos \theta & \sin \theta & 0 \\
0 & -\sin \theta & \cos \theta & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]
That is, \(P' = P_h \cdot [R_x]_\theta \)-------(d)
Similarly, the rotation about y-axis can be obtained by cyclic interchange of x\(\rightarrow\)y\(\rightarrow\)z\(\rightarrow\)x in equation (c) of the x-axis rotation \([R_x]_\theta\) i.e.,
\[y' = y \cos \theta - z \sin \theta\]
\[x' = x \sin \theta + z \cos \theta\]
\[z' = z\]
After cyclic interchange of x\(\rightarrow\)y\(\rightarrow\)z\(\rightarrow\)x
\[x' = x \cos \theta - y \sin \theta\]
\[y' = y \sin \theta + x \cos \theta\]
\[z' = z \cos \theta - x \sin \theta\]
So, the corresponding transformation matrix in homogeneous coordinates becomes
\[
\begin{bmatrix}
\cos \theta & 0 & -\sin \theta & 0 \\
0 & 1 & 0 & 0 \\
\sin \theta & 0 & \cos \theta & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]
That is, \(P' = P_h \cdot [R_y]_\theta \)-------(f)

### 3.4 TRANSFORMATION FOR 3-D SCALING
As we have seen earlier, the scaling process is mainly used to change the size of an object. The scale factors determine whether the scaling is a magnification, \(s > 1\), or a reduction, \(s < 1\). Two-dimensional scaling, as in equation (8), can be easily extended to scaling in 3-D case by including the z-dimension.
For any point \((x,y,z)\), we move into \((x.s_x,y.s_y,z.s_z)\), where \(s_x\), \(s_y\), and \(s_z\) are the scaling factors in the \(x\), \(y\), and \(z\)-directions respectively.

Thus, scaling with respect to origin is given by:

\[
\begin{align*}
x' &= x.s_x \\
y' &= y.s_y \\
z' &= z.s_z
\end{align*}
\]

(reduction, \(s<1\). Two-dimensional scaling, as in equation (8), can be easily extended to scaling in 3D case by including the \(z\)-dimension.

For any point \((x,y,z)\), we move into \((x.s_x,y.s_y,z.s_z)\), where \(s_x\), \(s_y\), and \(s_z\) are the scaling factors in the \(x\), \(y\), and \(z\)-directions respectively.

Thus, scaling with respect to origin is given by:

\[
\begin{align*}
x' &= x.s_x \\
y' &= y.s_y \\
z' &= z.s_z
\end{align*}
\]

In matrix form,

\[
\begin{bmatrix}
x' \\
y' \\
z'
\end{bmatrix} = \begin{bmatrix}
x \\
y \\
z
\end{bmatrix} \begin{bmatrix}
s_x & 0 & 0 \\
0 & s_y & 0 \\
0 & 0 & s_z
\end{bmatrix}
\]

In terms of HCS, equation (g) becomes

\[
\begin{bmatrix}
x' \\
y' \\
z'
\end{bmatrix} = \begin{bmatrix}
x \\
y \\
z
\end{bmatrix} \begin{bmatrix}
s_x & 0 & 0 \\
0 & s_y & 0 \\
0 & 0 & s_z
\end{bmatrix}
\]

That is, \(P' = P \cdot s_x.s_y.s_z\)  

3.5 Transformation for 3-D Shearing

Two-dimensional \(xy\)-shearing transformation, as defined in equation (19), can also be easily extended to 3-D case. Each coordinate is translated as a function of displacements of the other two coordinates. That is,

\[
\begin{align*}
x' &= x + a \cdot y + b \cdot z \\
y' &= y + c \cdot x + d \cdot z \\
z' &= z + e \cdot x + f \cdot y
\end{align*}
\]

where \(a, b, c, d, e,\) and \(f\) are the shearing factors in the respective directions.

In terms of HCS, equation (j) becomes

\[
\begin{bmatrix}
x' \\
y' \\
z'
\end{bmatrix} = \begin{bmatrix}
1 & c & e \\
a & 1 & f \\
b & d & 1
\end{bmatrix} \begin{bmatrix}
x \\
y \\
z
\end{bmatrix}
\]

That is, \(P' = P \cdot Sh_{xyz}\)

Note that the off-diagonal terms in the upper left 3x3 sub matrix of the generalized 4x4 transformation matrix in equation produce shear in three dimensions.

3.6 TRANSFORMATION FOR 3D SCALING

For 3-D reflections, we need to know the reference plane, i.e., a plane about which the reflection is to be taken. Note that for each reference plane, the points lying on the plane will remain the same after the reflection.

4. TRANSFORMATION MIRROR REFLECTION ABOUT XY PLANE

Let \(P(x,y,z)\) be the object point, whose mirror reflection is to be obtained about \(xy\)-plane (or \(z=0\) plane). For the mirror reflection of \(P\) about \(xy\)-plane, only there is a change in the sign of \(z\)-coordinate.
In matrix form, 

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & -1
\end{bmatrix}
\]

In terms of HCS (Homogenous coordinate systems), equation (I) becomes

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & -1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

That is, \( P' = P M_y \).

Similarly, the mirror reflection about yz plane shown in Figure 12 can be represented as:

\[
x' = -x \\
M_{yz} = y' = y \\
z' = z
\]

In matrix form, 

\[
\begin{bmatrix}
-1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

In terms of HCS, equation (n) becomes

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

That is, \( P' = P M_{yz} \).

Similarly, the reflection about xz plane, shown in Figure , can be presented as:

\[
x' = x \\
M_{xz} = y' = -y \\
z' = z
\]

In matrix form, 

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & -1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

In terms of HCS, equation (q) becomes

\[
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & -1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]
5. EXPERIMENTAL RESULT

Experiments are performed to verify the effective result of the proposed scheme. In 3-D registration, motion can occur along all three dimensions. Previous work on image registration can be broadly classified into feature-based & area-based methods. Proposed a contour-based method, which uses region boundaries and other strong edges as matching primitives. Features from Accelerated Segment and perform descriptor matching to estimate the image transformation. In contrast to feature-based methods, area-based methods attempt to perform registration without extracting salient features. Common area-based approaches include cross correlation (CC) methods and mutual information (MI) methods. The proposed system integrates the adaboost algorithm and area-based methods to improve the 3D information for more accurate results. Experimental results using the proposed show that the new approach can increase the speed and give 3D output with excellent accuracy.
6. CONCLUSION

This paper presents a face detection approach with high speed and 3 dimensional information with accuracy. The transformations are formed by composing the basic transformations of translation, scaling, and rotation. Each of these transformations can be represented as a matrix transformation. This permits more complex transformations to be built up by use of matrix multiplication or concatenation. We can construct the complex objects/pictures, by instant transformations. We represent all these transformations using homogeneous coordinates. The results obtained from the proposed approach are illustrated as effective and practical.

REFERENCES


ACKNOWLEDGEMENT

We would like to acknowledge Prof. K.P.Rane Associate Prof Godavari COE Jalgaon & Prof. V.D. Chaudhari Assistant Prof Godavari COE for their support.

AUTHOR

Prof. PriyankaKulkarni
She has completed her B.E.(E&TC) from Ndmvp’s COE Nashik and is pursing M.E(VLSI & EMBEDDED) from Godavari College of Engg. Jalgaon. She is working as Assistant Prof in Sapkal COE & Research Center and her teaching experience is 4.5yrs.