# EE368 Project
# FACE DETECTION AND GENDER RECOGNITION

Michael Bax, Chunlei Liu, and Ping Li

26 May 2003

## 1 Introduction

The full face detection and gender recognition system described here is made up of a series of components connected in both serial and parallel, as illustrated in Figure 2. The successive stages are explained in detail in the body of this report.

## 2 Colour-based segmentation

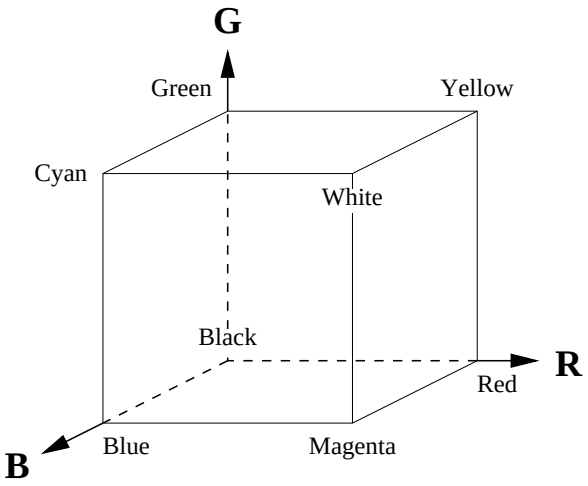
The first step in this face detection algorithm is that of colour segmentation. The goal is to remove the maximum number of non-face pixels from the images in order to narrow the focus to the remaining predominantly skin-coloured regions.

### 2.1 Colour space selection

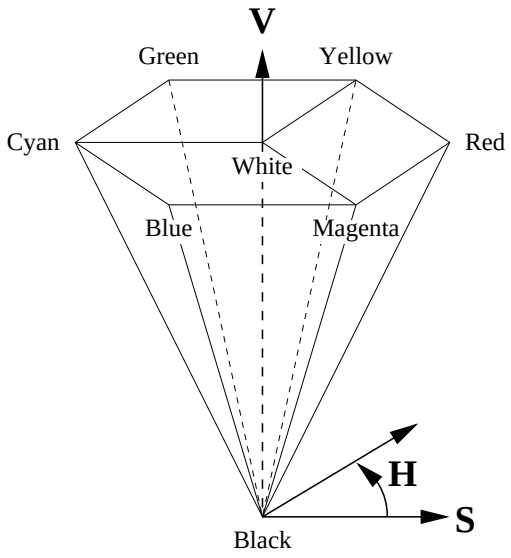
The first step in performing colour-based segmentation is choosing an appropriate colour space in which to operate from the wide variety of choices such as RGB, HSV, CMYK, YCbCr, etc [1]. Of these, RGB (red-green-blue) and HSV (hue-saturation-value) have been the most widely used. Figure 1 illustrates the geometries of the two spaces.

By way of example, HSV representation has certain advantages over RGB when it comes to face detection. As Garcia *et al* [2] note, skin colours are sensitive to the lighting condition. In the RGB space, each of the three components may exhibit substantial variation under different lighting environments. In HSV space, however, the hue and saturation components are virtually unchanged.

Figure 3 shows the histograms of RGB component values of both face and non-face pixels over all seven training input images. Similarly, Figure 4 shows the histograms of the same images in HSV space, where the S component and in particular the H component are well-clustered for face-pixels, while H and S are spread over a wide range for the remainder of the image. This observation favours using an HSV colour space if only a simple thresholding colour segmentation is desired.




(a) RGB model




(b) HSV model

Figure 1: The RGB and HSV colour models.

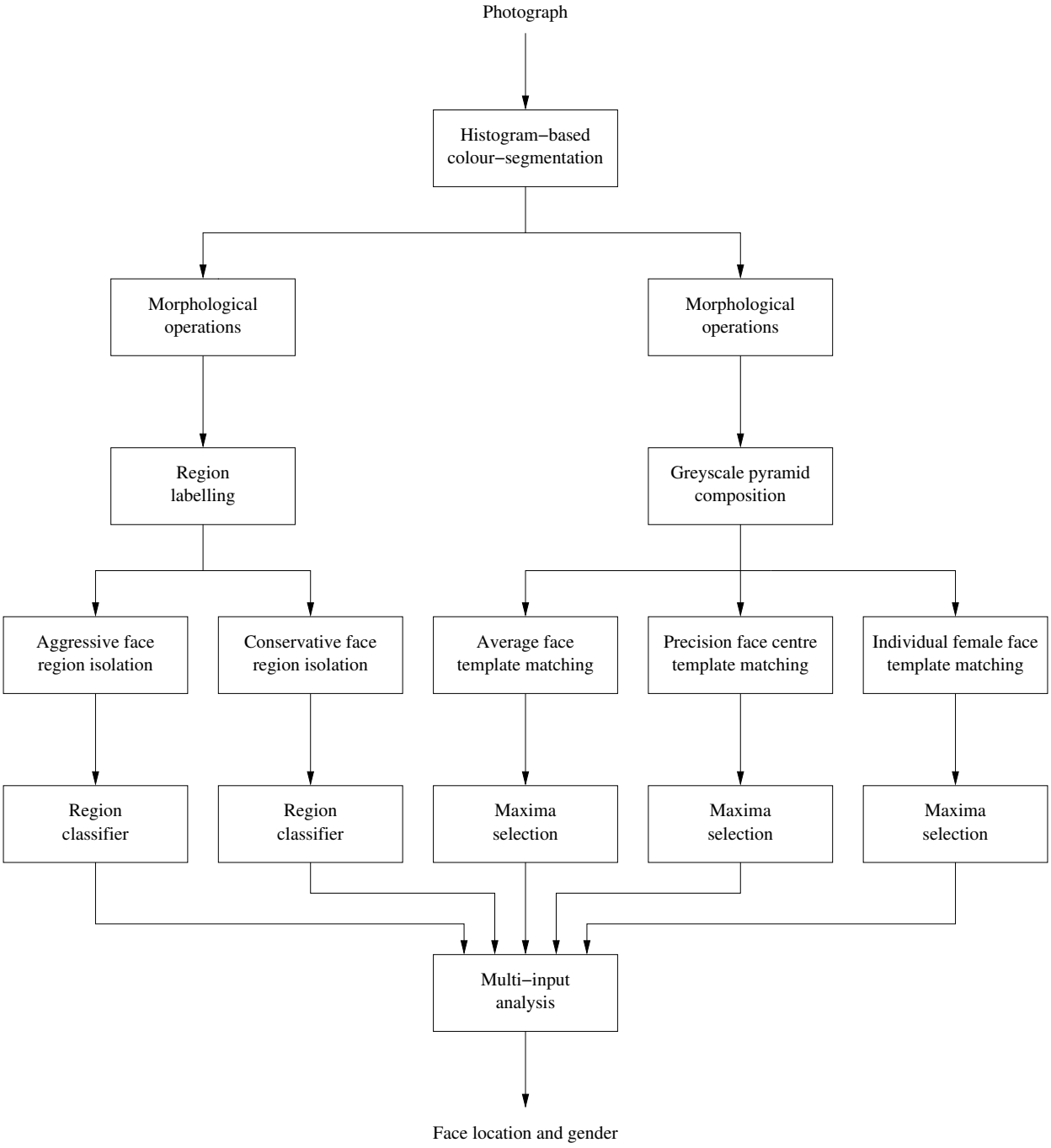



Figure 2: The face detection system schematic.

The same is true of the YCbCr colour-space, but the clustering does not quite as readily lend itself to projection onto the axial dimensions, and hence requires a somewhat more complicated 2D basis for segmentation.

Manual segmentations of this type suffer as a consequence of approximating a complex bounding region using simple geometric relations. Given that the unknown images were taken in similar lighting and with similar equipment to the training images, a probability-based segmentation may be used.

In this case, the choice of spatial domain is not as significant. However, the fundamental characteristics of a colour space can complicate processing — for example, the non-Cartesian nature of the HSV colour cone indicates that non-uniform quantization may be appropriate.

## 2.2 Probability-based classification

Due to the large number of pixels in the training images, there is enough data to create a reasonable estimate of the underlying probability density functions for both face and non-face skin colours. Let

$$f_{\mathbf{X}}(\mathbf{x}|\mathbf{X} \in \Phi)$$

be the colour space probability density function for a pixel vector $\mathbf{X}$ in the set of face pixels $\Phi$; the vector components $X_i$ represent the colour components — R, G and B in this case. Similarly, let

$$f_{\mathbf{X}}(\mathbf{x}|\mathbf{X} \notin \Phi)$$

be the probability density function for non-face skin pixels. These two density functions can be estimated from the empirical distribution of the pixels in the training images.

Conversely, let the probability that a given colour pixel is part of a face be

$$p_{\mathbf{X}}(\mathbf{X} \in \Phi|\mathbf{X} = \mathbf{x}).$$

The Bayesian formula [5] gives

$$\begin{aligned} R &= \frac{p_{\mathbf{X}}(\mathbf{X} \in \Phi|\mathbf{X} = \mathbf{x})}{p_{\mathbf{X}}(\mathbf{X} \notin \Phi|\mathbf{X} = \mathbf{x})} \\ &= \frac{f_{\mathbf{X}}(\mathbf{x}|\mathbf{X} \in \Phi)}{f_{\mathbf{X}}(\mathbf{x}|\mathbf{X} \notin \Phi)} \times \frac{\pi_\Phi}{\pi_{\bar{\Phi}}} \end{aligned} \quad (1)$$

where $\pi_\Phi$ and $\pi_{\bar{\Phi}}$ are the prior probabilities of a randomly-selected pixel falling in a face or the background, respectively. Without further information, the ratio of $\frac{\pi_\Phi}{\pi_{\bar{\Phi}}}$ can be estimated from the ratio of the total number of face pixels to the total number of non-face pixels.

For convenience, Equation 1 may be reformulated as

$$\begin{aligned} \log R &= \log \frac{p_{\mathbf{X}}(\mathbf{X} \in \Phi|\mathbf{X} = \mathbf{x})}{p_{\mathbf{X}}(\mathbf{X} \notin \Phi|\mathbf{X} = \mathbf{x})} \\ &= \log \frac{f_{\mathbf{X}}(\mathbf{x}|\mathbf{X} \in \Phi)}{f_{\mathbf{X}}(\mathbf{x}|X \notin \Phi)} + \log \frac{\pi_\Phi}{\pi_{\bar{\Phi}}}. \end{aligned} \quad (2)$$

If the prior information is known, the classification rule would normally be:

$$\begin{aligned} \mathbf{X} \in \Phi \quad & R \geq R_0 = 1, \textit{i.e.}\ \log R \geq \log R_0 = 0 \\ \mathbf{X} \notin \Phi \quad & \text{otherwise.} \end{aligned}$$

However, the prior information is not necessarily available; conversely, in many situations it is desirable to adjust the classification threshold $R_0$. For example, since in this approach colour-based segmentation is used for subsequent face detection, it is desirable to bias towards misclassifying a non-face pixel as a face pixel rather than the reverse. Consideration of these factors yields the following classification rule

$$\begin{aligned} \mathbf{X} \in \Phi \quad & \log \frac{f_{\mathbf{X}}(\mathbf{x}|\mathbf{X} \in \Phi)}{f_{\mathbf{X}}(\mathbf{x}|\mathbf{X} \notin \Phi)} + \alpha \geq 0 \\ \mathbf{X} \notin \Phi \quad & \text{otherwise,} \end{aligned} \quad (3)$$

where

$$\alpha = \log \frac{\pi_\Phi}{\pi_{\bar{\Phi}}} - \log R_0 \quad (4)$$

is the undecided parameter to be chosen.

A range of values for $\alpha$ for image pixel classification in both RGB and HSV space were evaluated, the results of which are shown in Figure 5. The total classification error is composed of false positive error (mis-classifying non-face pixels as face pixels) and false negative error (misclassifying face pixels as non-face pixels).

The two sub-figures are quite similar. The false negative error increases with $\alpha$, while false positive error decreases with increasing $\alpha$. The total error reaches minimum at $\alpha = 3$. This optimal value of $\alpha$ is supported by the observation that, over all 7 training images, the ratio of the total number of face pixels to non-face pixels is roughly $\frac{1}{16}$; $\log \frac{1}{16} = -2.77$.

Because this colour-based segmentation is only the first step in the larger face detection algorithm, it is best to retain as many face pixels as possible, minimizing false negative error (even at the cost of including additional non-face pixels), leading to the chosen value of $\alpha \approx 2$.

Although the performance differences between RGB and HSV colour space are quite small, RGB is nonetheless superior [3], and that was the colour space chosen for this algorithm.

In practice an empirical histogram derived from manual image segmentation is somewhat noisy. As Figure 6 shows, it has rough edges and isolated bins that are not likely features of the true distribution. In addition, pixel value noise in unknown images may cause marginal pixels to "skip" out of the histogram bin in which they should fall — abrupt discontinuities in the estimated histogram will exacerbate this problem.

The solution is to filter the empirical histogram to smooth out the transients locally, but without losing the

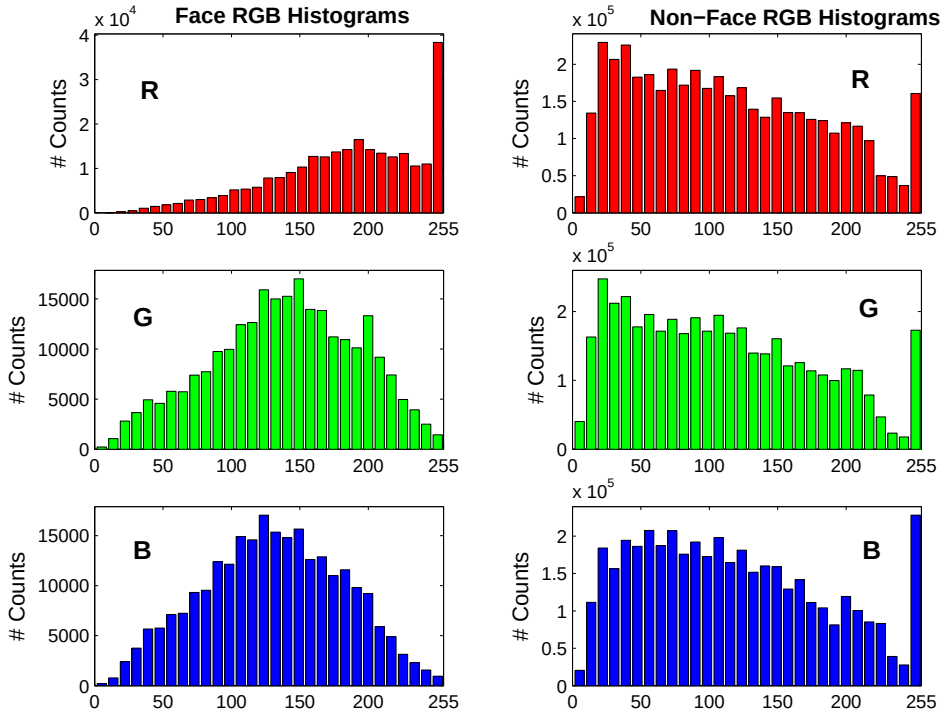



Figure 3: Face and non-face histograms of the RGB colour components, computed over all 7 training images.

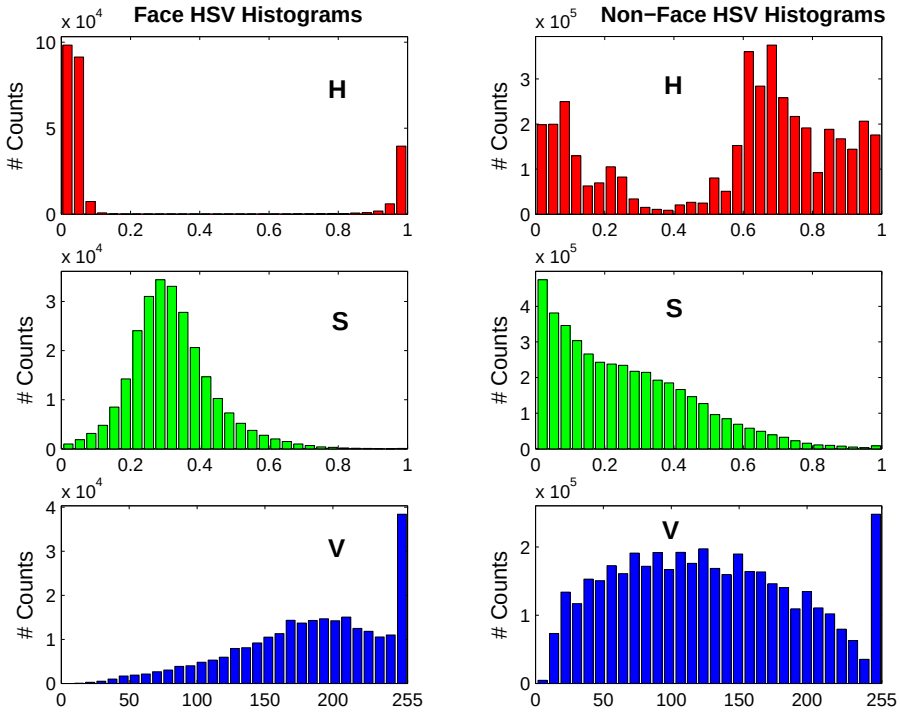
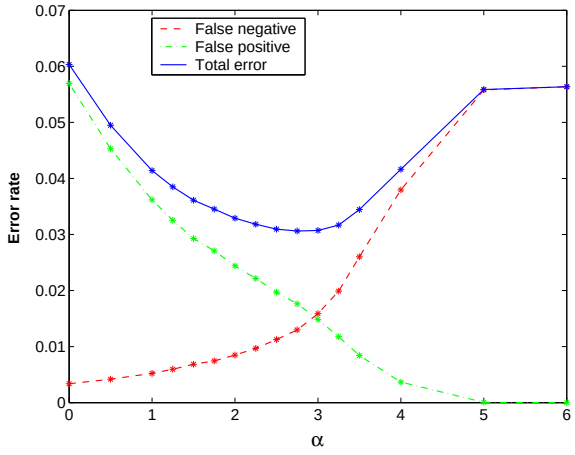
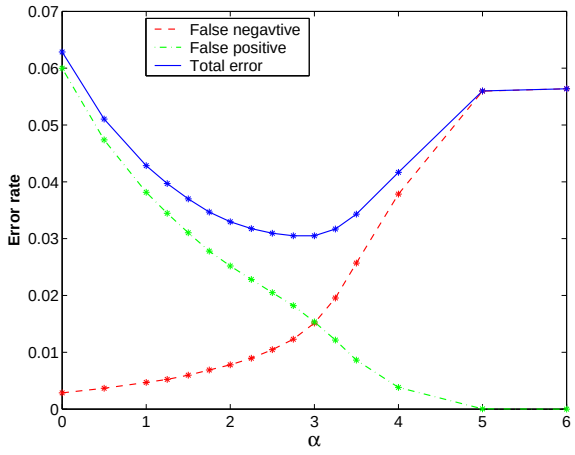



Figure 4: Face and non-face histograms of the HSV colour components, computed over all 7 training images.




(a) RGB




(b) HSV

Figure 5: Classification error versus α, determined by testing all 7 training images in both RGB and HSV colour space. The classification errors can be decomposed into false positive errors and false negative errors.

macroscopic detail. This was performed by convolution with a 3D Gaussian kernel, and the effects are shown in Figure 7.

Figures 8 and 9 show the colour segmentation results for training images #1 and #2 respectively. It can be seen that the classification algorithm preserve almost all the face pixels. Retention of the exposed arms and hands, the colour statistics of which are very close to those of the faces, is unavoidable at this stage. Some very small objects are also included due to their colour properties. In Figure 9, some significant areas of clothing (weakly connected) are included.

# 3 Non-face background removal

The colour segmentation algorithm performs as desired, but some unwanted background objects, such as hands, arms, and clothing patches, are too similar to facial skin tones to omit in this way. Object features such as size, location, and shape were used to eliminate these objects using morphological processing. Morphological processing was also used to "repair" facial periphery suffering from false-negative mis-classification errors. The goal of this step is to supply a "clean" segmentation to later processes. The amount of morphological processing is therefore minimal and conservative.

## 3.1 Image preprocessing

This step is carried out using a minimal amount of morphological processing. Given the image segmentation, the smallest objects are first removed; a closing operation is then performed, and the next smallest objects are then removed. A more aggressive closing operation follows, succeeded by a modest erosion. At this point all other objects smaller than the minimum face area are removed. Figure 10 shows the results at each step. The maximum size of object to be removed was experimentally determined as described below.

## 3.2 Location-based small object removal

The faces in the training sets are of very different sizes, due to variations in distance from the camera and, more importantly, due to partial faces. Figure 11 shows the face size statistics. All (partial) faces are larger than 500 pixels in area. In fact all but two faces are larger than 600 pixels; the mean face size is 1633 pixels. These statistics are the basis for a threshold of 500–550 pixels for removing small background objects.

The vertical locations of faces are clustered at around 20–50% of the distance from the top edge of the image.

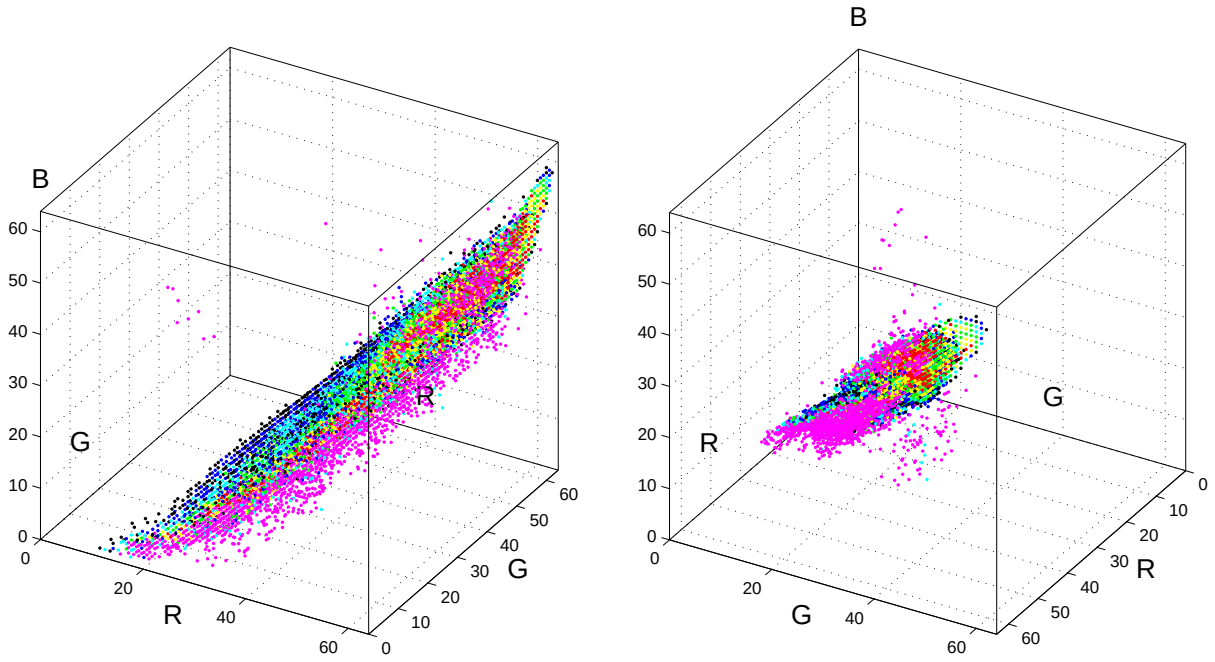



Figure 6: Raw unfiltered 3D RGB face-pixel histogram derived from the training images. Only points where the face-pixel bins contained more entries than the non-face-pixel bins are shown. Black represents the lowest values, with progressively higher values represented by blue, cyan, green, yellow, red and finally magenta; the bin size is 4.

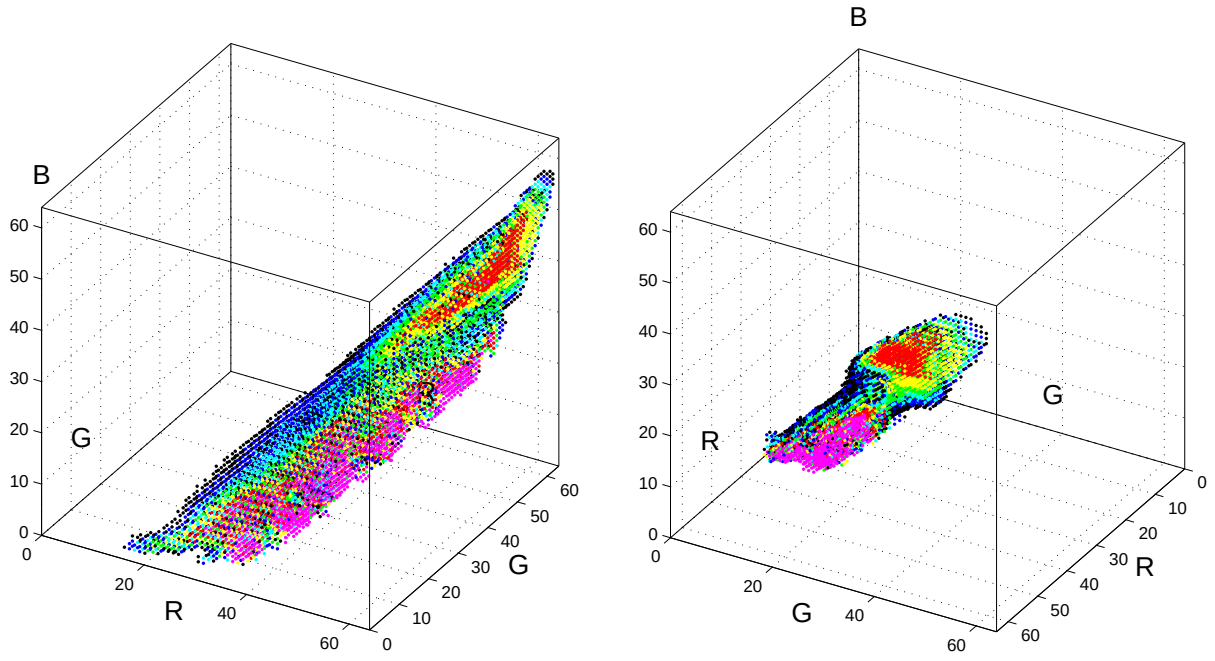



Figure 7: The smoothed RGB face-pixel histogram. Only points where the face-pixel bins contained more entries than the non-face-pixel bins are shown. Black represents the lowest values, with progressively higher values represented by blue, cyan, green, yellow, red and finally magenta; the bin size is 4.

Figure 8: Colour segmentation of training image #1.

Figure 9: Colour segmentation of training image #2.

(a) Colour segmentation

(b) Morphological closing

(c) Morphological erosion

Figure 10: An example showing the results from each step of background-removal processing (the face fragment at the bottom-right is removed by operation on the clipped image).

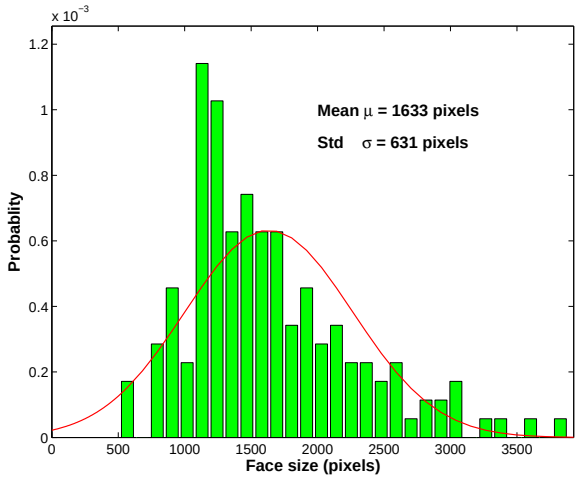



Figure 11: Face size histogram and the modelled Gaussian PDF fit to this data, computed over all 7 training images.

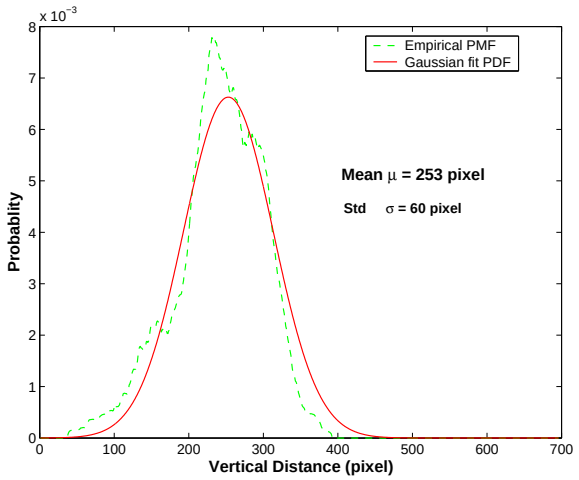
Figure 12(a) shows the histogram of the face pixels' vertical locations. The histogram is narrow and peaked, and drops to zero below 60% of the total image height. Based on these statistics, larger objects that reside in the upper and lower portion of the image may be removed. However, this step should not be too aggressive, because the face detection performance will be very poor in the event that the faces in the testing image has significantly different spatial distribution to that of the training set.

Figure 12(b) shows the smoothed PDF curve generated by convolving with a Gaussian with large standard deviation ($\sigma = 0.3\times$total image height). Motivated by the observation that the faces at the top (back) of the image are smaller than those that are lower (nearer the front), a "location correction curve" was implemented based on a modified version of the smoothed PDF. This curve can be used to modify the threshold at which small objects are removed: for example, the basic threshold of 500 pixels grows to $500 \times 3 = 1500$ pixels at the bottom edge of the image frame.

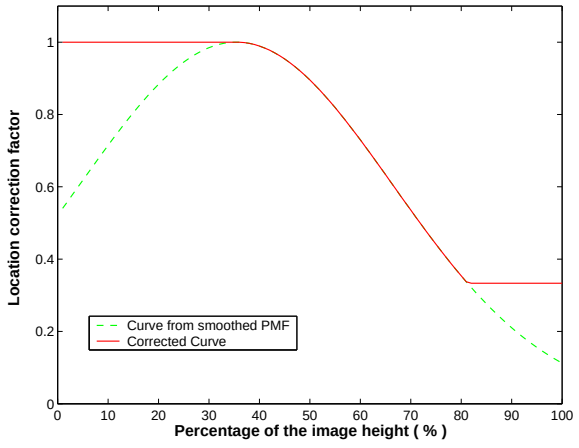
### 3.3 Principal component analysis-based object removal

A significant portion of the remaining background objects, such as arms, are thin and long. The aspect ratio of these objects may be found to selectively remove them. In order to handle rotated, non axially-aligned objects, principal component analysis is used to estimate the aspect ratio.

Given the a binary image of *e.g.* an arm with an area of $N$ pixels, vertical coordinates $\mathbf{x}^{N\times 1}$, and horizontal coor-




(a)




(b)

Figure 12: Face vertical location statistics (above) and the derived face-location correction-factor curve (below).

dinates $\mathbf{y}^{N\times 1}$, let

$$\mathbf{A}^{N\times 2} = \left[ \begin{array}{cc} \mathbf{x}^{N\times 1} & \mathbf{y}^{N\times 1} \end{array} \right].$$

The application of singular value decomposition to the matrix $A$ gives

$$\mathbf{A}^{N\times 2} = \mathbf{U}^{N\times 2}\mathbf{D}^{2\times 2}\mathbf{V}^{T\,2\times 2}$$

$$\mathbf{D} = \left[ \begin{array}{cc} D_1 & 0 \\ 0 & D_2 \end{array} \right]$$

$$\mathbf{V} = \left[ \begin{array}{cc} V_1 & V_2 \end{array} \right],$$

where $D_1$ and $D_2$ are the major and minor singular values of $\mathbf{A}$, and $\mathbf{V}_1$and $\mathbf{V}_2$ are its major and minor principal directions, respectively.

Once the singular values and principal directions have been computed, the estimated aspect ratio $\frac{D_1}{D_2}$ is known. The major principal vector $D_1\mathbf{V}_1$ also gives the principle length of the object. Knowing this information, long, high-aspect-ratio objects may be selectively removed from the image. Figure 13 shows an example of principal component analysis on an arm object after colour segmentation of an image.

### 3.4 Final image after background object removal

Figures 14 and 15 show the final image after the full background object removal sequence. The results are now suitable to supply as input to the next stage for further analysis.

## 4 Connected-component analysis and face detection

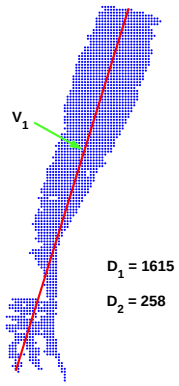
A multiple-layer face classifier was implemented using template matching. The template used is the average of all the faces. However, a simple convolution of the original image with the template will not detect all the faces without introducing a significant number of false positives or missing some number of faces. This has several causes: faces have different sizes depending on their distances from the camera; some faces are rotated; and there are structures in the images that are similar enough to the faces to produce comparable peak convolution values. It is therefore advantageous to identify the face candidates as accurately as possible first.

### 4.1 Connected-component analysis

Following histogram-based colour segmentation, the image is then subject to a series of low pass filtering, hole-filling and erosion steps. This stage is optimized to keep

(a) An arm object after colour segmentation

(b) The binary image of the arm




(c) PCA

Figure 13: Principal component analysis is used to selectively identify and remove long, thin objects.

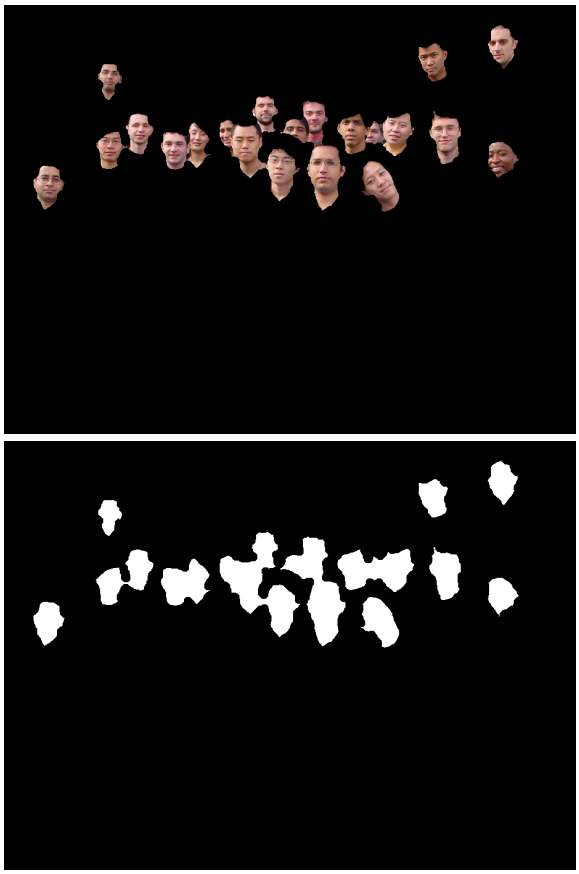


Figure 14: Final segmentation results after background removal for training image #1.

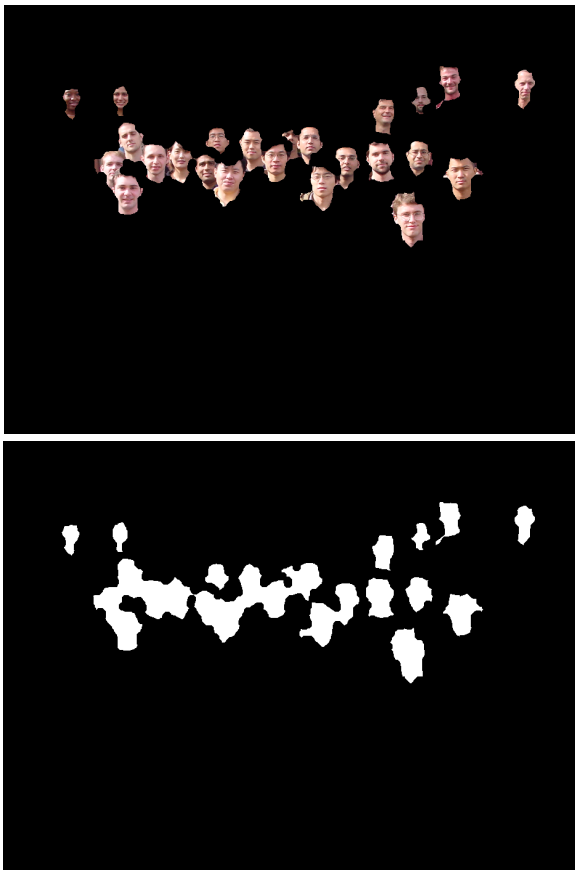


Figure 15: Final segmentation results after background removal for training image #2.

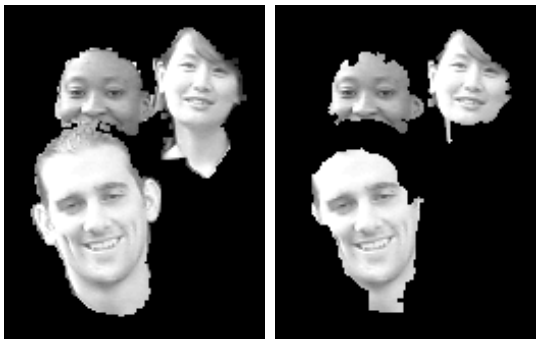


Figure 16: A loosely-connected region, before and after connected component analysis.

the non-skin areas as small as possible. After removing the non-face background as described in the previous sections, the face areas can be successfully identified.

Unfortunately, some faces are connected: in fact, about half of the faces in the training images are connected to some degree after this initial step. The connected areas are labelled, sorted by their sizes, and fed into the connected component analysis stage.

The width of a connection is usually narrower than that of the face itself: further erosion is required to break up those connections. Homogeneous erosion will eventually break up weak connections, but it will also erase small faces. An iterated, spatially-variant algorithm was therefore implemented: it includes a combination of dilation and erosion operating on only the potentially connected faces.

The algorithm was trained to find the most likely connected faces. This is accomplished by analyzing the positional and size distribution of the faces. By using different size criteria for the upper and lower regions of the face clusters, connected faces can be located. Iterated erosion and dilation are performed on each possibly connected group of faces until its size is reduced below a threshold. Since dilation and erosion are not mutual inverses, this iteration will eventually break up a loosely connected region without completely erasing the faces. A comparison of faces both before and after connected component analysis is shown in Figure 16.

## 4.2 Template matching

After connected component analysis, all the faces in the training images have been separated. A few non-face objects are left. These face candidates are convolved with the template, and the peak value is computed. This value is used to exclude those potential non-face objects. Because the majority of non-face objects have been removed by preprocessing, this step is only used in a few occasions,

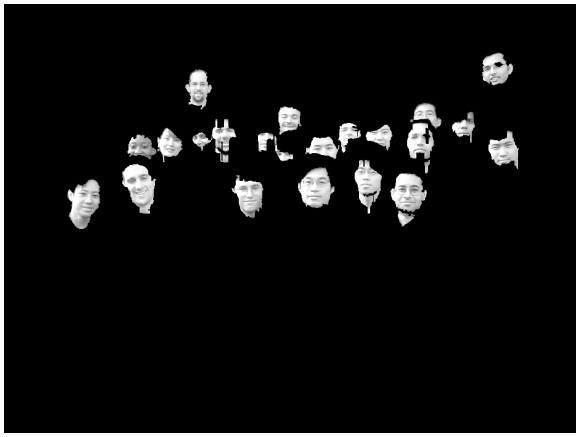


Figure 17: Face candidates prior to template matching.

Figure 18: A potential double-counting error due to the disconnected neck segment below the primary face object.

for example, to remove the roof and clothing regions. A set of sample face candidates is shown in Figure 17.

### 4.3 Repeated face correction

Iterated erosion and dilation can successfully break up loosely-connected faces, but it can also break one face up into separate objects, introducing a double-counting error; a typical situation is shown in Figure 18.

To solve this problem, the location of each newly-identified face is compared to the previously detected faces. If it is within the range of any other faces, then it is labeled as a repeated face and disregarded. In this way double counting was eliminated for all 7 training images.

### 4.4 Robustness

To improve the robustness of this face-classification algorithm, a second version is implemented in parallel using more conservative parameters than the more aggressive version described above. In the end, the results of the two algorithms are compared based on the number of faces detected, and one is chosen as the final result. If the number of faces detected by both these methods is implausible, the system falls back to face detection by pure template matching as described in the following section.

## 5 Face position refinement

Image segmentation and analysis is very effective at finding both whole and partial faces in the image. Its weakness however, is that it is is not good at finding the midpoint of a face — since faces are often at angles to the camera, the side of the face or neck is frequently added to the region, biasing the estimation of the centre. In order to to more accurately locate face centres, a pyramid-based template-matching subsystem was developed.

### 5.1 Pyramid-based template matching

The input image is converted to greyscale and successively sub-sampled over a range of sizes spaced 20% apart. Normalized cross-correlation is then performed using a constant-size average-face template to locate peaks corresponding to faces.

In order to maximise the accuracy of the template, the eye locations of each face in the input image were marked manually; this information was used to extract an aligned, rotation-free version of each face, scaled to a standard resolution.

An oval mask was applied to the average face image in order to screen out background effects, as shown in Figure 19.

Following correlation, the maximum peak that intersects with each colour-segmented face is chosen as the face centre, and all other peaks within a face-sized radius are removed. Since the centre of the template corresponds to the bridge of the nose, this operation very often locates the centre of the face; a high threshold is used to screen out unreliable results, as the morphological centre is more likely to be accurate in such cases. The resulting co-ordinates are then forwarded to the last stage for integration into the final results.

### 5.2 Robustness extension

This system also functions as a robust backup, should the connected-component analyses fail to perform reliably. In such a case, template matching is performed using a lower

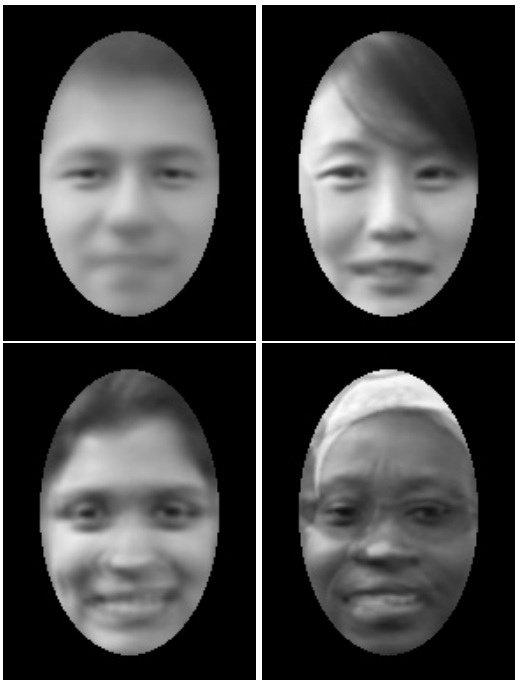


Figure 19: The masked overall average face template and the average face templates for each of the three girls.

threshold to locate as many faces as possible while minimising false positives; while the centre-finding accuracy is lower, a larger number of faces is typically found.

### 5.3 Gender recognition

The final extension of the template-matching subsystem is used for gender recognition. Average templates of each female are used to attempt to locate their faces. However, due to the high likelihood of false positives given the small number of female faces (one in nine), the threshold is particularly conservative in order to prevent finding more false positives than genuine female faces. These templates are shown in Figure 19.

## 6 Results

Our program successfully detected all faces for all the training images. The results for training image #4 are shown in Figure 20, and comprehensive detailed statistics are shown in Table 1.

This system has been shown to be an effective face detector with gender recognition capability. While its performance on further test data is as yet unknown, we are encouraged by its results to date.

## 7 Contributions

All ideas were discussed within the group and roughly equal participation was made by each member.

## References


[1] Foley, Van Dam, Foriner, and Hughes. *Computer Graphics: Principles and Practices*. Addison-Wesley, second edition in C.

[2] C Garcia and G Tziritas. *Face Detection Using Quantized Skin Colour Regions Merging and Wavelet Packet Analysis*. IEEE Trans. Multimedia 1999 Sep; 1(3):264–277.

[3] MC Shin, KI Chang, and LV Tsap. *Does Colorspace Transformation Make Any Difference on Skin Detection?* Proc. 6th IEEE Workshop on Applications of Computer Vision (WACV'02), 2002.

[4] R-L Hsu, M Abdel-Mottaleb, and AK Jain. *Face Detection in Color Images*. IEEE Proc. Image Processing, 2001. 1:1046–1049.

[5] T Hastie, R Tibshirani, and J Friedman. *Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer-Verlag, New York.

[6] B Girod. *EE368: Digital Image Processing.* Class notes, 2003 Spr.

Table 1: Final algorithm evaluation results for 7 training images. Distance (average deviaton from face centre) is computed based on the original image dimensions. Bonus is a function of the number of girls that the algorithm identified correctly.

| Image | Number of Hits | Repeated Hits | False Positives | Distance (pixels) | Run Time (s) | Bonus |
|---|---|---|---|---|---|---|
| 1 | 21 | 0 | 0 | 11.1 | 91 | 2 |
| 2 | 24 | 0 | 0 | 15.6 | 90 | 2 |
| 3 | 25 | 0 | 0 | 10.5 | 97 | 0 |
| 4 | 24 | 0 | 0 | 11.8 | 97 | 1 |
| 5 | 24 | 0 | 0 | 10.7 | 103 | 0 |
| 6 | 24 | 0 | 0 | 9.6 | 94 | 0 |
| 7 | 22 | 0 | 0 | 11.2 | 88 | 1 |
| Average | 23.4 | 0 | 0 | 11.5 | 94 | 0.86 |

Figure 20: Face detection results for training image #4.