Detection of Anchor Points for 3D Face Verification

Dirk Colbry, George Stockman, and Anil Jain
Department of Computer Science and Engineering
Michigan State University
East Lansing, Michigan 48824-1226
{colbrydi, stockman, jain}@cse.msu.edu

Abstract

This paper outlines methods to detect key anchor points in 3D face scanner data. These anchor points can be used to estimate the pose and then match the test image to a 3D face model. We present two algorithms for detecting face anchor points in the context of face verification; One for frontal images and one for arbitrary pose. We achieve 99% success in finding anchor points in frontal images and 86% success in scans with large variations in pose and changes in expression. These results demonstrate the challenges in 3D face recognition under arbitrary pose and expression. We are currently working on robust fitting algorithms to localize more precisely the anchor points for arbitrary pose images.

1 Introduction

The registration problem is a pressing one for many automatic pattern recognition systems. In order to properly compare patterns, the data need to be normalized. We have investigated various anchor point detection methods that can be used to register 2.5D face images taken with a Minolta VIVID 910 range scanner [11]. A 2.5D image is a simplified 3D (x, y, z) surface representation that contains at most one depth value (z direction) for any point in the (x,y) plane (see Figure 1).

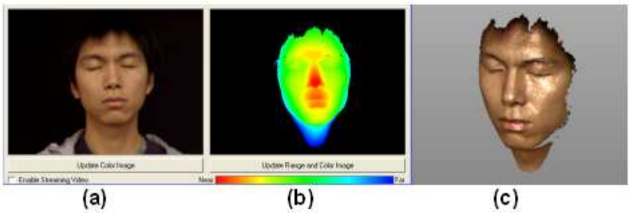

Figure 1. Example 2.5D image (a) 2D Texture Image (b) Depth Map (c) Rendered 3D Model.

Current 2D face recognition systems can achieve acceptable performance for face verification in constrained environments; however, they encounter difficulties in handling large variations in pose and illumination [7]. Utilizing 3D information can improve face recognition performance [3, 5]. Moreover, 3D face pose is required in related applications such as telepresence. Range images captured by common 3D sensors [9, 11] explicitly represent face surface shape information as well as provide registered 2D color images. Face recognition based on range images has been addressed in different ways [10, 5]; however, most of these systems do not attempt to automatically find anchor points and register faces with arbitrary pose, lighting and expression. Some existing work on frontal face anchor point detection can be found in [4].

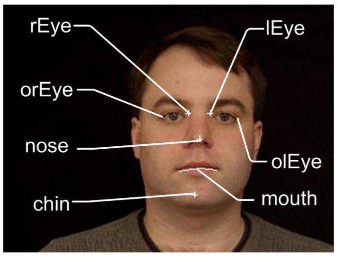

Figure 2. Anchor point locations. rEye - Inside of the right eye; orEye - Outside of the right eye; lEye - Inside of the left eye; olEye - Outside of the left eye; nose - Nose tip; chin - chin tip; mouth - corners and middle of the mouth.

In Section 2 we describe how to calculate the shape index [6], which is a pose invariant representation of the surface curvature. In Sections 3 and 4 we investigate automatic methods for detecting face anchor points in both frontal face scans and arbitrary pose scans, respectively. The anchor points are used to generate a coarse alignment between the

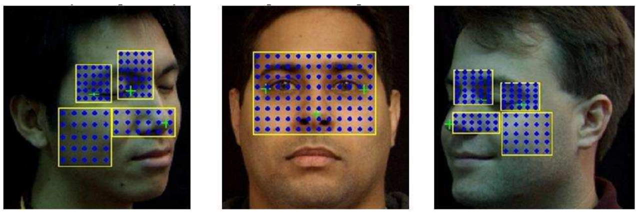

Figure 3. Control points used by ICP.

scan and the database model. This coarse alignment is used as a starting point for the ICP algorithm [2], which is then used to finely align the scans. Only a single anchor point is necessary to calculate the coarse transformation between two frontal scans. In contrast, three points are needed if the test scan and the model have different poses. Figure 2 shows examples of some of the anchor points our system is designed to detect. These anchor points are easy to find within the image, span the width of most faces, and do not require that the eyes be open during image capture.

In addition to calculating the coarse transformation, the detected anchor points are also used to calculate the location of the grid of control points needed by the ICP algorithm for fine alignment (see Figure 3).

2 Shape Index

To find anchor points on the face, it is advantageous to use attributes that do not change with pose. One such attribute is surface curvature. The local curvature information about a point is independent of the coordinate system. Dorai and Jain [6] proposed local shape information, called the Shape Index, to represent the surface curvature at each point within a 2.5D image. The Shape Index at point p is calculated using the maximum (k_1) and minimum (k_2) local curvature (see Equation 1). This calculation produces a shape scale between zero and one (see Figure 4).

$$S(p) = \frac{1}{2} - \frac{1}{\pi} tan^{-1} \frac{k_1(p) + k_2(p)}{k_1(p) - k_2(p)} \qquad (1)$$

Since the shape index is independent of the coordinate system, it is useful for finding similar points between scans from different poses. The faces shown in Figure 5 are examples of 2.5D face images with intensity representing the shape index. Notice that there are several consistencies between these two scans of the same face. For example, the area between the eyes and the bridge of the nose is consistently *rut* shaped as is the area around the mouth. We use these consistencies to help locate prominent anchor points within the face regardless of the orientation of the face in the scan.

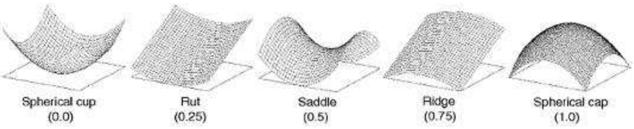

Figure 4. Shape index scale. A value of 0 represents a spherical cup while a value of 1.0 represents a spherical cap. This index is independent of the coordinate system.

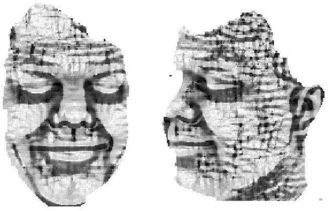

Figure 5. Example shape index images. The dark regions correspond to a shape index of 0 while the lighter regions represent a shape index of 1.

3 Frontal Anchor Point Detection

Commercial three-dimensional face verification systems require cooperative subjects, where it is expected that the subject stand at a given location with a controlled pose and facial expression. The Minolta scanner that we are using takes approximately 1 second per scan during which time the subject must hold still. Hopefully, future scanning systems will not have these limitations.

Detecting anchor points in a frontal scan is not a difficult problem. The structure of the head is known and there are prominent anchor points that can be detected easily if the pose of the head is known. We have found that the easiest point to find is the tip of the nose. In our database of 330 frontal faces, the tip of the nose is the point closest to the camera in 92% of the scans.

When dealing with frontal posed faces we define the z gradient as the change of the z distance with respect to the x and y directions. This change is normally large from the base of the nose to the tip of the nose, and from the edges of the face and hair. However, in most other locations this gradient is small. It is common for spike noise to appear in the scans; this is easily removed by filtering out points that have a gradient that is larger than those found around the nose. Filtering the scan before selecting the nose point increases the success rate of the nose detector to 97%. If we limit our search to areas that may reasonably contain the nose (see Section 3.1), we can increase the success rate to

99%. When the orientation of the head is known, it is sufficient to have one anchor point to initially align the head with the model. However, in order to allow for minor variations in the pose angle (± 15 deg) other anchor points on the face may be needed.

3.1 Interpoint Statistics

We have a statistical model that identifies where the anchor points may be located on the face. These statistics of distances between anchor points were developed by analyzing a database of over 2000 test scans. The minimum, maximum and average distance are included in the model to bound the region of search for each of the anchor points (See Table 1).

Using this simple model, bounding boxes can be generated that greatly reduce the search for a particular anchor point. For example, if the tip of the nose is found, bounding boxes can be generated to locate many of the other anchor points such as the eye corners or chin (see Figure 6). These bounding boxes not only increase the efficiency of the algorithm but can also improve the performance of individual anchor point detectors.

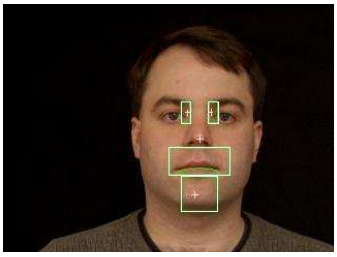

Figure 6. Example bounding boxes estimated by the statistical face model and the location of the nose tip.

3.2 Anchor Point Detection Algorithm

The frontal anchor point detection algorithm (outlined in Table 2) starts by finding the top of the head. Any point near the top of the head should do because it only establishes the vertical location of the head. Once the top of the head is found, a bounding box for the location of the nose can be produced. The algorithm then uses other bounding boxes to localize the search for other anchor points. Each point is found using detection decisions based on local shape characteristics with parameters trained on sample scans.

1. Use the gradient filter to filter out depth spikes.
2. Find the top of the head and generate nose bounding box.
3. Find the tip of the nose as the closest point of the scan.
4. Calculate the bounding box for the inside corners of the eye, mouth, and chin.
5. Calculate the location of each individual anchor point using the shape characteristics.

Table 2. Frontal Pose Anchor Point Detection Algorithm.

3.3 Experiments and Results

We tested our frontal anchor point detection algorithm on the following data sets:

- Data Set A - Our main data set was composed of 111 test subjects with approximately three frontal scans each (two with neutral expression and one smiling) for a total of 330 test scans. Scans have little spike noise and most faces have a neutral expression and are frontal. See Examples in Figure 7. This data was collected in our laboratory.
- Data Set B - 953 scans taken from a database provided by the University of Notre Dame. These scans are all frontal neutral expression with higher levels of noise compared to data Set A due to varying lighting conditions [12].
- Data Set C - Limited data set containing 49 images from around 5 subjects, but it explores significant variation due to background changes, rotations, occlusions, expression, lighting and noise. Most of these images do not satisfy the specifications for which the face verification system is being built. See examples in Figure 8. This data was collected in our laboratory.

Tests were run on all three data sets and the results are reported in Table 3. The anchor point detector was considered successful if it located three of the five anchor points (see Figure 2) correctly. Success was a qualitative judgment made by the investigator and verified by checking the convergence of the ICP algorithm with another scan from the same subject. Of the five anchor points detected, the worst was the mouth due to its large variation in expression, so the success rate for the mouth is also reported as a reference. The system was run on a Pentium 4 CPU with two 3.20GHz processors and 1.00GB of RAM. All of the times reported on are wall time and will vary depending on the load of the cpu, this load variation being most noticeable during file loading. Using the ICP to verify point detection added at most 2 seconds to the computation time.

Point A	Point B	direction	Minimum (mm)	Average (mm)	Maximum (mm)
Top of Head	Nose Tip	vertical	84.0	121.7	181.0
Nose Tip	Mouth	vertical	16.9	34.4	46.3
Nose Tip	Chin	vertical	49.7	70.6	95.4
Nose Tip	Nose Bridge	vertical	21.2	32.2	48.4
Nose Tip	Inside Eye	horizontal	13.2	17.5	23.6
Inside Eye	Outside Eye	horizontal	15.9	30.6	39.7

Table 1. Statistical Model of the inter-anchor point distances on a face. Each set of distances are calculated from point A to point B only in the specified direction.

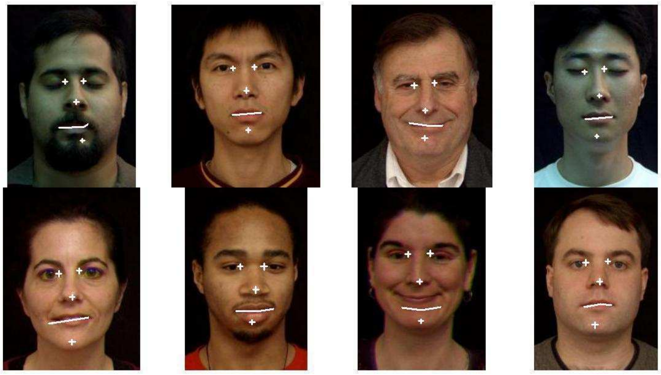

Figure 7. Detected anchor points in example images from Data Set A.

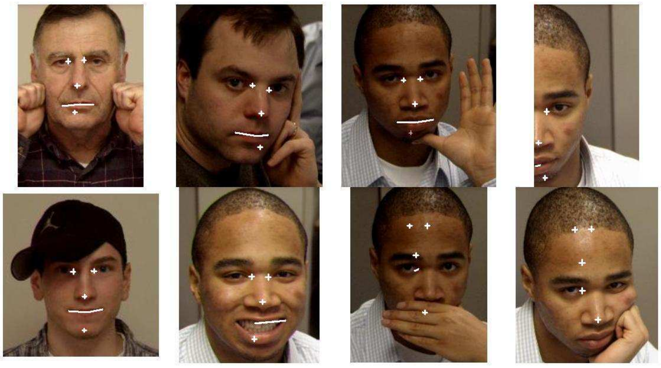

Figure 8. Detected anchor points in example images from Data Set C (these images violate the specs under which the face verification system is being built).

Data Set	Number of Frontal Scans	Detection Rate	Detection Rate Mouth Only	Average Algorithm Time (sec)
A	330	99.1%	97.0%	0.33
B	953	99.8%	97.4%	0.30
C	49	85.7%	83.7%	0.33

Table 3. Experimental Results for Frontal Anchor Point Detector.

4 Arbitrary Pose Anchor Point Detection

The frontal anchor point detection algorithm is sufficient when dealing with cooperative subjects in a constrained environment. However, we are also interested in performance under large variations in pose. We have found that detecting feature points under arbitrary poses is a difficult problem. There are three major steps in our arbitrary pose algorithm, which must align a scan to a model with significantly different pose; candidate point selection (Section 4.1), relaxation/search algorithm (Section 4.2), and fitness function evaluation (Section 4.3).

4.1 Candidate Point Selection

With unknown pose, we can no longer guarantee that the user is facing the scanner. Anchor points need to be detected robustly regardless of the pose of the face. The general locations of the anchor points are determined by using information about the shape index as well as a model of the face. It can be difficult to identify specific points based on the shape index alone. For example, finding a region of high curvature does not necessarily result in a single point. Once the general location of the point is found, we further refine the results by finding a more specific point. This is normally done using an edge map generated from the (registered) 2D color component of the scan. We can use this map to find a specific edge point (such as a corner point) closest to the general anchor point location.

In the remainder of this section, we describe the detection of a core set of candidate points: the inner eyes, the nose tip and the outer eyes. Problems can occur in profile scans where only a few of the desired anchor points are available for detection. For example, in a profile scan the eyeball may occlude the inside corner point of the eye. In these cases, additional labeled points are needed to increase the likelihood of finding three good anchor points. For this reason, we have provided four additional point labels for the arbitrary pose anchor point detector: two points on the tips of the mouth and a pair of non-surface points (centroid and top-centroid) that represents the central axis of the scan. The centroid is calculated as the average location of all the points in a scan and the top-centroid is the same point moved 100mm in the z-direction. In most scans these points do not align well with the models. However, due to their similar structure, they can provide face orientation information if other points are not available. The chin point was not detected in this study.

In our algorithm spurious candidate points are acceptable. This simplifies the development of detection decisions. Points can be selected by simple criteria and eliminated later by the constraint relaxation algorithm. This makes the system flexible in special cases because anchor points can be selected for specific situations.

4.1.1 Inner Eye Point Detection

The easiest anchor point to identify in arbitrary pose is the inside edge of an eye next to the bridge of the nose. This point has a shape index value that is close to zero and the area around this point has a consistent shape index value across all face images and poses. An averaging mask (size 30 x 10) can be used to identify this area in the shape space (see Figure 9). More complex masks were considered but we have found that the averaging mask is robust across face scans. The averaging mask identifies the point with the largest average *cup* shaped curvature. We use the inside edge of the eye as an anchor in order to identify other anchor points within the face image. Because there are two eyes, the two largest regions of high curvature are selected. Often, this results in a point within the ear or hair being classified as a possible eye (see Fig 9b); this is tolerable as long as the constraints can eliminate these points later.

4.1.2 Nose Tip Detection

The next easiest anchor to detect is the nose tip. For most people this point sticks out beyond the face and is salient in most poses. For example, Figure 10 shows a face scan where up to six different simple criteria (outlined in Table 4) were used to find nose tip candidates. The expectation is that one of the points produced by these six criteria is in fact the tip of the nose. In order to evaluate the quality of this set of points, these criteria were applied to over 1000 face scans in our database and the minimum distance to the actual nose tip was recorded. The average minimum distance was 5.1mm with a standard deviation of 8.3mm. We have experimentally determined that any point on the nose within a 15mm radius from the tip is an adequate starting point for the ICP algorithm. This may seem like a large tolerance, but the goal of our anchor point detection algorithm is to come up with a coarse alignment that can then be passed to ICP for fine alignment of the scans. By far, most of the nose points detected by our algorithm had close to zero error; however, there are still cases where we missed the nose tip completely. In the cases where the nose point is missed, there is still a possibility that the relaxation algorithm (discussed in Section 4.2) will find a good set of three anchor points that does not include the nose tip.

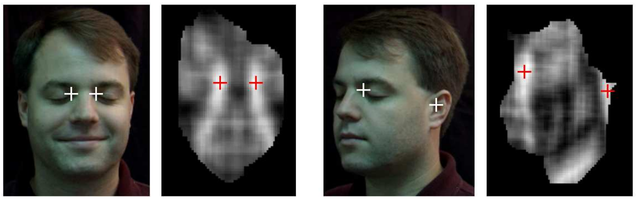
(a) (b)

Figure 9. Convolution of the shape space with a simple vertical mask to find the inside points of the eyes. (a) frontal view, (b) a semi-profile view. The bright spots in the scan represent the locations of the inside of the eye next to the bridge of the nose.

1. Point closest to the scanner (minimum z direction point).
2. Point farthest to the left (minimum x direction point).
3. Point farthest to the right (maximum x direction point).
4. Point farthest from the vertical plane formed by points 1 and 2.
5. Point farthest from the vertical plane formed by points 1 and 3.
6. Point with the largest shape index.

Table 4. Criteria for finding candidate nose points.

4.1.3 Outer Eye Point Detection

Once the inner eye candidate points are found (Section 4.1.1), the outside of the eye is detected by following the *rut* (defined by the shape space) that consistently runs along the bottom of the eye (see Figure 11). This calculation is made from all possible eye candidates.

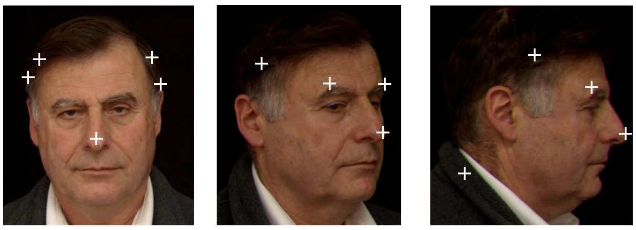

Figure 10. Nose candidate points for three different poses. Notice that in all of the scans at least one of the candidate points is actually the nose tip.

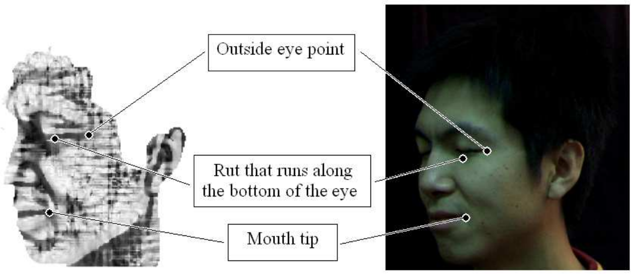

Figure 11. An example shape-space scan with key anchor points identified.

4.2 Relaxation / Search Algorithm

After calculating the list of candidate points, the next step is to search through these candidates to find possible sets of three anchor points and the associated labels. For example, even if only three candidate points are found, each of these three points could be assigned up to nine labels (Nose Tip, Inside Right Eye, Inside Left Eye, Outside Right Eye, Outside Left Eye, Left Mouth Corner, Right Mouth Corner, Centroid, and Top Centroid), which results in 504 possible labelings. When more candidate points are selected, the total number of labelings goes up exponentially. Therefore, an exhaustive search of all candidate points with all possible labelings is not feasible. Instead, some simple constraints (see Section 4.2.2) are developed to help filter the labeling possibilities. There are many methods for solving constraint satisfaction problems [1]. We have chosen discrete relaxation filtering because it is easy to implement and is efficient given that there is a limited number of candidate points as well as a limited number of possible labelings [8].

There are two components to discrete relaxation: the relaxation matrix, and the relaxation constraints. Using a well-formulated combination of the relaxation constraints, the relaxation algorithm is able to iteratively eliminate impossible labellings using the following rule:

Relaxation Rule - If a particular label assignment for any point fails ALL the constraints, then this labeling is not possible and the corresponding row and column in the relaxation matrix should be changed to zero.

This relaxation rule follows the philosophy of least commitment where only the obviously wrong labels or points are eliminated.

4.2.1 Relaxation Matrix

The relationship between the candidate feature points and the possible feature labels is represented as a $(n \times m)$ Relaxation Matrix (R), where n represents the number of candidate points and m is the number of point labels. A value of one in this matrix represents the possibility of a point having a particular label. A value of zero means the point cannot have that particular label. A null label is used for points that do not match up with any label. The matrix is initialized with all ones because without any knowledge from our constraints, every point could have every possible label.

4.2.2 Constraints

For the arbitrary pose anchor point recognition problem, the following four constraints are used: shape constraint, absolute distance constraint, relative distance constraint and relative cross-product constraint.

Shape constraint - The shape index range of a candidate point with a specific label must fall between a minimum and maximum value. Note that the shape index is a unary constraint and only needs to be applied once when the registration matrix is being initialized.

$$\begin{gathered} S_{min} < S_L < S_{max} \\ \text{Example: } 0.7 < S_{nose} < 1.0 \end{gathered} \tag{2}$$

Absolute Distance Constraint - The distance between any two labeled points must be between a minimum and maximum value.

$$\begin{gathered} D_{min} < |P_{L1} - P_{L2}| < D_{max} \\ \text{Example:} \\ 30mm < \sqrt{\sum_{i \in x,y,z} (P^i_{nose} - P^i_{rEye})^2} < 140mm \end{gathered} \tag{3}$$

Relative Distance Constraint - The distance between a single component (x, y, or z) of any two labeled points must be between a minimum and maximum value.

$$\begin{array}{c} D_{min} < (P^x_{L1} - P^x_{L2}) < D_{max} \text{ or} \\ D_{min} < (P^y_{L1} - P^y_{L2}) < D_{max} \text{ or} \\ D_{min} < (P^z_{L1} - P^z_{L2}) < D_{max} \end{array} \quad (4)$$

Example: $10mm < (P^x_{rEye} - P^x_{lEye}) < 100mm$

Relative Cross-Product Constraint - The cross product between the vectors formed by (P_{L1}, P_{L2}) and (P_{L1}, P_{L3}) has a particular direction within a single ($x, y,$ or z) component.

$$\begin{array}{c} [(P_{L1} - P_{L2}) \times (P_{L1} - P_{L3})]^x \cdot Dir > 0 \text{ or} \\ [(P_{L1} - P_{L2}) \times (P_{L1} - P_{L3})]^y \cdot Dir > 0 \text{ or} \\ [(P_{L1} - P_{L2}) \times (P_{L1} - P_{L3})]^z \cdot Dir > 0 \\ \text{where } Dir = \pm 1 \end{array} \quad (5)$$

Example: $[(P_{L1} - P_{L2}) \times (P_{L1} - P_{L3})]^z \cdot Dir > 0$

These four constraints were derived by analyzing example scans and models and determining the relative distances between anchor points. Similar constraints can be added to speed up the algorithm; however, the four constraints described in this section proved to be sufficient. One challenge in identifying these constraints was to not make too many assumptions about the relative locations of the anchor points. For example, the left eye points are not always located to the left of the nose; in some profiles, the left eye is actually to the right of the nose.

4.3 Fitness Functions

The goal for the anchor point detection algorithm is to find three labeled points that correspond to three points labeled on the 3D face model. These three pairs of points are used to calculate a transformation from the test scan to the model in the database. This transformation coarsely aligns the test scan to the 3D model, and then the ICP algorithm calculates the best matching distance between the scan and the model using a grid of approximately 100 control points defined by the anchor points (see Figure 3).

In order to evaluate the quality of the set of anchor points the matching score generated by the ICP algorithm is used as a fitness function. The key to using ICP is to have a set of consistent control points from within the scan. Because we do not yet know the pose of the scan, we cannot use the control points shown in Figure 3. Instead, all noisy data (in the z direction) is filtered out of the scan and then an evenly spaced 20 x 20 grid of points is placed on the remaining scan. Because a symmetrical grid is used, not all of these points are valid due to holes in the scan. The invalid points are discarded. The same final set of control points is used in all of the fitness calculations so that the distance errors generated by the ICP algorithm are comparable. The downside of using ICP as a fitness function is that it requires the use of a 3D face model to calculate the distance value.

4.4 Experiments and Results

For arbitrary pose, finding the correct set of anchor points takes more time than in a frontal scan because all of the candidate points must be searched and for each valid set of three candidate point labelings, the ICP algorithm is run to determine the fitness of the points. On an average scan, the arbitrary pose anchor point detector takes around 15 seconds to complete. In order to evaluate the quality of the anchor points, each detected anchor point was compared to a manually selected point. However, many times the manually selected points were not as good as the automatically selected points. This occurred when the experimenter selected a point that looked correct in the texture image but actually was incorrect in the registered range image due to the movement of the subject's head. This was a common problem when points were near the edges of the scan where the range data was invalid. The system was tested on a database of 600 scans taken of 100 subjects (6 scans per subject). The scans include frontal, full profiles and smiling expressions. Each test scan in the data set was compared to the 3D model of the same subject generated from an additional 5 scans stitched together to produce a 3D surface model.

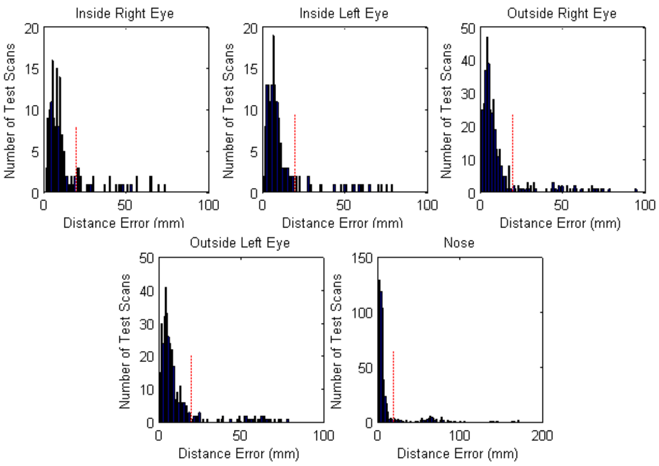

Figure 12. Error Histograms for each of the five anchor points. The error represents the distance from the final detected point to the manually selected point. The dotted vertical line represents the 20mm cutoff; 90% of the scans fall below this value.

The histograms of the error for all five of the core surface anchor points are shown in Figure 12. The median error of the five points is around 10mm and approximately 90% of the scans are below 20mm error for each anchor point. Notice that the outliers (exspecially with the nose) represent

Attribute	Population Size	Success Rate
Female	25.2%	85.4%
Male	74.8%	85.7%
Facial Hair	11.2%	80.6%
Dark Skin	10.0%	81.7%
Eyes Closed	12.0%	98.6%
Asian Features	26.5%	84.3%
Profile	67.3%	79.6%
Frontal	32.7%	97.7%
Smile	47.6%	82.7 %
No Smile	52.4%	88.5%

Table 5. Arbitrary pose anchor point detection success rate as a function of subject attribute.

Pose	Neutral Expression	Smile
Front	99.0% (17.1%)	97.7% (15.5%)
Profile	82.7% (82.7%)	75.0% (32.1%)

Table 6. Percent of correct anchor points detected versus the pose direction and facial expression. Numbers in the parenthesis indicate the number of test scans.

a total incorrect match. This can occure when other objects such as a coat or hair takes on a shape appearance that is similar to the feature. Extensive testing shows that the ICP algorithm can tolerate up to 20mm of anchor point errors for matching.

To quantify the success rate of the entire transformation, all of the output transformations were evaluated by hand. Each test scan produced a transformation of the test scan onto the 3D model. If this transformation was judged to be good enough for ICP to converge, then the anchor point detection was labeled as successful for this scan. Of the approximately 600 test scans, there was an 85.6% success rate when matching a subject's test scan to the same subject's 3D model.

To fully understand where the errors are occurring, the test scans were also separated into groups based on subject attributes (see Table 5). From this table we can see that facial hair and dark skin make it more difficult to identify key facial anchor points. This is an understandable result because both of these attributes increase the noise produced by the scanner. It is also interesting to note that it is easier to identify anchor points in scans with eyes closed than with the eyes open. This is probably also due to the increase in surface noise that occurs with the eyes open.

What is more interesting is how these attributes work together. Table 6 shows how the success rate of the system varies with pose and facial expression. As Figure 6 demonstrates, the anchor point detection system works 99.0% of the time on neutral expression, frontal pose scans.

5 Concluding Discussion

We have developed algorithms that detect face anchor points using 3D models and knowledge of the structure of the face. These algorithms produce good results, especially for frontal scans, given our current system goals. However, in order to achieve higher levels of accuracy we need to consider algorithms that are more robust to noise and local variations. The frontal anchor point detection system is vulnerable to cascade style problems because each anchor point is dependent on the quality of the previous anchor point. To overcome this, we could detect multiple candidates and backtrack. The arbitrary pose anchor point detection is slow and has significantly lower accuracy than frontal pose point detection. We are currently working toward a new anchor point detector that is more robust to noise and other variations in input.

With robustly identified anchor points, the distances between the points can be more accurately calculated. These distances may then be used for soft binning of the subject pool in order to reduce the number of comparisons needed for a face identification system.

References

[1] R. Bartak. Theory and practice of constraint propagation. In *CPDC2001 Workshop*, pages 7–14, Gliwice, 2001.

[2] P. Besl and N. McKay. A method for registration of 3-d shapes. *IEEE Trans. PAMI*, 14(2):239–256, 1992.

[3] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *IEEE Trans. PAMI*, 25(9):1063–1074, 2003.

[4] C. Boehnen and T. Russ. A fast multi-modal approach to facial feature detection. In *Workshop on Applications of Computer Vision*, 2004.

[5] K. I. Chang, K. W. Bowyer, and P. J. Flynn. An evaluation of multimodal 2d+3d face biometrics. *IEEE Trans. PAMI*, 27(4):619–624, 2005.

[6] C. Dorai and A. Jain. Cosmos - a representation scheme for 3d free-form objects. *IEEE Trans. PAMI*, 19(10):1115–1130, 1997.

[7] FRVT. Face recognition vendor test, 2002.

[8] R. Hummel and S. Zucker. On the foundations of relaxation labeling processes. *IEEE Trans. PAMI*, 5:267–287, 1983.

[9] C. Inc. http://www.cyberware.com/.

[10] X. Lu, A. K. Jain, and D. Colbry. Matching 2.5d face scans to 3d models. *To Appear in IEEE Trans. PAMI*, 2005.

[11] Minolta. Vivid 910 non-contact 3d laser scanner http://www.minoltausa.com/vivid/.

[12] P. Phillips, P. Flynn, T. Scruggs, K. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek. Overview of the face recognition grand challenge. In *CVPR*, 2005.