A Fast Multi-Modal Approach to Facial Feature Detection

Chris Boehnen
University of Notre Dame, cboehnen@nd.edu
Sandia National Laboratories[1], cbboehn@sandia.gov

Trina Russ
Sandia National Laboratories[1], tdruss@sandia.gov

Abstract

As interest in 3D face recognition increases the importance of the initial alignment problem does as well. In this paper we present a method utilizing the registered 2D color and range image of a face to automatically identify the eyes, nose, and mouth. These features are important to initially align faces in both standard 2D and 3D face recognition algorithms. For our algorithm to run as fast as possible, we focus on the 2D color information. This allows the algorithm to run in approximately 4 seconds on a 640X480 image with registered range data. On a database of 1,500 images the algorithm achieved a facial feature detection rate of 99.6% with 0.4% of the images skipped due to hair obstruction of the face.

1. INTRODUCTION

Face recognition (FR) is becoming an increasingly attractive technique for real world systems in which unobtrusive reliable identification of individuals is desired. As a result, the ability to automate the entire face recognition process from acquisition to verification/identification is important. For most 2D and 3D recognition algorithms, it is critical that faces are aligned before being compared. Typically, alignment begins with the detection of facial features. At the research and development stage, alignment features are normally obtained by manually identifying consistent points or features across different faces. In commercial 2D FR systems, techniques exist to find alignment features, but additional robustness is needed to deal with facial, image, and environmental variations (i.e. lighting, image quality, resolution, facial expression, etc.) [1].

Most face recognition algorithms utilize either the 2D or 3D data but not both simultaneously. However, most current range sensors provide 2D color intensity information and 3D geometric information in a corresponding grid; thus, one should use the benefits of both to achieve greater performance and robustness. In addition, 2D feature extraction algorithms (i.e., FaceIt [2]) typically require very good image quality (i.e 1024x768, good lighting, etc.) in order to achieve high performance rates. This makes using traditional techniques on a 2D color image acquired during a range scan difficult because the associated images are typically small and of low image quality. To further aggravate the problem, lighting conditions, which would be ideal for a 2D color image, are usually non-ideal for active range scanners. Bright lighting conditions which are needed for high quality 2D color images reduce the visibility of the active projection elements, such as lasers used in some range scanners. These factors motivate the work presented in this paper. We propose an algorithm that utilizes a combination of 2D and 3D registered data which should result in an accurate facial feature detection algorithm capable of running quickly because it strongly leverages its 2D components.

2. PREVIOUS WORK

2.1 2D methods

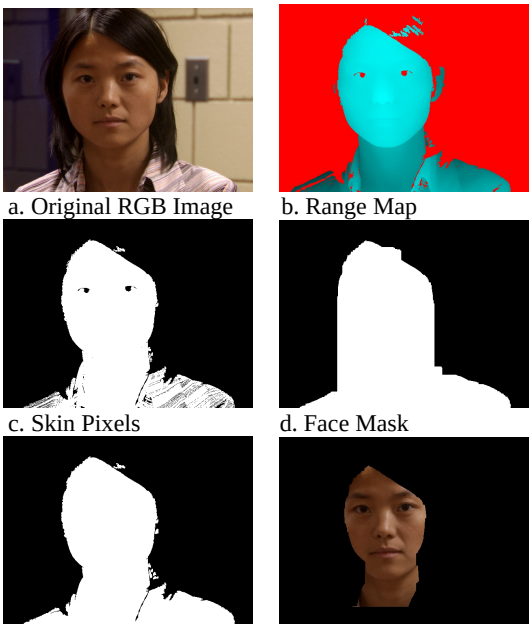
Facial feature detection algorithms operating on 2D color and grayscale images exist and are able to identify the eyes and mouth somewhat reliably. Examples of current methods for identifying facial features use eigenfeatures [3], deformable templates [4] [5] [6], Gabor wavelet filters [7], color manipulation methods [8], edge holistic [9], graph matching [10], etc. These methods work fairly well with facial feature identification rates ranging from 90-98% [1]. The eyes are an important feature that can be consistently identified. In fact, facial identification algorithms that use only the eyes can achieve an 85% facial classification rate [11]. As a result, the center or edges of the eyes are commonly used by feature detection algorithms as a reference point. Commercial software to identify the eyes exists as well. For example, Faceit relies on identifying facial symmetry in black and white images to locate the eyes. Genetic algorithms also exist that can deal with minor pose

[1] Sandia is a multi-program laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

variation with success rates of 80% [12].

2.2 3D methods

3D facial feature extraction is a relatively new area of research with a less extensive body of work than 2D facial feature extraction. Lu *et al.* [13] identified features through a 3D model from multiple scans stitched together and proposed using a shape index to identify the inside of the eye (next to the nose) as a saddle location. The average distance to the true feature locations was 10 mm resulting in relatively low precision. Gordon [14] proposed a method using face shape based on curvature to identify facial features. Her method worked well, however it was tested on a small database of 24 range scans. 2D and 3D motion tracking of faces has also been performed [15] using active appearance and morphable models. Another interesting method is a Gabor filter based curvature approach [16].



a. Original RGB Image b. Range Map c. Skin Pixels d. Face Mask e. Z-based Erosion f. Final Eroded Facial Region

Figure 1. The facial region identification process

3 ALGORITHM

The work presented in this paper uses the purely 2D approach of Hsu and Jain [8] as a foundation. The algorithm first segments skin regions using an elliptical grouping model in *YCbCr* color space for skin. Then a convex polygon hull around the largest connected skin group is computed to fill in facial pixels that were not detected as skin. Subsequently, eye and mouth maps are defined by performing operations in *YCbCr* color space which produces a grayscale image map with higher values in regions of the desired features. Eye and mouth candidates are determined by processing these maps. The ratio of pixel distances is then used to determine if the detected features are typical of a human face. This algorithm obtained a 90% detection rate in identifying facial features in 206 640X480 color images from the Heinrich Hertz Institute (HHI). Several other databases were also tested with accuracy ratings from 80-97%.

Our approach first segments the foreground from the background using the range data. Then, a modified version of the skin detection algorithm from [8] is used to identify potential skin pixels. We refine these pixels further using z-based range erosion and standard erosion algorithms, and then compute the eye and mouth maps. Initially, the formulas from [8] are used, but then we modify them to use additional range and color information. Finally, the geometric-based confidence of potential feature candidates is computed to aid in the selection of the best feature set.

Our algorithm differs from the approach in [8] in that we seek to tightly couple information in 2D color and range imagery to increase feature detection performance without suffering from the computational load associated with using a purely 3D approach. In addition to computing the eye and mouth locations, we also detect the nose since it is a distinctive 3D feature on the face and is likely to improve the overall performance of our algorithm. In our approach, range imagery is used to refine the skin region of interest, determine the nose location, better isolate potential eye locations, refine the mouth location, and assist in identifying the best eye, nose, mouth combination.

Another unique attribute of our approach is a geometric-based confidence measure. Instead of using relative distances between points to determine the best feature group candidate, we use a measure that incorporates actual 3D distances from the range image. This is a powerful metric because it allows the determination of the best feature group and of incorrect feature detections after a low-confidence grouping has been identified.

3.1 2D Skin region refinements

Although the 2D components outlined in this paper are based on [8], some enhancements were made to improve system performance when confronted with poor color images and to improve the computational cost associated with the algorithm. Prior to processing, light compensation was performed to adjust for adverse lighting conditions [8]. Also, we have a lower quality 2D color image compared to [8], which necessitates greater latitude in defining skin regions. We accomplish this by defining a larger elliptical area in *YCbCr* to be

detected as skin. When the skin detection algorithm is combined with only examining valid range pixels one obtains an initial skin estimate as shown in Figure 1.c. In addition, instead of computing the geometric convex hull to fill in skin region holes, we use a less computationally intensive method that fills in the pixels between the minimum and maximum rows and columns for each detected skin component which produces the result shown in Figure 1.d. Currently, our system works under the assumption of a single face in the scan. However, to allow for multiple faces, one would identify groups of skin pixels over a certain size as a face and process each one for facial features.

3.2 Skin refinement using z-based erosion

Range information can be used to refine skin region detection by using valid range pixels to better determine regions of interest. It is expected that valid range pixels will correspond to regions on the face because in many cases range sensors have limited operational range, causing objects out of range to have an invalid value. Determining an accurate representation of skin regions reduces the number of potential feature groups to examine and may reduce the number of incorrect feature detections.

When filling in holes and applying masks to refine the skin region, one often misclassifies pixels with invalid range values as skin. These misclassified pixels can cause an undesirable affect. Thus, a z-based erosion algorithm was implemented to improve the classification of skin/non-skin pixels. The z-based erosion algorithm works by removing every skin pixel which has an invalid range value and has an 8-connected neighbor *n(i)* that is a non-face/skin pixel. A non-face/skin pixel is one that was previously classified as skin, but upon further investigation (i.e. examining range) was determined not to be skin. The variable *Q(i)* is true if *i* is a non-face skin pixel as is determined by (1). In the following, all of the variables are binary such that *S(i)* is true if pixel *i* is currently a skin pixel, and *R(i)* is true if pixel *i* has a valid range value.

$$Q(i) = S(i) \cap \overline{S(n(i))} \cap \overline{R(i)} \qquad (1)$$

This approach erodes the edge of the face group that has invalid range values and neighbors that have not been classified as skin, but will maintain values in the interior of the group such as in the eyes and near the nose as part of the face region. This process is iteratively repeated until there are no remaining pixels that meet these requirements. The algorithm effectively tightens the region around the face and segments it from the background as in Figure 1.e. This is very important as misidentifying parts of the background as part of the face can lead to a many errors that are much more difficult to handle than if range information was not available.

While this approach effectively produces a tight region around the face, small bits of hair and other edge based artifacts can remain. Therefore, we subsequently perform a standard erosion algorithm to remove the edge pixels that may contain edge and hair artifacts as in Figure 1.f. While it is desirable to get rid of some of the exterior artifacts, it is important not to erode too much (i.e. erode the eyes, nose and mouth) because if critical regions are totally removed the algorithm cannot recover.

At the end of this process, we have a face region which may be smaller than the actual face, but which contains a connected pixel group without holes including the eyes, nose, and mouth. Traditional 2D methods have a greater difficulty segmenting the person from the background because they do not have the advantage of using invalid range regions or jump edges. This segmentation is important because it limits the nose search region and reduces potential errors dramatically.

3.3 Eye and mouth map refinements

A number of refinements are made to the eye and mouth maps to reduce false alarms. These refinements include narrowing the search region for perspective eye and mouth features and filtering to produce a higher signal in the eye and mouth regions of interest. The search for potential eye locations is refined by eliminating the bottom 30% and top 10% of the *EyeMap* region since the eyes should not be located in the very top or bottom of the face region. Figure 2.a shows an example of this result. Removing the bottom 30% of the data may remove the mouth; however, this does not effect eye detection. Subsequently, we run a smoothing mask (Figure 2.b), minimum filter (Figure 2.c), and a z-based filter (Figure 2.d) over the *EyeMap* in that order to enhance the region around the eyes, and hopefully attenuate the non-eye regions.

a. EyeMap b. Smoothing c. Minimum Filter d. Z-Filter

Figure 2. EyeMap process

The z-based filter adds a 40% weight to pixels on the *EyeMap* which have an invalid range value associated with them. On typical range scans holes appear in the pupils because they are transparent and absorb light. This causes the *EyeMap* to appear even brighter in the pupil

region and allows for a more precise identification of the exact center of the eyes. Using formulas from [8] we produce the initial grayscale *MouthMap* which we then improve with a smoothing filter.

3.4 Facial feature detection

Potential eye candidates are identified by searching the *EyeMap* for the highest N values with a minimum threshold distance from each other (in our experiments N=7). The minimum distance threshold is important because we are looking for distinct eye candidates, thus multiple feature candidates that correspond to the same underlying eye feature would provide no additional value. Thus, it is sufficient to find the highest value on each eye. After identification of these potential features, we search for eye pairs based on the distance between eye features. For instance, a potential pair with a feature to feature distance (measured by the XYZ distance) that is within the range of possible eye distances is considered a valid eye pair.

Once the valid eye pairs are detected the process of finding an associated mouth value is greatly simplified. The approach simply looks for the *MouthMap* value in the corridor running perpendicular from the midsection of the line formed by the eye pair. At about distance D down we search for a potential mouth location, where distance D represents the approximate distance that we expect to find the mouth.

The detection of the eye and mouth features forms a triangle that can be used to find the nose location with greater accuracy. This triangle not only provides a well defined space in 2D in which a nose should be detected, but it also defines a 3D plane indicating the 3D orientation of the face. Thus, we have a defined direction in which the nose should have maximum extent in 3D. We expect this to be extremely robust for performing nose detection when the face is in various orientations. The function simply examines face pixel range values in the interior of the eye-mouth triangle for the point farthest away from the eye-mouth plane. This point is believed to be the nose since the nose extends the farthest from the face.

At this stage we have a potential eye, nose, and mouth location. However, while the eyes and the nose use range information, the mouth has been defined based solely on the *MouthMap*. As a result, identification of the exact center of the mouth is not precise. Thus, we refine the mouth location purely geometrically based upon the eyes and nose locations to obtain a more precise location than the *MouthMap* alone would produce.

After feature detection a confidence metric is computed which analyzes the distances between the four feature points and compares them with the ideal distances. The feature group with the highest confidence value is returned as the believed eye-mouth-nose locations. This approach is discussed in detail in the next section.

3.5 Confidence estimation

The confidence metric is a geometric-based measure that incorporates ideal and measured feature location distances to provide an indication of feature group feasibility. The confidence metric has been extremely useful in 1) determining the most plausible feature group when many feature groups are considered for a given face and 2) determining when a final feature group selection may be incorrect. The confidence metric C is slightly altered to deal with 1) and will be explained in the following.

Table 1. Measured ideal distances

	Ideal Distance	Standard Deviation	Confidence Weight W_i
Eye to Eye	62.4 mm	3.7 mm	3
Eye to Mouth	72.5 mm	4 mm	2
Eye to Nose	57.5 mm	4 mm	6
Nose to Mouth	33.3 mm	.5 mm	1

A core component of C is the ideal distances between feature locations. For our analysis, these distances are computed from 148 different range scans of individuals separated from the testing set used in the results. In lieu of manually determining feature locations, we ran these images through our algorithm and verified the correctness of the detected features by visual inspection. Table 1 represents the average/ideal distances D_i and the standard deviation σ_i between the correct eye-nose-mouth pairings.

The measured feature locations $\hat{D}_i$ are determined by using the 3D coordinate associated with the detected pixel location (computed from the range image). However, in some cases, pixels do not have valid range data associated with them (especially around the eyes) because of the pupils, sensor drop-outs, etc. In this case the algorithm will traverse down the column of pixels for the closest valid range value and use the associated 3D coordinate for the distance measurement.

The confidence metric incorporates a weight W_i for each feature to feature confidence C_i. Each W_i is determined by our observation of the reliability of certain feature locations. For example, the mouth is refined based purely upon the geometric distance down from the tip of the nose. Therefore, no additional value is gained by giving it a high geometric confidence weight since it was determined using entirely geometric means. As a result we give any distance measurement to the mouth a weight of one. Since the eyes and nose are more reliable

each calculation involving both of them is given a weight of 3. These weights are shown in Table 1.

The geometric confidence measure is given in equation (2) where C_i represents the confidence of the *ith* feature to feature distance. *EM* is a parameter indicating the weight of a potential eye candidate in the *EyeMap* and is included in *C* since higher *EyeMap* values are highly correlated with being the true eye locations.

$$C_i = \frac{\left|\hat{D}_i - D_i\right|}{\sigma_i} \quad \text{and} \quad C = EM + \sum_{i \in F} W_i C_i \quad (2)$$

Equation 2 is used as the confidence metric for the selected feature group; however, modifications are required when examining potential feature group candidates. First, D_i is slightly decreased to discourage the eyebrows from being detected as eyes (a common error). In addition, we impose a symmetry term that indicates that eye to nose distances should be the same for both the left and right eyes. This discourages eye to eyebrow pairings which can be problematic. Despite these additions, it is still possible to obtain eyebrow pairs with a higher *C* than the true eye pairs. This is because ideal distance estimates D_i are used in *C* while different faces are different sizes. In the event that *C* is extremely close for two different feature groups, we take the pair of potential eyes that are closer to the center of the face. This allows us to eliminate the mistaken identity of eyebrows for eyes when *C*'s are extremely close.

Table 2. Performance on high and low quality datasets

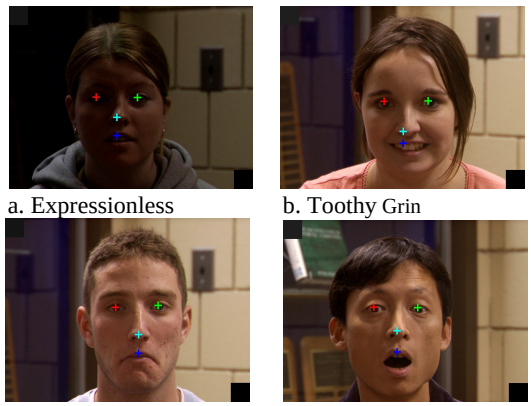
Dataset	High Quality	Low Quality
Correct	99.61%	82.33%
One Error	0.06%	4.10%
Wrong	0.00%	6.31%
Skipped	0.32%	7.26%

4 RESULTS

We ran our experiment on two different datasets from the University of Notre Dame [17] acquired using a Minolta Vivid 910 range scanner that produces a 640X480 range and corresponding color image. The first dataset contains 1,542 high quality images of approximately 300 subjects acquired in the fall of 2003. The database contained a variety of facial expressions (expressionless, smiling, frowning, *etc*.) as is shown in Figure 3. Therefore, the results shown here reflect the algorithm's ability to deal with different expressions.

The second dataset was acquired in the spring of 2003 and is included to show the limitations of the algorithm on very low quality real world data. These scans were taken with studio lights under different environmental conditions. The studio lights caused the Minolta, which is extremely sensitive to the near infrared spectrum for laser visibility, to produce low quality range and color information as seen in Figure 4. This data contains saturated samples with the red component skewed both substantially too high and too low. Also, the range images incorrectly identify the eyes as being extremely recessed into the face as seen in Figure 4.d. Furthermore, facial size tends to be smaller in the second dataset, so there are fewer pixels on the face, which may make feature detection more difficult.

Our algorithm reads in a standard Minolta Vivid 910 dataset. The algorithm has very low memory usage (around 68k) and takes approximately 4 seconds per image on a 3.2 GHz Pentium 4.



a. Expressionless b. Toothy Grin

c. Frowning Sample d. Open Mouth/Alarmed

Figure 3. Different expressions from sample image in the Notre Dame fall 2003 database

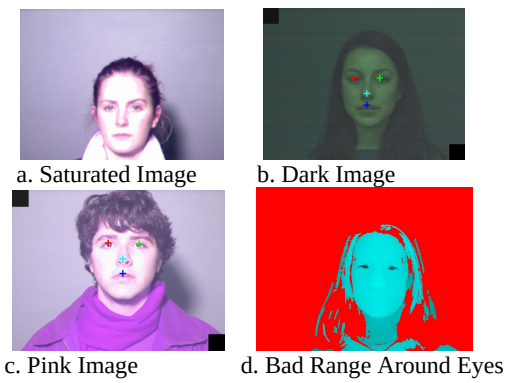
4.1 Real world results

For the first dataset all 1,542 images were processed and produced the results shown in the second column of Table 2. The samples shown in Figure 3 are representative of the images analyzed by our system. The color can be fairly dark, especially around the eyes, because no additional lighting other than overhead fluorescent lighting was used during capture and the quality of the RGB as shown here for our good dataset is acceptable but not optimal.

For this dataset, the correct features were detected in 99.6% of the faces. However, 0.3% of the images were rejected and skipped by the algorithm due to hair obscuration of the eyes, which resulted in incorrect identification of both eyes. One of the images had an eyebrow identified instead of the eye, which appears in the one error category. The one error category means that one of the four detected features was found in the incorrect location. We separate out this type of error as we feel it should be recoverable in later stages of a complete system. Other than this there were no errors in

the reported features located.

We also evaluated the algorithms performance using the second dataset (see Figure 4) and produced the results shown in the 3rd column of Table 2. For this dataset the percentage of correct classifications went down considerably. We attribute this to the combination of poor color quality, incorrect range values, and a reduced number of pixels on the face. This is to be expected and we believe standard 2D algorithms would not operate very well on these images either. Image saturation contributed to the majority of skipped images. The single errors were mostly eyebrow mistakes due to the incorrect or non-existent range values around the eyes which caused eyebrows to be viewed as better eye candidates than the eyes themselves. Finally, the completely incorrect feature detections were usually due to a combination of bad range and saturated pink images.



a. Saturated Image b. Dark Image

c. Pink Image d. Bad Range Around Eyes

Figure 4. Sample images from the spring 2003 ND database

4.2 Robustness experiments

After analyzing limitations on the low quality data (the second dataset), we sought to quantify the algorithms robustness. To accomplish this we took 513 of the images from high quality ND dataset and modified them to vary color bias/saturation, image resolution, and range error.

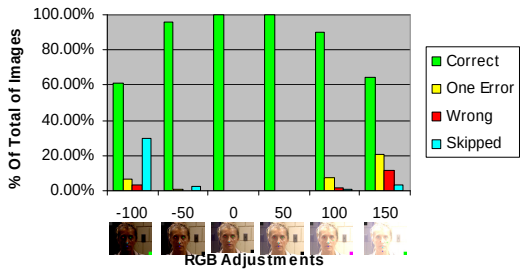


Figure 5. Affect of increasing or decreasing RGB

4.2.1 Color bias robustness

First, we biased the color components of the input image by increasing or decreasing all three RGB components simultaneously to produce the results shown in Figure 5. We discovered that there is a large (150 out of 255) pixel value range in which the algorithm was able to effectively deal with color bias. It is not until the image becomes extremely saturated (where large parts of the face are either solid black or white) that the algorithm's performance begins to degrade.

Next, we varied just the red color component to produce 'pinker' images similar to the low quality Notre Dame data shown in Figure 6. These images are expected to cause more difficulties for the algorithm. While the results still showed a large region (between -50 and 100) in which the algorithm performed well, there is a drop off at 150 as opposed to the more gradual one in Figure 5.

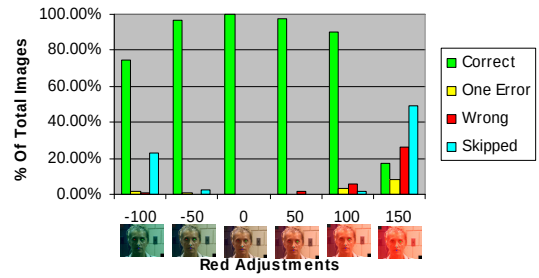


Figure 6. Affect of increasing or decreasing only the red component

4.2.2 Range error robustness

While our algorithm seems fairly robust to color variation, the impact of range distance errors required a separate test. In order to reproduce errors in a range scan, the z coordinate of each point was varied by plus or minus the values shown on the x axis of the graph in Figure 7. The range errors have no significant impact on performance until the error is approximately 20 mm. At this point about 40% of the feature detection errors are in the one error category. These errors are not eye errors, as were common in previous experiments, but mouth detection errors. The calculated ideal distance between the nose and mouth is 33.3 mm. Once the error between two points is greater than this, the nose mouth distance refinement function misidentifies points on the nose (potentially one pixel away from the nose location) as being a potential mouth location.

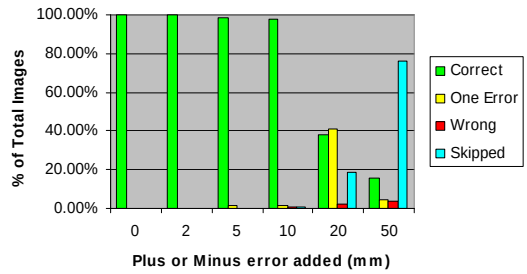


Figure 7. Affect of adding error to the z axis

Furthermore, because the nose detection algorithm looks for the point that is farthest from the defined plane, the detected nose point will likely have the greatest error associated with it because the normal to that plane is roughly aligned to the z axis which is where we introduced the error. Overall the method deals extremely well with range errors up until the error exceeds the ideal distances calculated. Algorithm performance could be improved by pre-processing the range data.

4.2.3 Image resolution robustness

In our final robustness test we reduced the resolution of the original 640X480 image by down sampling (simply selecting intermediate points, no smoothing was applied) to achieve the sizes shown in Figure 8. These results demonstrate our ability to deal with images of a wide range of sizes getting down to very small images in which the eye may only be a few pixels across. Even when the images are very small (58X43), feature detection errors in the one error category were caused by the eyebrow being selected instead of the eye which in most cases meant it was off by about 2-3 pixels. From these results we find that we can deal with low image resolution .

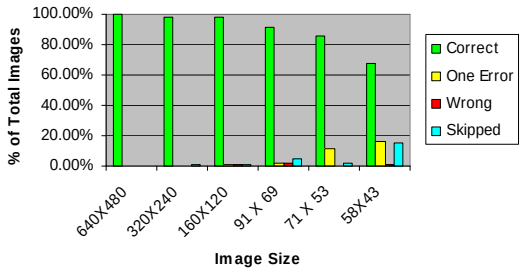


Figure 8. Performance of different image sizes

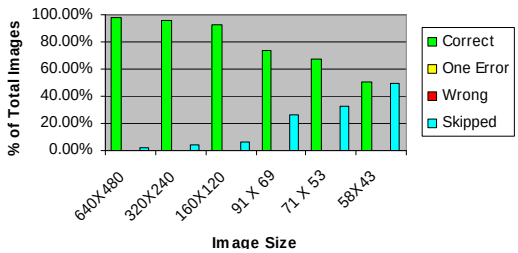


Figure 9. Performance of different image sizes where low confidence images are moved to skipped

4.3 Confidence measure

As discussed previously a confidence value associated with each feature group is returned as well. Up until Figure 9 we have only discussed the results regardless of the confidence of the feature group. Figure 9 assumes that the features returned with the lowest confidence from Figure 8 should be skipped, and automatically moves the lowest confidence images into the skipped category. This allows the removal of all of the feature detection errors for this example. However, it also moves some of the correctly classified results to the skipped column. This may be a desirable way to treat images if one desires a very conservative algorithm that ensures the detected features are correct. A different feature identification method could be used in the case of the skipped images to correctly detect features of interest.

4.4 Impact of 3D algorithm components

In this paper, a multi-modal approach for feature detection has been presented. However, we are interested in assessing the performance benefits that the 3D additions provide, and identifying those improvements that have the biggest impact on performance. To assess this, we analyze the performance impact of three components of our algorithm: z-filtering, z-based erosion, and 3D distance calculations for the confidence function.

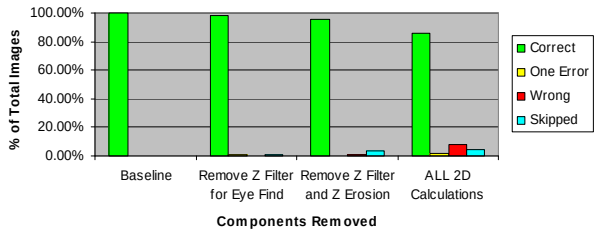


Figure 10. Impact of 3D algorithm components

In Figure 10 one can see the gradual degradation in performance as various 3D aspects are removed until ultimately in the "All 2D Calculations" category range data was not used at all (other than in the initial background segmentation). For the 2D case, the confidence estimator and all other calculations were modified to only use pixel coordinates. For instance, the 2D confidence estimator is the same as for 3D except ratios of pixel distances were used. In Figure 10, the removal of 3D components has only a moderate impact leading to an 86% accuracy once all 2D calculations are performed. This is consistent with the performance of known 2D implementations of a similar design to this algorithm.

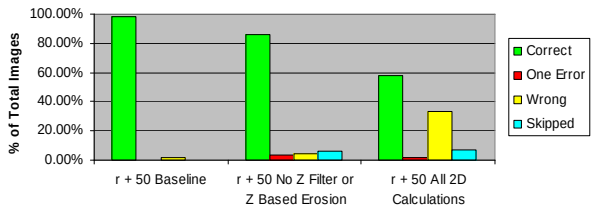


Figure 11. Removing components for the r+50 test

One of the advantages of this algorithm is its robustness. It can compensate for poor quality data by its combined use of 2D and 3D data. We ran the "r + 50"

test from the robustness section where all red values were increased by 50 for the "no z-filter" or "no z-erosion" and "All 2D Calculations" versions of the algorithm from Figure 10. The result is shown in Figure 11. The full 3D version of the algorithm with all of its 3D components performs at 98% correctness in the "r+50" test. However, the "All 2D Calculations" which relies solely on the color to determine eye, nose, and mouth location fails at this point with a 58% correctness for the same r+50 test. This demonstrates one of the advantages of our method; its multi-modal characteristics allow it to overcome in situations where a single modal method such as "All 2D calculations" fails.

5 CONCLUSIONS/FUTURE WORK

We have presented a multi-modal approach for facial feature detection that can achieve a 99.6% correct identification rate for high quality images. We have demonstrated the benefit of using 3D information in combination with 2D over using 2D information alone. The benefits of the multi-modal approach become extremely apparent when the quality of either the 2D or 3D representation is compromised as demonstrated in our robustness experiments. Another notable attribute of our approach is the geometric-based confidence measure which enables the selection of the correct feature group out of many potential groups and allows us to determine when the final feature group selection may contain feature detection errors.

Although the results presented in this paper represent an initial analysis of the multi-modal feature detection approach, we are extremely encouraged that the algorithm is robust and its performance competitive with current commercial feature detection systems. For instance, Wang and Flynn [1] analyzed the performance of a commercial program called Faceit on a dataset taken side by side with the high quality ND dataset (images were acquired within one minute of each other). Although, this database consisted of higher quality images than used in our test and the algorithm was only capable of identifying the eyes, correct feature detection was achieved for only 96.1% of the faces. We feel we can fairly compare their 2D results with our success of 99.6% correct classification rate.

Further, we have improved upon Hsu and Jain's [8] work. For the most comparable database they managed a 90% detection rate. The best rate they were able to achieve on all of their databases was 97% and their database sizes were significantly smaller than those used in this paper. We managed to improve upon this significantly.

Next steps include analyzing the algorithm with variations in facial pose and optimizing the approach for this case. Further testing may involve the utilization of other types of scanners and expanding our capability to deal with profile shots.

REFERENCES

[1] H. Wang and P. Flynn, "Studies on sensitivity of face recognition performance to eye location accuracy", *PAMI* under Review 2004.

[2] http://www.identix.com/

[3] Y. Ryu and S. Oh, "Automatic extraction of eye and mouth fields from a face image using eigenfeatures and multilayer perceptrons," *Pattern Recognition*, vol. 34, 2001, pp. 2459-2466.

[4] C. Chen and C. Huang, "Human face recognition from a single front view," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 6, 1992, pp. 571-593.

[5] G. Chow and X. Li, "Towards a system for automatic facial feature detection," *Pattern Recognition*, vol. 26, 1993, pp. 1739-1755.

[6] K. Lam and H, Yan, "Locating and extracting the eye in human face images," *Pattern Recognition*, vol. 29, 1996, pp. 771-779.

[7] J. Huang and H, Wechsler, "Eye detection using optimal wavelet packets and radial basis functions (rbfs)," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 13, no. 7,, November 1999, pp. 1009.

[8] R. Hsu and A. K. Jain, "Face detection in color images", *PAMI*, vol. 24, no. 5, May 2002, pp. 696-706.

[9] K. Lam and H. Yan, "An analytic-to-holistic approach for face recognition based on a single frontal view", *PAMI*, July 1998.

[10] T. K. Leung, M.C. Burl, and P. Perona, "Finding faces in cluttered scenes using random labeled graph matching", *ICCV*, 1995, pp. 634-644.

[11] E. Hjelmas and J. Wroldsen, "Recognizing face from the eyes only," *Scandinavian Conference on Image Analysis*, 1999.

[12] K. Wong, K. Lam, and W. Siu, "An efficient algorithm for human face detection and facial feature extraction under different conditions", *Pattern Recognition Letters*, 2001.

[13] X. Lu, D. Colbry, and A. K. Jain, "Three dimensional model based face recognition," *ICPR*, Cambridge UK, August 2004.

[14] G. Gordon, "Face recognition based on depth and curvature features", *CVPR*, 1992, pp. 108-110.

[15] J. Xiao, S. Baker, I. Mathews, and T. Kanade, "Real-time combined 2D + 3D active appearance models", *CVPR*, June 2004.

[16] Y. Wang, C. Chua, Y. Ho, "Facial feature detection and face recognition from 2D and 3D images", *Pattern Recognition Letters*, vol. 23, no. 10, August 2002, pp. 1191-1202.

[17] http://www.nd.edu/~cvrl/UNDBiometricsDatabase.html

[18] K. Sobbottka and I. Pitas, "Face localization and facial feature extraction based on shape and color information", *ICIP*, vol. 3, 1996, pp. 483-486.