Proc. VIIth Digital Image Computing: Techniques and Applications, Sun C., Talbot H., Ourselin S. and Adriaansen T. (Eds.), 10-12 Dec. 2003, Sydney

Robust Face Localisation Using Motion, Colour & Fusion

Darren Butler, Chris McCool, Matthew McKay, Scott Lowther, Vinod Chandran, and Sridha Sridharan *

Speech, Audio, Image and Video Technologies Program
Faculty of Built Environment and Engineering
Queensland University of Technology
GPO Box 2434, Brisbane QLD 4001, Australia
http://www.bee.qut.edu.au/research/prog_saivt.shtml

Abstract. Face detection is fundamental for several important applications, such as, face recognition and human-computer interaction. This paper details a novel hierarchical face detector which combines motion, colour and late fusion. Motion segmentation is employed to eliminate background clutter and to reduce the initial search space. Subsequently, skin segmentation is used to determine a candidate face. Five simple eye-detection algorithms are fused to robustly localise the eyes within the candidate. Fusion is beneficial because the large amount of variation within the face makes it difficult for any individual technique to perform well under all conditions. The resulting system is capable of localising faces from still images in real-time with an accuracy of 93.75%.

1 Introduction

Since the events of September 11, biometric research has received greater emphasis. Current biometrics include: fingerprints, palm-prints, hand geometry, gait, speech patterns, facial features, iris patterns and DNA. Currently, iris recognition is the most robust of the commercially viable biometrics with an equal error rate of 1 in $131,000$ or 0.0008% [1]. However, accurate images of the iris are required and hence, the subject must cooperate. Face recognition has grown in popularity because it is not obtrusive, does not require the subject to cooperate and most importantly, people seem willing to accept it. There has been a plethora of approaches proposed in the literature including, amongst others: Fractal codes [2], Principal Component Analysis (PCA) [3], Independent Component Analysis (ICA) [4], and Linear Discriminant Analysis (LDA) [5]. Much of our recent work was inspired by the original eigen-face technique of Turk and Pentland [3].

Clearly, for there to be any hope of recognising a face, it must first be located correctly. If the face cannot be found or it is poorly localised then recognition will undoubtably fail. In fact, misclassifications are often the result of failed localisation and not deficiencies in the recogniser. Variations in pose, scale, illumination

* This project was funded by an Office of Navy Research (ONR) grant.

and facial expression are the primary sources of difficulty. However, one problem that is often overlooked, is computational complexity. Face detection alone is not especially useful. Further analysis is required in order to recognise the face, compress it efficiently or interact accordingly. Therefore, it is imperative that the face localiser consumes as few processor cycles as possible. Its importance, not just for face recognition, has ensured that face detection has developed into a field study in its own right, a survey of which was recently conducted by Yang *et al.* [6].

In this paper, we propose a novel face detection scheme which fuses several efficient eye-detection algorithms to create a robust system with faster than real-time performance. We demonstrate that accurate results can be achieved by combining a few very simple heuristics. The remainder of the paper is organised as follows: Section 2 outlines the face detection algorithm and describes each stage in detail; Section 3 demostrates the different detectors in isolation and contrasts them with the fused approach; finally, Section 4 gives our conclusions and describes future research directions.

2 Algorithm

Our face detection algorithm follows a conceptually simple hierarchical procedure. Firstly, motion segmentation is used to separate moving objects from the stationary background. This is important because simple feature detectors, like those that we are using, are often confused by background clutter. It also serves to reduce our search space and hence improves the performance of latter stages. Next, the moving regions are searched for skin tones as human faces consist primarily of skin. Although skin colour is somewhat susceptible to illumination, it is still a very useful feature because it is invariant to pose and scale. Based on the size and shape of the skin regions, at most, a single candidate face region is selected for further processing. However, the algorithm can easily be extended to multiple faces by allowing more than one skin region to be a candidate face.

Within the candidate face region, five different techniques are used to determine the likelihood of each position belonging to an eye. The results of the five techniques are fused to produce a final eye likelihood map. This map is then searched for candidate eye locations and geometry contraints are used to select the best left and right eye pair. If a suitable pair of eyes could be found, the candidate face region is verified as being a face. Finally, the face region is normalised according to the eye locations and the location of the face and eyes are output to the recognition engine. The key phases of the algorithm are described in greater detail in Sections 2.1-2.3.

2.1 Motion Detection

The motion detector that we are utilising was previously developed by us [7]. It relies on the premise that the more often a pixel takes a particular colour, the more likely it is that it belongs to the background. Therefore, at the heart of

the algorithm is a very low complexity method for maintaining some limited but important information about the history of each pixel. To do this, each pixel is modeled by a *group* of K *clusters* where each cluster consists of a weight w_k and an average pixel value called the centroid c_k.

Incoming pixels are compared against the corresponding cluster group. The goal is to find the matching cluster with the highest weight and hence the clusters are searched in order of decreasing weight. A matching cluster is defined as one which has a Manhattan distance between its centroid and the incoming pixel below a user prescribed threshold, T. If a matching cluster could not be found, then the cluster with the minimum weight is replaced by a new cluster having the incoming pixel as its centroid and a low initial weight. Alternatively, if a matching cluster was found, then the weights of *all* clusters in the group and the centroid of the *matching* are adapted accordingly. After adaptation, the weights are normalised so they total to one.

Pixels are classified by summing the weights of all clusters that are weighted higher than the matched cluster. This calculation is simplified by sorting the clusters in order of increasing weight after which, we can employ the following calculation:

$$P = \sum_{k>M_k}^{K-1} w_k \tag{1}$$

The result, P, is the total proportion of the background accounted for by the higher weighted clusters and is an estimate of the probability that the incoming pixel belongs to the foreground.

2.2 Skin Detection

As aforementioned, skin detection is a useful technique for finding faces because it is invariant to pose and scale. Furthermore, it has been shown that regardless of race, the skin colours cluster fairly well in chrominance [8]. It follows that the perceived interracial difference in skin colours depends more heavily on luminance than on chrominance. Hence, by ignoring luminance, non-prejudicial skin detectors can more readily be developed. Ignoring luminance has the additional benefit of suppressing some of the effects of illumination.

Colour can be represented in a multitude of different ways and the optimal choice of representation, or *colour space*, is application dependent. Some colour spaces, like YCbCr, are useful for digital video compression, whereas others, like HSV (Hue, Saturation, Value) are best suited for graphic artists. Similarly, the choice of colour space influences the performance of skin detection. Therefore, following on from [8] and [9] we evaluated a number of popular spaces, including: YCbCr; HSV; TSL; normalised RGB; and the SCT (Spherical Coordinate Transform). An extract of the results our empirical study is contained in Section 3 based on which, we selected the SCT for use in our system.

The skin and non-skin colour distributions were both modeled using a mixture of Gaussians. Modeling both distributions is necessary for Bayesian classification and achieves far superior discrimination than either model in isolation.

Pixels are classified as skin whenever the ratio of the skin and non-skin likelihoods exceeds a threshold derived according to Bayes' theorem using Equation 2.

$$\frac{p\left(x_{sc}|\lambda_{skin}\right)}{p\left(x_{sc}|\lambda_{\overline{skin}}\right)} > \frac{P\left(\lambda_{skin}|x_{sc}\right)\left(1-P\left(\lambda_{skin}\right)\right)}{P\left(\lambda_{skin}\right)\left(1-P\left(\lambda_{skin}|x_{sc}\right)\right)} \tag{2}$$

where

x_{sc} is the source pixel.
$P\left(\lambda_{skin}\right)$ is the *a priori* probability of skin.
$P\left(\lambda_{skin}|x_{sc}\right)$ is the *a posteri* skin class probability threshold.

After skin detection, the largest connected region is chosen as the candidate face, subject to an area constraint. A fuzzy bounding box is then placed around the candidate face and outlying skin blobs are eliminated. Any remaining skin regions determine the bounds of the candidate face. Finally, within these bounds, high intensity pixels are suppressed (by setting them to the mean value) as they are predominantly caused by light reflecting from glasses and detract from eye localisation. A block diagram of the skin detection algorithm can be found in Figure 1.

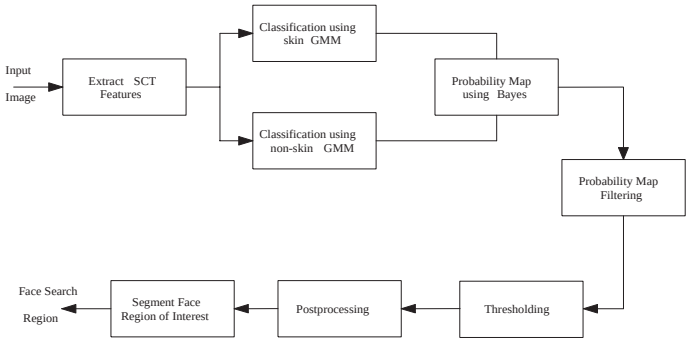

Fig. 1. Skin Detection Algorithm

2.3 Eye Localisation

The previous two phases of the algorithm were designed to progressively reduce the search space, resulting in a single candidate face. The mandate of this phase is to verify or reject the candidate by searching for eyes within it. There are a multitude of techniques that we could have employed to locate the eyes. However, the large variation within and around the eyes makes it difficult for a single technique to perform well in all conditions. As a consequence, many techniques achieve robustness at the expense of computational complexity and even given a limited search space, they would have been prohibitively slow. Therefore, we selected five very simple eye detectors and used late fusion to combine

their results. The hope was that although the detectors performed poorly in isolation, together they would be accurate and robust. Furthermore, since only two of the detector have dependencies, much of the code could run in parallel. The five methods chosen were: gradient image filtering, chrominance differencing; chrominance shifting; eigen-eyes (i.e. PCA) and eye/eyebrow discrimination. Scale tolerance is achieved by applying each method at number of different scales.

The gradient image filtering eye detector attempts to capture the variation that exists in and around an eye. First, the magnitude of the image gradient in YCbCr space is calculated according to Equation 3. The resulting image is then truncated to the range [0, 100] and convolved with a two-dimensional elliptical kernel. The shape of the kernel was chosen because it emphasises the elliptical structure of the eye and its size was estimated from training data. The resulting filtered gradient image is inverted so that eye locations correspond with image minima.

$$G = \left|\frac{\partial Y}{\partial x}\right| + \left|\frac{\partial Y}{\partial y}\right| + \left|\frac{\partial Cb}{\partial x}\right| + \left|\frac{\partial Cb}{\partial y}\right| + \left|\frac{\partial Cr}{\partial x}\right| + \left|\frac{\partial Cr}{\partial y}\right| \tag{3}$$

Face images that are transformed into the YCbCr colour space exhibit a high concentration of blue-chrominance and a low concentration of red-chrominance around the eyes [10]. The chrominance differencing eye detector exploits this observation by convolving the difference between the chrominance channels, $Cr - Cb$ with an elliptical filter. Eyes can then be located by searching for minima in the resulting image. However, eyebrows often also appear as minima and can significantly degrade performance.

Chrominance shifting attempts to suppress the interference caused by eyebrows by shifting the chrominance difference eye map vertically and then subtracting it from itself. Although this is a very simple heuristic, it is useful for the following reasons:

- Eyebrow regions will be subtracted by the corresponding (vertically shifted) eye regions and since both are consistently low, the result will *approximate zero*.
- The eyes will be subtracted by the high-valued skin regions that are located directly below them and will therefore receive *large negative values*.
- Finally, regions of skin will in general be subtracted by other regions of skin and hence will also *approximate zero*.

Thus, the final eye map will be approximately zero everywhere except at the eyes where it will exhibit large negative values. The magnitude of the vertical shift is constant and was derived from training data.

The eigen-eyes detector is derived from the *eigen-face* technique of Turk and Pentland [3]. Eigen-eyes are simply the principal components of a distribution of eye images or equivalently, the eigenvectors of the covariance matrix that is formed by treating the images as vectors. Low energy noise is suppressed by only retaining the eigenvectors which correspond to the M largest eigenvalues. Eye images can be efficiently approximated by a linear combination of eigen-eyes.

However, non-eye images cannot accurately be recovered after projection into the subspace. Therefore, the reconstruction error or *distance from feature space* (DFFS) is a measure of how similar an image is to an eye.

Rudimentary eye detectors, like those we are using, are often confused by eyebrows and falsely classify them as eyes. The goal of eye/eyebrow discrimination is to counteract this effect. Individual eye and eyebrow eigen-spaces are first created and then combined into a single orthonormal basis according to the Gram-Schmidt process [11]. If LDA is applied directly to a two-class problem such as this, then the resulting *discriminant space* (DS) will have a dimensionality of one. Since neither the eye or eyebrow classes can be adequately modeled by single Gaussians, intra-class clustering is used to form 8 eye and 16 eyebrow pseudo-classes [12]. This effectively transforms the two-class problem into a 24-class problem and hence LDA produces a 23-dimensional DS. Within the DS, the eye and eyebrow densities, $p\left(y|\lambda_{eye}^{\{DS\}}\right)$ and $p\left(y|\lambda_{eyebrow}^{\{DS\}}\right)$ are approximated using GMMs and as before, Bayes' theorem can be used to estimate the probability of pixels belonging to either class.

After each detector has completed, the resulting eye maps are normalised to the range [0, 1] and are fused using a weighted sum with empirically derived weights, an efficient and effective fusion technique [13]. Candidate eyes are located for each scale by searching within the fused eye map. Finally, geometry contraints and *a priori* knowledge are exploited to determine the pair of candidates that most likely correspond with the true eyes. If a suitable pair of eyes is not found then the candidate face is rejected. Otherwise it is said to be verified and is passed on to the recogniser along with the eye locations. The entire eye localisation algorithm is given in Figure 2

3 Results

The motion detector was tested independently from the remainder of the system as very few face databases contain useable video. A standard 1.8 Ghz Pentium 4 computer with 512 Mb of RAM and running Redhat Linux 7.3 (kernel 2.4.20) was used to collect the results. The choice of camera was limited to those with manual white balance and manual gain control from which we selected the Bosch 1153P analog security camera. Under this configuration, our algorithm proved capable of segmenting 320×240 PAL video at full frame rate, using only 35% of the CPU. Some typical segmentation results can be found in Figure 3. Empirical testing was conducted to determine the optimal colour space to use for skin/non-skin segmentation. The skin colour models were trained from manually segmented skin regions of 156 images (39 individuals, 4 images per person) taken from the XM2VTS [14] database. The non-skin models were constructed from 179 images from a natural imagery database that was collected in-house. Based on our evaluation, the Spherical Coordinate Transform provided the best results and was selected for use in our algorithm. Table 3 details the four best performing colour spaces and the optimum GMM orders.

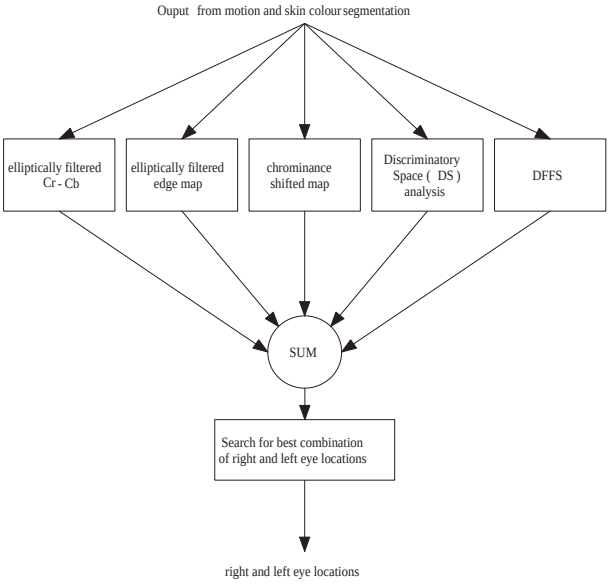

Fig. 2. Eye Localisation Algorithm

Fig. 3. Segmentation results: (a) Original; (b) Moving Objects.

Colour Space	Skin GMM Order	Non-Skin GMM Order	Equal Error Rate (%)
SCT	2	15	5.18%
YCbCr	2	4	5.5%
RGB	4	20	5.56%
Normalised RGB	6	2	6.02%

Table 1. Colour Spaces for Skin Segmentation

The face detection algortihm (consisting of skin colour segmentation and eye localisation) was tested on the XM2VTS database. Each eye localisation algorithm was tested in isolation to one another (the skin colour segmentation algorithm was always present). Combinations of the eye localisation algorithms were then tested and finally all the eye localisation algorithms were tested together. The output of each eye localisation algorithm was an eye map image, where the minima represent the most probable left and right eye locations. Weighted summation was used to fuse each algorithm and equal weights for the fusion was the default. The best (lowest) minima for the output from three scales was used as the final eye locations. The output of each of the separate systems can be seen in Figure 4. The eyes were correctly localised if they satisfied Equation 4 [15].

$$e_{eye} = \frac{max(d_l, d_r)}{d_{eye}} \tag{4}$$

where

d_{eye} is the distance between the two true eye centers.
d_l is the distance between true and estimated left eye position.
d_r is the distance between the true and estimated right eye position.

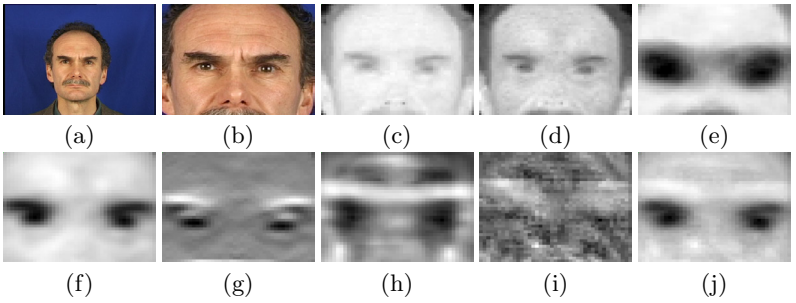

Fig. 4. (a) Original Image; (b) ROI; (c) Cb; (d)Cr; (e) Gradient Filtering; (f) Cr-Cb; (g) Shifted Cr-Cb; (h) DFFS; (i) DS; (j) Final Eye Map

The tests using the XM2VTS database were based on Configuration I. The training data consisted of all Training subsets defined within the Clients set (200 individuals, 600 images). The testing was conducted using the full Test subset defined within the Impostors set (70 individuals, 560 images). The results for the various combinations of eye localisation algorithms can be found in Table 2.

Test	Correct Left and Right Eye Positions (%)
DS Only	31.43%
Chrominance Difference Only	65.89%
Gradient Filtering Only	66.43%
Chrominance Shift Only	74.46%
Full system minus DS and DFFS	85.18%
Full system minus DFFS	87.69%
Full system minus Chrominance Shift and DS	91.07%
Full system minus DS	92.5%
Full system minus Chrominance Shift	92.5%
Full system	93.75%

Table 2. Face Detection Results

4 Conclusions

The face detection system presented in this paper demonstrates that the fusion of multiple eye localisation algorithms can result in a more robust system. The most accurate standalone eye localisation algorithm when used in isolation has an accuracy of 74.46% which is far lower than the 93.75% of the full system. This level of accuracy was obtained from an equally weighted summation of the eye detector outputs.

Future work will involve determining the optimal weights for weighted summation fusion. In addition, the potential of other eye localisation algorithms will be investigated and if they prove beneficial they will be included in future iterations of the system. It is also possible that detecting other facial features such as the nose and mouth will allow more rigid geometry constraints to be used when verifying candidate faces. Finally, an infrared pupil detector is currently being developed as a replacement for the eye detectors for certain applications.

References

1. Daugman, J.G.: High confidence visual recognition of persons by a test of statistical independence. IEEE Transactions on Pattern Analysis and Machine Intelligence **15** (1993) 1148–1161
2. Ebrahimpour-Komleh, H., Chandran, V., Sridharan, S.: Face recognition using fractal codes. In: Proceedings of the Third Australian Workshop on Signal Processing and Applications. (2000)
3. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience **3** (1991) 71–86
4. Jiali, Z., Jinwei, W., Siwei, L.: Face recognition: a facial action reconstruction and ica representation approach. In: Proceedings of the 2001 International Conferences on Info-tech and Info-net. Volume 3. (2001) 456–461
5. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenface vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence **19** (1997) 711–720

6. Yang, M.H., Kriegman, D.J., Ahuja, N.: Detecting faces in images: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **24** (2002) 34–58
7. Butler, D., Sridharan, S., V. M. Bove, J.: Real-time adaptive background segmentation. In: Proceedings of ICASSP '03. (2003)
8. Terrillon, J.C., Akamatsu, S.: Comparative performance of different chrominance spaces for color segmentation and detection of human faces in complex scenes. In: Proceedings of Vision Interface 99. (1999) 180–187
9. Butler, D., Sridharan, S., Chandran, V.: Chromatic colour spaces for skin detection using gmms. In: Proceedings of ICASSP '02. Volume IV. (2002) 3620–3623
10. Hsu, R.L., Abdel-Mottaleb, M., Jain, A.K.: Face detection in color images. IEEE Transactions on Pattern Analysis and Machine Intelligence **24** (2002) 696–706
11. Anton, H.A., Rorres, C.: Elementary Linear Algebra. 7th edn. John Wiley and Sons, Inc., New York (1994)
12. Lucey, S.: Audio-Visual Speech Processing. PhD thesis, Queensland University of Technology (2002)
13. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence **20** (1998) 226–239
14. Messer, K., Matas, J., Kittler, J., Luettin, J., Maitre, G.: XM2VTSDB: the extended M2VTS database. In: Proceedings of AVBPA '99. (1999)
15. Jesorsky, O., Kirchberg, K., Frischholz, R.: Robust face detection using the hausdorff distance. In: Proceedings of the Third International Conference on Audio and Video based Biometric Person Authentication. (2001) 90–95