AUTOMATIC IDENTIFICATION OF PORTRAITS IN ART IMAGES DATABASES

E. Šikudová, M. Gavrielides, I. Pitas

Department of Informatics
Aristotle University of Thessaloniki
Box 451, Thessaloniki 541 24, GREECE
E-mail: (elena, marios, pitas)@zeus.csd.auth.gr

ABSTRACT

We have developed a method for automatic identification of portraits in art images database. The method uses color, intensity and edge information to segment candidate regions. It fits an ellipse on the segmented candidate regions from which a set of features is extracted. Those features serve as input to a neural network which is trained to distinguish between face and non-face regions. Images containing face regions were classified as portrait images. The method was evaluated with ROC analysis on a set of 200 art images. The results show a sensitivity of 90% at 32% false positive rate and they are encouraging for the further development of this method.

Keywords: portrait retrieval, art images, face detection

1. INTRODUCTION

The growing use of electronic media for storing pictorial information leads to large amount of data in image databases. Imagine an electronic art image library which provides material that can be used for fine art reference books and electronic publications, for the front covers of novels, for TV and film use, product packaging, advertising, etc. Locating certain images in such a databases is a challenging topic.

Lots of content based retrieval systems were introduced in the last years, but they cannot address queries which are not described by the rules used in the systems. Such a query is finding portraits in a digital painting database.

The problem of portrait recognition can be seen as a two-class recognition problem. A painting is segmented by color and each segmented region is classified as a face or non-face. Once there is a region in the image classified as face we annotate the image as a portrait.

Recently an extensive survey on face detection appeared in [1]. The authors give an overview of existing systems for face detection, but these systems use photographic images of real world and do not address the fine art images. Even among these systems there are only a few dealing with distinguishing between portraits and non-portrait images.

Saber et al. [2] developed a system for automatic annotation of images. Their method was based on Gaussian distribution of colors and application of an adaptive threshold of color histogram. The true positive rate for portrait annotation was 80% and false positive rate 0% in a database of 31 images (10 portraits). However they applied the method on photographs only, not on paintings.

Gevers et al. [3] also addressed the issue of distinguishing between portraits and non-portraits, but they used only photographic and synthetic images. The true positive rate of their method is shown to be 81% and 72% for identifying portraits in two different test sets.

We have developed a method for the recognition of portraits in a database of art images. We used the assumptions that a portrait is an image which contains a face in mainly frontal view without any occlusion. The face is in the focus (i.e. it is a foreground object) of the image. The method uses the combination of color and shape features.

This paper is organized as follows. Section 2 describes the available set of training and testing images. Section 3 focuses on the developed method describing the region segmentation, the feature collection and the classification steps. Section 4 describes the achieved results and the evaluation of this method using the ROC and fROC analysis and provides the discussion of these results. The final section contains the conclusions.

2. DIGITAL PAINTING DATABASE

A sample collection of images was provided for the project MOUMIR by the Bridgeman Art Library (BAL), a fine art photographic library. BAL is now the most comprehensive source of fine art images for publication in the world, acting as an agent to over 1,000 international museums, galleries and private collections.

The provided database contained images of different fine art artifacts - paintings, drawings, lithographes, statues, fur-

This work has been supported by the European Union research training network "Methods for Unified Multimedia Information Retrieval" (MOUMIR)

niture etc. We were interested only in paintings. The images were provided in JPEG format with resolution ranging from 72 to 160 dpi and average image size 590×470 pixels.

A total number of 200 images of paintings was available from the BAL database, which we used for training and testing of our method. For evaluating of the performance of our method, we created two sets containing 100 images each. From these 100 images there were 50 portraits and 50 non-portrait pictures.

3. METHOD

The method comprises of a region segmentation step, a feature collection step and a classification step as indicated bellow.

Segmentation of face regions. Here we identify the skin-colored pixels which lie in the foreground of the image. We also use the edge information for identifying connected regions.

Feature collection. In this step we fit an ellipse to each region and extract region and ellipse based features

Classification. We use an artificial neural network to classify a region as face or non-face. Determination if an image is a portrait is based on the presence of a face region.

The details of each step of the method are described in following sections and their results will be illustrated on two selected portrait images (Figure 1). The portraits are the *Portrait of Countess Sophie Matiuskina (1755-1796)*, by Kirill Ivanovich Golovachevsky and the *Portrait of M.A. Bek*, by Karl Pavlovich Bryullov (©Bridgeman Art Library).

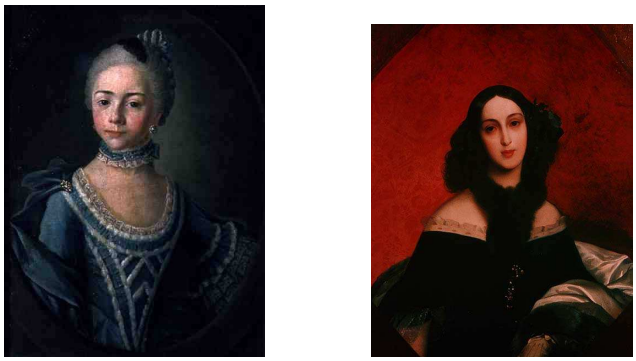

Fig. 1. Original paintings (©Bridgeman Art Library)

3.1. Segmentation of face regions

3.1.1. Identifying the skin-colored pixels

Several studies have shown that the human skin color composes an easily identified cluster in different color spaces. Among the most used color models is RGB and the normalized rgb space. Other color models used in the face detection are HSI, HSV and HLS systems, which are compatible with the human perception of color. Moreover, a number of other spaces including YIQ, YES, YCrCb, YUV, CIE-xyz, CIE L*a*b, CIE L*u*v were used in human skin detection [1]. There are several ways for identifying the skin color pixels: thresholding the color space, histogram intersection, statistical methods (Gaussian probability density functions or mixture of Gaussians) [1].

In our method, the HSV color model was chosen because of its compatibility with the human color perception. In this space the H and S components describe color hue and saturation. To identify the skin-colored pixels we used two-dimensional Gaussian probability density function (pdf) in the HS space. The intensity level of colors (component V) was used to identify the foreground of the image. The parameters of the Gaussian pdf were estimated using pixels from manually extracted 50 face regions in the portrait images in the training set.

The probability density function of a bivariate normal (Gaussian) distribution describing facial color distribution is given by

$$f(\boldsymbol{x}) = \frac{1}{2\pi|\boldsymbol{C}|^{-\frac{1}{2}}} * exp\ (-\frac{1}{2}[\boldsymbol{x}-\overline{\boldsymbol{x}}]^T\boldsymbol{C}^{-1}[\boldsymbol{x}-\overline{\boldsymbol{x}}]) \quad (1)$$

where $\boldsymbol{x} = [H, S]^T$ is the vector of hue and saturation values, $\overline{\boldsymbol{x}} = [\overline{H}, \overline{S}]^T$ is the vector of the mean values, $\boldsymbol{C}$ is the covariance matrix. Since the skin color cluster is localized in red segment of HS space, we transformed the $\langle 0, 2\pi\rangle \times \langle 0, 1\rangle$ HS space into $H'S$ space $\langle -0.5, 0.5\rangle \times \langle 0, 1\rangle$, where the hue coordinate was transformed as follows:

$$H' = \begin{cases} \frac{x}{2\pi} & \text{if } x \leq \pi \\ \frac{x}{2\pi} - 1 & \text{if } x > \pi \end{cases} \quad (2)$$

In the $H'S$ space the mean and covariance matrix of 1 have the following values:

$$\boldsymbol{C} = \begin{pmatrix} 0.0035 & -0.0008 \\ -0.0008 & 0.0332 \end{pmatrix} \quad (3)$$

$$\overline{\boldsymbol{x}} = [\overline{H}, \overline{S}]^T = [0.0597, 0.4025]^T. \quad (4)$$

The resulting contour lines of the Gaussian distribution can be seen in the Figure 2. They take the shape of an ellipse with the center in $\overline{\boldsymbol{x}}$ and orientation and axes given by the covariance matrix.

Each pixel of an image having coordinates $\boldsymbol{x}$ in the H'S space was assigned a value

$$G(\boldsymbol{x}) = exp\ (-\frac{1}{2}[\boldsymbol{x}-\overline{\boldsymbol{x}}]^T\boldsymbol{C}^{-1}[\boldsymbol{x}-\overline{\boldsymbol{x}}]). \quad (5)$$

Pixels with the same $G(\boldsymbol{x})$ value belong to the same contour line. The values lie in the range $\langle$0,1$\rangle$ and the closer the

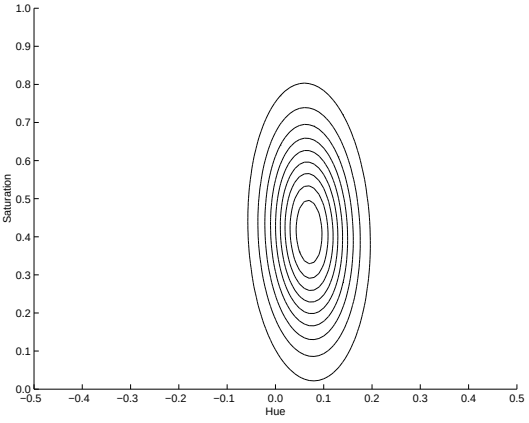

Fig. 2. Contours of Gaussian distribution of face colors

contour is to the mean values (i.e. the center) the higher value of G it has. For the mean values $G(\overline{\boldsymbol{x}}) = 1$.

Only pixels with value greater than a threshold were taken as possible face pixels. The threshold was experimentally set to 0.05. The result of this step for the two portrait examples is shown in the Figure 3.

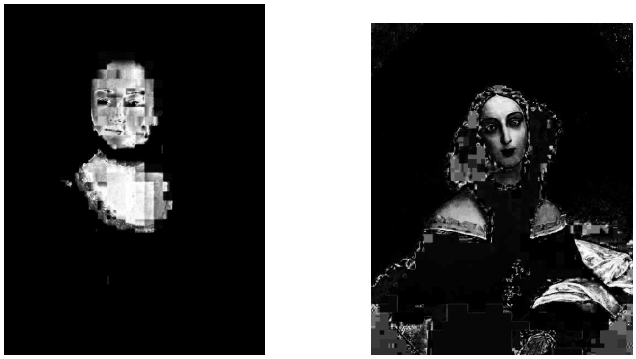

Fig. 3. $G(\boldsymbol{x})$ values of image pixels

3.1.2. Identifying the foreground pixels

In order to get only foreground object pixels, the intensity V was adaptively thresholded by the following algorithm [4]. Based on the idea that the background is found in the corners, the arithmetic mean of the intensities in corners (10%x10% of the image size) was taken as an initial value of the threshold. Then the algorithm recursively adapted the threshold until the error of interpreting background pixels as objects pixels, and vice versa was minimized. Pixels with values smaller than the threshold were considered to be background pixels. If the histogram of an image was purely bimodal, the threshold was set in the middle of the valley between the 2 modes.

In this step, we used the assumption that, in a portrait image, the face is in the foreground, since the face is the region which the attention has to be drawn to. The threshold values of the two example images are 0.3171 and 0.2609.

3.1.3. Edge manipulation

The pixels identified as skin-colored ones sometimes cover bigger area than the face itself. We used the edge information in the image formed by the $G(\boldsymbol{x})$ values of the original image to separate the face from the body. For detecting edges we used the range edge detector [5] defined as

$$E(x, y) = max_B(x, y) - min_B(x, y) \tag{6}$$

where max_B and min_B denote the maximum and minimum in a given neighborhood B of the pixel (x, y).

3.1.4. Identifying connected regions

Facial region pixels are only the intersection of pixels identified in 3.1.1 and 3.1.2, with the incorporation of the edge information from section 3.1.3.

Furthermore, under the assumption that a face in a portrait should cover at least a certain percentage of the painting area, regions smaller than a threshold size were removed. The threshold depends on the image size and is given as follows:

$$(\text{image width} * \text{image height})/400. \tag{7}$$

The final sets of the potential facial regions in our example images of Figure 1 are seen in Figure 4.

Fig. 4. Final set of possible candidate regions

3.2. Feature collection

In order to describe a facial region, the features of bounding box and of an ellipse fitted to the region were used. We used 2 features of the bounding box:

- Relative height

- Relative width

A bounding box of the region with sides parallel to the borders of the image was used. The width and the height of the bounding box relative to the image width and height respectively was computed.

Another 5 features were collected from the fitted ellipse.

- Orientation
- Aspect ratio of the axes
- Relative horizontal placement
- Relative vertical placement
- Region/ellipse overlap

The features are described in the following section.

3.2.1. Ellipse fitting

For fitting an ellipse we used an algorithm described in [6]. The method was based on a reformulation of the fitting task as an linear optimization problem with a quadratic constraint. The algorithm solved this problem directly by a standard least squares minimization. The method worked on segmented data (that means that all data points are assumed to belong to one ellipse), which is the reason why only the borders on segmented regions were used.

The ellipse is a conic, which can be described as an implicit second order polynomial:

$$F(x,y) = a_1x^2 + a_2xy + a_3y^2 + a_4x + a_5y + a_6 = 0 \quad (8)$$

or in vector form $F_a(x) = \mathbf{a}^T\mathbf{x}$, where $\mathbf{a}^T = [a_1, a_2, a_3, a_4, a_5, a_6]$ and $\mathbf{x}^T = [x^2, xy, y^2, x, y, 1]$.

The objective of the fitting method is to find a parameters vector **a** which minimize the sum of squared algebraic distances of the facial region border points to the conic $F_a(x)$. This problem is solved directly by the standard least squares approach, with a constraint ensuring that the resulting conic will be an ellipse.

The resulting vector of ellipse parameters **a** is then transformed to the implicit representation

$$\frac{(x-x_c)^2}{a^2} + \frac{(y-y_c)^2}{b^2} = 1 \quad (9)$$

where the ellipse is identified by its center (x_c, y_c) and the length of its major and minor axes (a and b). Equation (9) defines an ellipse with axes parallel to the x and y axes of the plane. Moreover, a generic ellipse is rotated around its center by the angle θ, as it is seen on Figure 5. Using the notation of Figure 5 we can now describe the features.

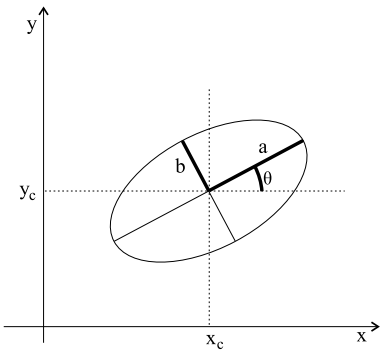

Fig. 5. Ellipse description

The orientation θ of the ellipse determines the angle between the x axis and the major axis a. The aspect ratio of the axes is the ratio of the major and minor axes a/b. The relative horizontal and vertical placement is the position of the ellipse center (x_c, y_c) relative to the image size. The region/ellipse overlap feature determines what percentage of the best-fit ellipse is covered by the region:

$$Overlap = \frac{area(Region \bigcap Ellipse)}{area(Ellipse)} \quad (10)$$

Examples of fitted ellipses are shown in Figure 6.

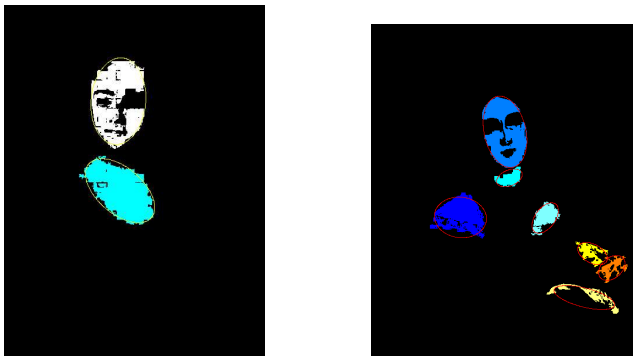

Fig. 6. Fitted ellipses

3.3. Classification

3.3.1. Classifiers

We used neural networks to train the system to classify the facial regions. The values of 7 features for all segmented regions were normalized to have values in $< -1, 1 >$. Principal component analysis showed that all 7 features should be retained (principal components which account for 99.9% of the variation in the data set). The values were then the input for a feed-forward multi-layer neural network [7]. There was one hidden layer in the network containing 20 neurons.

The output of the network was a single value in the range $< 0, 1 >$ for each region. We used the hyperbolic tangent sigmoid transfer function (tansig) for the hidden layer and the symmetric saturating linear transfer function (satlins) for the output layer. For training, the Levenberg-Marquardt algorithm was applied using batch processing.

3.3.2. Training

The neural network was trained using the features extracted from the candidate regions in the training set S1. There were 55 faces in the set. The location of the faces in images were manually determined and the extracted features were labelled with target values 1 (face) and 0 (non-face). For training the network after segmentation we had features from 46 facial regions and 387 non-facial regions. After the training, the weights of the neural network were fixed and the network was tested on the test set S2. In the test set S2 we had 50 faces and after segmentation we had features from 47 regions corresponding to a face and 440 non-facial regions.

4. RESULTS AND DISCUSSION

The performance of the proposed method was evaluated using two different ways. We used the Receiver Operating Characteristic (ROC) curve and the Free-Response Receiver Operating Characteristic (FROC) curve.

4.1. Evaluation per region

In this evaluation the performance of the classifier to distinguish between face and non-facial region was described with a FROC curve. The points of the curve were acquired by sweeping the threshold of the neural network output from 0.001 to 1 and calculating the pairs of *sensitivity* (true positive rate) and *average number of false positive regions per image*. The sensitivity is defined as

$$\frac{\text{number of regions correctly classified as faces}}{\text{total number of faces in the set}}. \quad (11)$$

In images, any segmented region is a potential false positive, so there is no limit to the number of FP's. That's why the average number of false positive regions per image is a meaningful characteristics:

$$\frac{\text{number of regions falsely classified as faces}}{\text{total number images}}. \quad (12)$$

The results of this evaluation is shown in Figure 7. It can be seen that the sensitivity rate of 80% for detecting a face region can be achieved at the relatively low average false positive rate 0.34 of region per image.

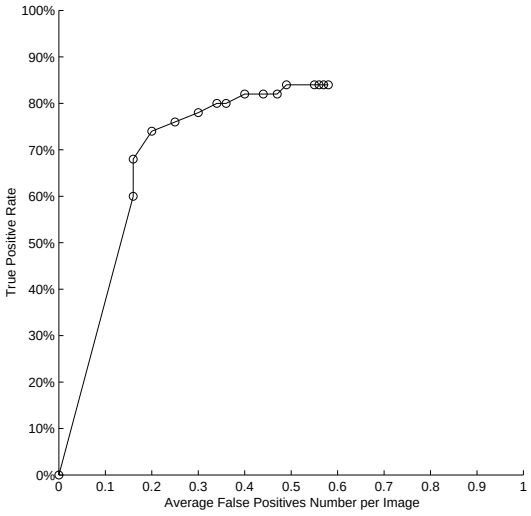

Fig. 7. FROC, performance in the set S2

The same evaluation was done also on the subset containing only portraits, in order to see where are the false positives accumulated. The results are shown in Figure 8. Here the sensitivity of 80% is achieved at average false positive rate 0.38 of region per image. By comparing the two curves in Figure 7 and Figure 8, we can see that the false positives are evenly distributed between portraits and non-portraits.

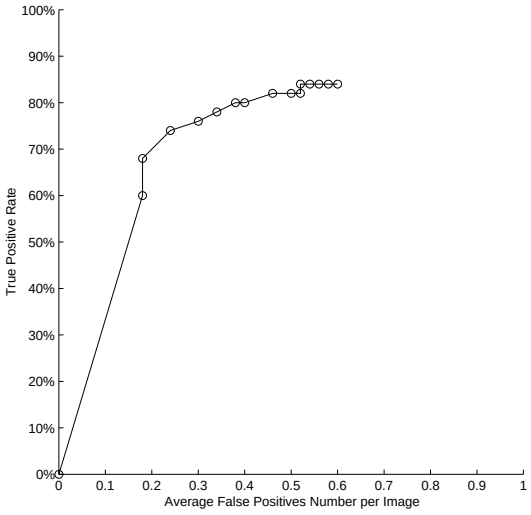

Fig. 8. FROC, performance in the subset of S2 containing only portraits

4.2. Evaluation per image

This evaluation was done on the per image basis. It measured the performance of the method in classifying an image

as a portrait or a non-portrait. The result is described with a ROC curve in the Figure 9. Each point of of the curve was a pair of *sensitivity* defined as

$$\frac{\text{number of images correctly classified as portraits}}{\text{total number of portrait images in the set}} \quad (13)$$

and *false positive rate* defined as

$$\frac{\text{number of images falsely classified as portraits}}{\text{total number of non-portrait images in the set}} \quad (14)$$

An image was classified as a portrait when there was a face region present. As it can be seen from the resulting ROC curve, the method can achieve the sensitivity level of 90% at the false positive rate of 32% and the sensitivity level of 82% at the false positive rate of 16%.

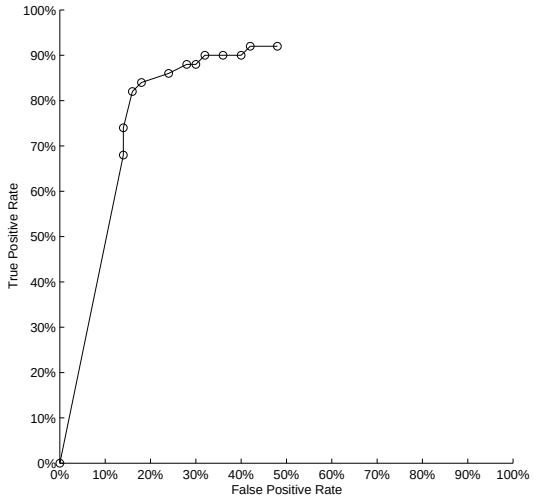

Fig. 9. ROC, performance on image annotation

4.3. Discussion

The true positive rate (sensitivity) of the method could not achieve the ideal 100% rate because the segmentation step failed to identify some faces. In the training set S1 46 out of 55 faces were segmented correctly and in the set S2 47 out of 50 faces. The low resolution of images contributed to the segmentation errors.

The method could be improved in several ways. After color segmentation, region merging can be applied to get regions which better approximate the face ellipse because the face can be broken into several region (e.g. forehead, cheeks, chin) after the edge identification. The edge identification can be improved using other edge detectors since in some cases it did not help to split the region and in other cases undesired splitting occurred. For identifying the foreground pixels an other thresholding method can be used. The ellipse fitting step can be changed and Hough transform can be used for finding the ellipses.

After detecting the skin-colored pixels various assumptions about portraits can be applied to remove some regions. For example regions touching the borders, regions in the lower part of the image, or regions too big can be removed from the set of face candidates. An analysis of the portraits in our database showed that the size of a bounding box of a face varied from 1.4% to 35.8% of the image.

5. CONCLUSION

We have presented a method for automatic identification of portraits in art images database. The method consists from a multistage algorithm which segments the images based on color, intensity and edge information. After the segmentation features of the resulting regions are collected and a neural network is trained to distinguish between face and non-face regions. This classification is then used to identify an image as a portrait or non-portrait.

The results are very encouraging for further development of the method.

6. REFERENCES

[1] Ming-Hsuan Yang, David J. Kriegman, and Narendra Ahuja, "Detecting faces in images: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34–58, 2002.

[2] Eli Saber, A. Murat Tekalp, Reiner Eschbach, and Keith Knox, "Automatic Image Annotation Using Adaptive Color Classification," *Graphical Models and Image Processing*, vol. 58, no. 2, pp. 115–126, March 1996.

[3] Theo Gevers, Frank Aldershof, and A.W.M. Smeulders, "Classification of images on internet by visual and textual information," *IST/SPIE Electronic Imaging, Internet Imaging*, vol. 3964, pp. 16–27, January 2000.

[4] Felix Toran-Marti, "Matlab code for optimal threshold," .

[5] Ioannis Pitas, *Digital Image Processing Algorithms*, Prentice Hall, 1993.

[6] Radim Halir and Jan Flusser, "Numerically stable direct least squares fitting of ellipses," in *Proceedings of WSCG'98*, Feb. 1998, Plzen-Bory, Czech Republic.

[7] Howard Demuth and Mark Beale, *Neural Network Toolbox: For use with MATLAB: User's Guide*, The Mathworks, 1993.