A Pre-filter Enabling Fast Frontal Face Detection

Stephen C. Y. Chan* and Paul H. Lewis

Multimedia Research Group, University of Southampton, Zepler Building, Highfield, Southampton S017 1BJ.
[scyc96r | phl]@ecs.soton.ac.uk

Abstract. We present a novel pre-filtering technique that identifies probable frontal illuminated face regions in colour images regardless of translation, orientation, and scale. The face candidate regions are normalised and provide the basis for face verification using published face detection algorithms. The technique focuses on a fast search strategy to locate potential eye-pairs in an image or video frame. The eye-pair candidates indicate areas that may contain faces. Scale and orientation is inferred from the eye-pairs, and a neural network is used to confirm the normalised face candidates.

1 Introduction

Detecting the presence of human faces can provide important cues for many image and video analysis tasks[1]. We are interested in enhancing multimedia tasks, in particular content-based video browsing, retrieval and navigation but face detection and location is also used as a pre-requiste for face analysis tasks such as recognition and expression interpretation. It is non-trivial if faces encoded in visual data can appear in any pose, position, orientation, and scale. The task is further compounded by problems associated with illumination variation and noise.

Some of the most robust available techniques for face detection are computationally intensive, applying their elaborate detection algorithms at many scales and orientations in all possible positions in each image. The aim of this paper is to present a pre-filtering technique which can identify, relatively quickly, regions in an image or video frame likely to contain human faces. Face candidates are detected regardless of position, orientation, and scale but initially we have assumed full frontal illuminated faces.

The paper is presented in the following manner: Section 2 describes some related work contributing to face detection; Section 3 presents an overview of the pre-filtering technique; Sub-sections 3.1, and 3.4 present the technique in detail; Section 4 reports results of some experimental work; and Section 5 gives the conclusions.

* Stephen Chan would like to acknowledge support from the EPSRC.

2 Related Work

Detecting frontal profile faces has been investigated using a variety of different approaches. Recently, the detection of faces with the head in varying poses has been reported by Yow *et al.*[12]. Their approach detects features using spatial filters, and forms face candidates using geometric and grey-level constraints. A probabilistic framework then evaluates the face candidates for true faces. Chow *et al.*[2], detect facial features to isolate faces in a constrained manner.

Chen *et al.*[1], use colour characteristics to detect faces in images against complex backgrounds. A neural network trained to recognise skin coloured pixels is used to isolate areas of skin, and eventually forms candidate face regions. Face regions are processed for lips to verify the existence of faces. Techniques that use motion have been used to isolate areas of an image, and analysing colours that make-up facial features, Choong *et al.*[5]. Dai *et al.*[3], use colour to hypothesise the location of faces, where faces are evaluated as a texture that is based on a set of inequalities derived from a Space Grey Level Dependency(SGLD) matrix, described in Haralick *et al.*[4].

A computationally expensive, but arguably the most robust, approach to face detection is proposed by Rowley *et al.*[10]. A small input window is passed over every part of an image, and a neural network filter is used to establish whether or not a face is present. Scale invariance is achieved by sub-sampling each image at different resolutions, and searching each of the sub-images. A rotational invariant version of this, also by Rowley *et al.*[8], is achieved by estimating the angle of the sub-image within the input window. The sub-image is then de-rotated and presented to the neural network for classification.

There is a growing amount of literature concerned with verifying the existence of faces at a given location. However, the fast and automatic location of face candidate regions, as a pre-filtering operation, is important if rapid and reliable face detection is to be achieved in video applications.

3 Overview of the Technique

This paper proposes a pre-filtering technique which rapidly identifies locations in video frames where preliminary evidence suggests a face may be sited. A more elaborate and established technique is then used to confirm or deny the existence of faces at these locations. The pre-filtering technique is based on the fact that, for frontal illuminated faces, the eyes are a usually a prominent feature of the face[6][7][11]. They have a spatial distribution that is roughly related to other facial features such as the nose and mouth. The distance between a pair of eyes gives an indication of the size of the face, and the positions of the eyes can be used to estimate the orientation.

Using this premise the technique generates regions that are most likely to contain faces. These regions are then verified, in turn, to test whether a face actually exists. Generating these regions relies on detecting possible pairs of eyes(eye-pairs) that may or may not belong to a face. The eye-pairs inherently provide information about the location, orientation and scale of potential faces. Square regions around the eye-pairs are used to establish the area that may contain the rest of the face. These areas are then normalised so that they represent possible upright faces. A suitable face verification technique can then be used to verify the captured areas to confirm the existence of faces. The current system uses a neural network for the final face verification stage and is based on the approach of Rowley *et al.*[10]. Figure 1, illustrates the individual stages of the pre-filtering process and face verification.

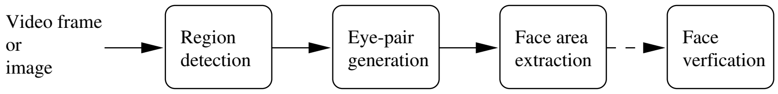

Fig. 1. The stages in isolating regions that may contain faces.

3.1 Region detection

The initial stage receives an image and segments it into regions. Each of these regions are evaluated in turn to see whether they satisfy certain criteria pertaining to eyes. The visual input is segmented by remapping pixel values to a voxelised RGB colour-space. Mapping colours of an image to its representative voxel produces homogeneous regions. The segmentation process has complexity of $O(n)$ where n is the number of image pixels, and it is ideal for applications where speed is a concern. It can be efficiently implemented by reducing colour bits in the red, green, blue colour channels for each pixel using bit masks. Our system uses only the first significant bit. Dark regions are extracted from the segmented image by visiting every dark pixel(seed pixels) and flood filling surrounding areas that have the same voxel number as the current seed pixel. During the flood filling process, the number of flooded pixels are counted and the extreme co-ordinates of the fill are preserved.

To reduce the computational complexity in the next stage, each region is evaluated with a set of heuristics that determine whether it could be a potential eye region. The heuristics are as follows but it should be noted that parameter values are not critical. They are used to eliminate candidate regions which have properties sufficiently different from those of an eye region that they may be eliminated.

Definitions:

w and h are the width and height of the segmented image in pixel units.
R_n, where n is the region number in the set of regions R.
$R_n.width$ is the width in pixels.
$R_n.height$ is the height in pixels.
$R_n.aspect$ is the aspect, defined as $\frac{R_n.width}{R_n.height}$.
$R_n.numberofpixels$ is the number of pixels the region occupies.

$R_n.homogeneity$ is a measure of homogeneity, defined as $\frac{R_n.numberofpixels}{R_n.width*R_n.height}$.

1. Elimination of regions that are too small and too large.
   $1 < R_n.width < 0.5w$ **and** $1 < R_n.height < 0.5h$
2. Regions associate with eyes have a limited range of aspect ratio.
   $\frac{1}{7} < R_n.aspect < 7.0$
3. This criterion determines how much the region covers its minimum enclosing rectangle.
   $R_n.homogeneity > 0.5$

We found that smoothing the image reduced noise and produced better results. A 3x3 smoothing mask was convolved with the input before the region detection process was initiated. Segmentation of the filtered input produced smoother regions and a reduction of false positive eye regions was recorded. Figure 2 illustrates an image passing through the stages described in this section.

Fig. 2. Images displaying the results of each sub-stage during region detection. The filtered regions are indicated by the rectangles in the *Region filtering* image.

3.2 Eye-pair Generation

The eye-pair generation process attempts to pair regions together that may potentially belong to a face. Given that there are n regions after the region detection stage, the number of possible eye-pairs is $\frac{n^2-n}{2}$. It is desirable to reduce the number of eye-pairs by comparing regions with other regions using a set of eye-pair heuristics. Again, parameters are not critical and were obtained from observations of a wide variety of images containing faces. The algorithm is as follows:

Definitions:
$distance_x(R'_j, R'_k)$, horiz. distance between the centres of regions R'_j and R'_k.
$distance_y(R'_j, R'_k)$, vert. distance between the centres of regions R'_j and R'_k.

For all possible eye-pairs (R'_j, R'_k):
if $distance_x(R'_j, R'_k) > distance_y(R'_j, R'_k)$ **then**
$$relative\_width = \frac{R'_j.width}{R'_k.width}, \quad region\_aspect1 = \frac{R'_j.width}{R'_j.height},$$

$region\_aspect2 = \frac{R'_k.width}{R'_k.height}$, $sum\_of\_widths = R'_j.width + R'_k.width$
else
$relative\_width = \frac{R'_j.height}{R'_k.height}$, $region\_aspect1 = \frac{R'_j.height}{R'_j.width}$,
$region\_aspect2 = \frac{R'_k.height}{R'_k.width}$, $sum\_of\_widths = R'_j.height + R'_k.height$
endif
if $0.2 < relative\_width < 5.0$ **and**
$k_1 * sum\_of\_widths < region\_distance < k_2 * sum\_of\_widths$ **and**
$0.8 < region\_aspect1 < 7.0$ **and** $0.8 < region\_aspect2 < 7.0$ **then**
Store eye-pair (R'_j, R'_k)

The condition $distance_x(R'_j, R'_k) > distance_y(R'_j, R'_k)$ determines if the eye-pair (R'_j, R'_k) is more horizontal or vertical. The reason for having such a condition is that the aspect ratios can be calculated roughly relative to the vertical position of a face, where the width of a region relates to the width of an eye region of an upright face. An input image with a face on its side will have the eye regions with the width being the actual height of the eyes in the image. The term *relative_width* ensures that no two regions have greatly exaggerated size differences, since regions belonging to the same face should not vary by orders of magnitude. Illumination will affect the size of eye regions in the segmentation process and thus a range is considered. *region_aspect*1 and *region_aspect*2 ensures that the eye regions are approximately in-line with each other. This eliminates eye-pairs with one eye region in a horizontal position and an eye region in a vertical position. The $k_1 * sum\_of\_widths < region\_distance < k_2 *$ sum_of_widths, where $k_1 < k_2$, ensures that the distance between an eye-pair is not exaggerated relative to the size of eye regions. In this case the sum of widths relative to the upright face position is used to give a measure of the size of eye regions.

3.3 Face Area Extraction

The resulting eye-pairs possess information that allows rotation and scale invariance of faces. This stage takes each eye-pair and extracts a square region which covers the main facial features(eyes, nose, mouth). Figure 3a, presents the definition of the square region. Two square regions must be extracted to achieve full rotation invariance. The eye-pairs form an imaginary line between the two squares and both areas on either side must be taken into account. Figure 3(b...g), shows a face image and all the captured areas on both sides of the generated eye-pairs. The captured face candidate areas are rotationally normalised.

Our implementation captures face candidates which are rotationally normalised on the fly. This is achieved by scanning pixels parallel to the eye-pairs and remapping them to an orthogonal grid that is of the same pixel dimensions as the pre-determined square capture area.

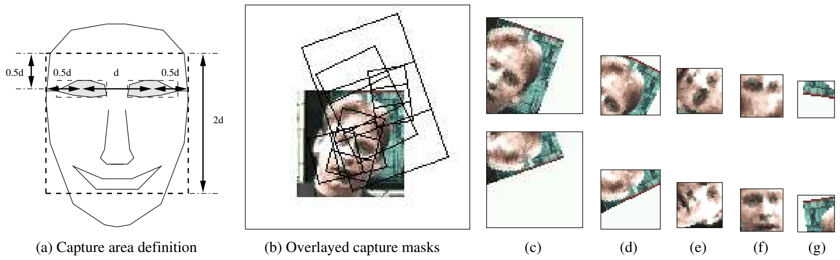

Fig. 3. An image with captured face candidates based on eye-pairs. The columns of images(c)...(g), show two images captured for each eye-pair.

3.4 Face Verification

The face candidate images captured in the previous stage now present us with a pattern classification problem for upright frontal faces. We use a neural network based on the work by Rowley *et al.*[10], to classify each face candidate sub-image. They use a 20 x 20 pixel window moved over the entire image and perform filtering functions to enhance the image viewed by the window, before it is passed to a neural classifier. Rowley *et al.* pre-process the input image by correcting the lighting and then performing histogram equalisation to improve the contrast. Our system only needs to perform histogram equalisation on the face candidate images since we have initially assumed frontal illuminated faces.

Video frames and scanned images were used to generate training patterns. Training the network used visual data generated from the pre-filtering process where over 450 *representative faces* were manually selected. False positives generated by the neural network were augmented to the non-faces training set, and the network retrained.

Face candidates are resized to 20 x 20 pixel dimensions, greyscaled, and histogram equalised, before mapping to the trained neural network and the output is thresholded to give a binary decision, face or non-face.

4 Experimental Results

Our system uses 24 bit colour images or video frames and is being developed on a Pentium 133Mhz machine running Linux. Each frame is mapped to a 300 x 300 pixel frame buffer before any processing takes place.

Figure 4, shows various frontal illuminated face views, where located faces are signified with a box that also indicates the orientation. When a general database contain over 400 faces was used, including many very small faces, the pre-filtering algorithm detected eye-pairs for 53% of the total faces. In order to test the algorithm more fairly, a subset of the original database was established with 103 images containing at least one full frontal face. No faces were less than 60 pixels across but apart from this lower limit on size, faces could appear at any scale or orientation.

Fig. 4. Representative examples of faces found by the system. Each image shows 4 numbers: the number of faces in the image, the number of detected faces, the number of false positives, and the number of eye-pairs.

The total number of faces contained in the images was 128. After running the pre-filtering algorithm, eye-pairs were found for 75% of the faces and of these 65% were correctly confirmed as faces by the neural net verifier. This result was obtained with our initial implementation of the verifier and it is expected that the proportion correctly verified will improve with more careful training. On average about 70 eye-pairs were found per image, which is an important statistic since it is this number which determines the number of applications of the computationally intensive neural net verification algorithm in our approach. The benefits of the approach are clear when it is recalled that, in Rowley *et al.*'s original approach[9], neural nets are applied 193737 times per image with a processing time of 590 seconds on a Sparc 20 although they describe modifications which give a small degradation and a processing time of 24 seconds. Currently our approach is averaging about one image per second on a Pentium 133. Rowley reports detection rates of between 78.9% and 90.5% and although these are higher than our current rates, we believe that the speed improvement in our approach shows substantial promise when working towards real-time applications.

5 Conclusions

We have developed a pre-filtering technique for face detection which provides an order of magnitude improvement in processing time on the method described by Rowley *et al.* Our pre-filtering technique can currently detect 75% of eye-pairs belonging to faces in a test database containing full frontal faces of reasonable size. We believe that, although parameters in the algorithm are not critical, it will be possible to extend the cases considered in order to improve the robustness of the technique.

References

1. C. Chen and S. P. Chiang. Detection of human faces in colour images. In *Vision Image Signal Processing*, volume 144 of *6*, pages 384–388. IEE, 1997.
2. Gloria Chow and Xiaobo Li. Towards a system for automatic facial feature detection. *Pattern Recognition*, 26:1739–1755, 1993.
3. Ying Dai and Yasuaki Nakano. Face-texture model based on sgld and its application in face detection in a color scene. *Pattern Recognition Society*, 29(6):1007–1017, 1996.
4. Robert M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 3:610–621, 1973.
5. Choong Hwan Lee, Jun Sung Kim, and Hyu Ho Park. Automatic human face location in a complex background using motion and color information. *Pattern Recognition*, 29(11):1877–1889, 1996.
6. David E. Benn Mark S. Nixon and John N. Carter. Robust eye centre extraction using the hough transform. In *1st International Conference on Audio-and Video-Based Biometric Person Authentication, Lecture Notes in Computer Science*, pages 3–9, 1997.
7. Daniel Reisfeld and Yehezkel Yeshurun. Preprocessing of face images: Detection of features and pose normalization. *Computer Vision and Image Understanding*, 71(3):413–430, September 1998.
8. Henry A. Rowley, Shumeet Baluja, , and Takeo Kanade. Rotation invariant neural network-based face detection. Technical report, CMU CS Technical Report CMU-CS-97-201, 1997.
9. Henry A. Rowley, Shumeet Baluja, and Takeo Kanade. Human face detection in visual scenes. Technical report, CMU-CS-95-158R, Carnegie Mellon University, http://www.cs.cmu.edu/ har/faces.html, November 1995.
10. Henry A. Rowley, Shumeet Baluja, and Takeo Kanade. Neural network-based face detection. In *Transactions On Pattern Analysis And Machine Intelligence*, volume 20 of *1*, pages 23–38. IEEE, January 1998.
11. Li-Qun Xu, Dave Machin, and Phil Sheppard. A novel approach to real-time non-intrusive gaze finding. In *BMV, Southampton*, volume 2, pages 428–437, 1998.
12. Kin Choong Yow and Roberto Cipolla. Feature-based human face detection. *Image And Vision Computing*, 15(9):713–735, 1997.