Gaussian Model Based Face Extraction Algorithm

Conrad Sanderson and Kuldip K. Paliwal

Abstract— **This paper describes an algorithm for extraction of faces from frontal images, such as those found in the M2VTS database [1]. Using a combination of statistical methods, the algorithm removes the background, hair and shoulders, leaving only the face. This facilitates further processing for purposes such as locating eyes, lips or face recognition.**

I. INTRODUCTION

There is a need for an algorithm which removes unnecessary information from frontal images of persons, facilitating further processing (and increasing robustness) for purposes such as lip tracking and face recognition. For example, gradient based techniques for finding eye locations can easily be confused in the presence of long hair around the face.

We present an algorithm which removes the background, clothes and shoulders, leaving only the face. The algorithm is in part based on statistical models of the Red, Green and Blue (RGB) distribution of the background and face pixels. Further refinement is done by a statistical model of the location of pixels.

The paper is organized as follows: Section II provides an overview of the database used, Section III describes the algorithm, and Section IV shows a comparison with a similar algorithm.

II. THE M2VTS DATABASE

The M2VTS audio-visual database provides a good testing ground for the proposed algorithm. It is comprised of 37 people counting from zero to nine (mostly in French) while facing the camera. The database is made up of 5 sections. Each section contains a video sequence for each person in the database. The video sequences of each person often differ in hair styles, clothes, lighting conditions and zoom factors. The background remains mostly uniform. There are additional video sequences where each person rotates their head from one side to the other. If the person is wearing glasses, another head moving sequence is available without them. Frontal images without glasses of all persons were extracted from the head rotating sequences.

III. THE ALGORITHM

Throughout this section, each component of the algorithm is first briefly described, followed by detailed explanation with example images for completeness. This algorithm shall be referred to as the *Gauss-remove algorithm.* Each step of the algorithm progressively refines the RGB and luminance images, with the exception of the last part.

The algorithm utilizes multivariate Gaussian distributions to roughly classify pixels belonging to the background and to the face, as well as simple refinement for remaining pixels. The algorithm requires RGB (colour) and luminance (grey level) representations of the frontal image. Each pixel in the RGB image is made up of the 3 colour components (Red, Green, Blue). The outline of the algorithm is as follows:

1. Blur RGB and luminance images
2. Remove background
3. Remove non-face pixels
4. Remove spurious pixels
5. Fill holes

A. Blurring

To smooth the pixel colour and intensity distributions, the RGB and luminance images are blurred by neighbourhood averaging [4] - ie. each pixel is replaced with the average value of the pixel and its immediate neighbouring pixels (3x3 mask). See Figures 1 and 2 for an example. Blurring is employed over proper low pass filtering due to speed issues as well as its lack of generating "ringing effects" (Gibb's phenomenon [3]).

B. Removal of the background

In this part, a 3 dimensional Gaussian model of the background pixels is constructed. The RGB components make up each dimension. Every pixel in the image that fits the model is then "removed" (set to zero - ie. black). See Figure 3 for an example, and below for details.

1. Sobel operator based edge detection [4] is applied to the luminance image to find the rough outlines of the person.
2. For each row in the RGB image, pixels are "collected" from the left towards the middle, until a pixel representing an edge is hit at the corresponding position in the edge-detected image. This process is repeated from right to left.
3. Treating each of the collected RGB pixels as a point in 3D space - ie. a 3 dimensional vector, where each colour component is considered a dimension, the mean vector and covariance matrix are calculated.
4. Probability for each pixel in the RGB image belonging to the background is calculated using (1).

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2}|\mathbf{\Sigma}|^{1/2}} \times ...$$

$$...exp\{-\frac{1}{2}(x-\mu)^T(\Sigma)^{-1}(x-\mu)\} \qquad (1)$$

where

· $\mathbf{x}$ is a 3D vector (eg. a RGB pixel),
· $\mathbf{\Sigma}$ is the covariance matrix,
· $\boldsymbol{\mu}$ is the mean vector,
· D is the number of dimensions.

If the probability is above a certain threshold, then the pixel is considered to belong to the background, and is set to black in both the RGB and luminance images.

C. Removal of non-face pixels

In this part, a 3D Gaussian model of skin pixels is constructed. Each pixel in the image that fits the model is kept while all other pixels are removed. See Figure 4 for an example, and below for details.

1. The rough location of the face is found by looking for the brightest ellipsoid area in the luminance image. The size of the ellipsoid is set to the average size of the faces in the database. Brightness is calculated by summing all the pixels within the ellipsoid area.
2. Using the location of pixels inside the ellipsoid area, corresponding pixels from the RGB image are collected.
3. Again, treating each of the collected RGB pixels as a point in 3D space - ie. a 3 dimensional vector, where each colour component is considered a dimension, the mean vector and covariance matrix are calculated.
4. Probability for each pixel in the RGB image belonging to the face is calculated using (1).
5. If the probability is below a certain threshold, then the pixel is considered not to belong to the face, and is set to black in both the RGB and luminance images. This has the effect of removing hair and clothes, and unfortunately, the eyes and the lips. They are restored in section III.E.

At this point it may seem that removing the background is redundant since a bright ellipse area representing the face is found. However, it is possible that the image is of a person where their skin is darker than the background. Hence without removing the background, removal of non-face pixels would be impossible in these situations.

D. Removal of spurious pixels

As well as pixels belonging to the face, sometimes there is a relatively small number of non-black non-face pixels. Assuming that the majority of non-black pixels belong to the face, it is possible to remove the spurious pixels by constructing a 2D Gaussian model of the location of pixels belonging to the face. The 2D Gaussian model can be interpreted as an ellipsoid area indicating the location of the face.

See Figure 5 for an example and below for details.

1. All non-black pixels in the luminance image generated in section III.C are reated as points in 2D space - ie. the row and column locations of each pixel make up the first and second dimension respectively.
2. The mean vector and covariance matrix are calculated. Here, the mean represents the center of the face, while the covariance matrix describes the radii of the ellipse.
3. Non-diagonal elements in the the covariance matrix are set to zero to make sure the resulting ellipse is "upright" as well as reducing the effect of spurious pixels.
4. Probability of each pixel in the image belonging to the face calculated using (1), however 2D vectors are used.
5. If the probability is below a pre-defined threshold, then the pixel is considered not to belong to the face, and is set to black in both the RGB and luminance images.

E. Fill Holes

Removal of non-face pixels in Section III.C has the unfortunate side-effect of removing the eyes and the lips. Since they are enclosed by other pixels, "restoring" them is done by settting pixels to black in the original RGB image using the image from Section III.D as a reference image. For each row in the original image, scanning from left to right, pixels are set to black until a non-black pixel is hit at the corresponding location in the reference image. This process is repeated from right to left. See Figure 6 for an example.

IV. COMPARISON

In this section the proposed algorithm is compared in segmentation capability to a face segmentation algorithm based on the work presented in [5]. This alternative method shall be referred to as *ext-Wark* and is described below. An example of the resulting image from this algorithm is shown in Figure 7.

1. Segmentation of the facial skin region from the rest of the image is done by keeping only pixels which satisfy $1.2 \leq \frac{R}{G} \leq 1.45$.
2. Spurious pixels are removed via morphological opening [6].
3. Since the resulting picture contains holes for the eyes and the mouth, the original algorithm [5] is extended by a restoration process similar to the one in section III.E.

To objectively compare the algorithms the following criteria has been used:

1. Successfull hair removal, with the exception of hair on the forehead, which is allowed to remain.
2. Successfull removal of clothing and shoulders.
3. Extracted face has no sections missing.
4. Complete segmentation failure. This occurs where more than half the face is missing or when the shoulders and hair have been left completely intact.

The comparison was done on sections 1 to 4 of the database, comprising of 148 frontal images of persons without glasses. Results are presented in Table 1, where *hair* represents the percent of images where the hair has been completely removed, *clothes* represents the percent of images where the shoulders and other clothing have been completely removed, *face* represents the percent of images where the face was extracted without missing sections, and *failure* represents the percent of images where there was a segmentation failure. It has to be noted that the severity of faults in the proposed algorithm is much less than the severity of faults generated by the *ext-Wark* algorithm. This is demonstrated in Figures 6 to 9.

Fig. 1. Original RGB image (*pf 2*)

V. Conclusion

The proposed algorithm has been shown to be useful for face segmentation in images from the M2VTS database. It facilitates and likely increases robustness of further processing for purposes such as locating eyes, lips or face recognition. It has been shown to work better than the alternative *ext-Wark* algorithm, also designed to work with the M2VTS database.

Algorithm	*hair*	*clothes*	*face*	*failure*
ext-Wark	15%	52%	10%	3%
Gauss-remove	99%	76%	63%	<1%

TABLE I

Comparison of the *ext-Wark* and *Gauss-remove* algorithms.

References

[1] M2VTS Database: *http://www.tele.ucl.ac.be/M2VTS/*

[2] *Audio- and Video-based Biometric Person Authentication - proceedings of AVBPA'97,* Crans-Montanta, Switzerland, March 12-14, Josef Bigun et al. (ed), Springer, 1997.

[3] S. S. Soliman and M. D. Srinath *Continuous and Discrete Signals and Systems,* Prentice Hall, 1990.

[4] R. C. Gonzales and R. E. Woods, *Digital Image Processing,* Addison-Wesley Publishing Company, 1993,

[5] T. Wark, S. Sridharan, "A Syntactic Approach to Automatic Lip Feature Extraction For Speaker Identification", *proc. ICASSP,* 1998, pages 3693 - 3696.

[6] A. K. Jain, *Fundamentals of Digital Image Processing,* Prentice Hall, 1989.

Fig. 2. Blurred version of Figure 1

Fig. 3. Background removed from Figure 2

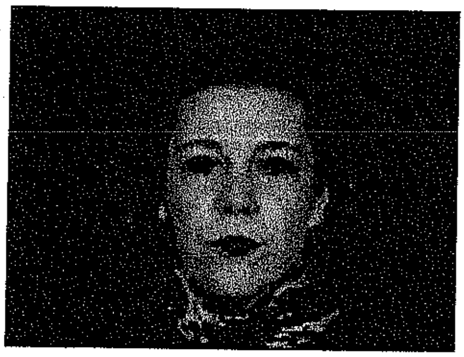

Fig. 4. Non-face pixels removed from Figure 3

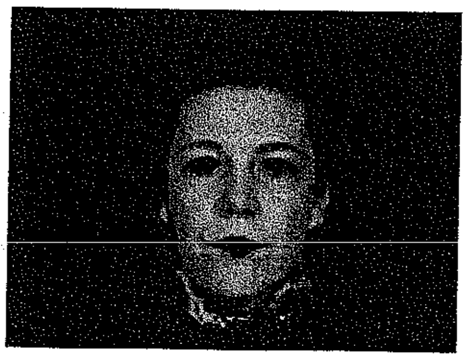

Fig. 5. Spurious pixels removed from Figure 4

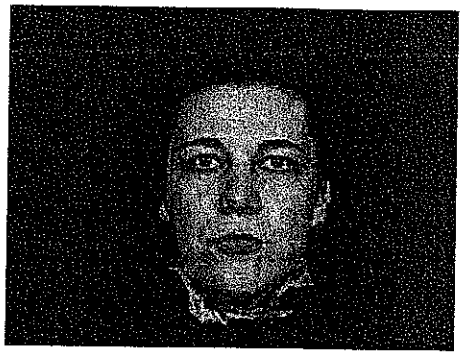

Fig. 6. Face extracted from Figure 1 using Figure 4 as an aid

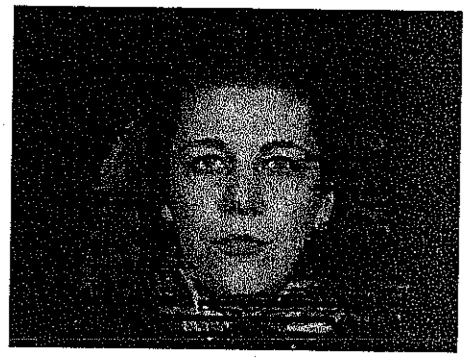

Fig. 7. Face segmentation of image *pf 2* using *ext-Wark*

Fig. 8. Face segmentation of image *bp 2* using *Gauss-remove*

Fig. 9. Face segmentation of image *bp 2* using *ext-Wark*