Automatic Eyes and Nose Detection Using Curvature Analysis

J. Matías Di Martino(✉), Alicia Fernández, and José Ferrari

Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay
{matiasdm,alicia,jferrari}@fing.edu.uy

Abstract. In the present work we propose a method for detecting the nose and eyes position when we observe a scene that contains a face. The main goal of the proposed technique is that it capable of bypassing the 3D explicit mapping of the face and instead take advantage of the information available in the Depth gradient map of the face. To this end we will introduce a simple false positive rejection approach restricting the distance between the eyes, and between the eyes and the nose. The main idea is to use nose candidates to estimate those regions where is expected to find the eyes, and vice versa. Experiments with Texas database are presented and the proposed approach is testes when data presents different power of noise and when faces are in different positions with respect to the camera.

Keywords: Landmark detection · Differential 3d reconstruction · Nose tip detection · Eyes detection

1 Introduction

One of the most popular and challenging problems in the field of pattern recognition and computer vision consists is the analysis and recognition of human faces[1]. It has many applications such as security control and prevention, medical and biometrical analysis or gesture understanding. In the last decade, lot of research included three-dimensional (3D) face information to improve recognition rates and make the methods more robust to pose, gesture and illumination variations[3]. See e.g. the work of Chang et al. [6], Faltemier et al. [10], Mahoor et al. [15], and Li et al. [14]. Most of these approaches, uses features of the face collected from the eyes forehead and nose regions, and hence they require in a initial step to find the position of the eyes and nose on the input images. To achieve a robust localization of the eyes and nose position, we follow a curvature approach (see e.g.[8] and references therein). This methods is very efficient for detecting those possible nose and eyes candidates, despite that, it is important to solve the problem of removing false positive candidates. To solve this issue we will follow a message passing methodology; the main idea behind it is to use nose candidates to estimate those regions where is expected to find the eyes, and vice versa.

A. Pardo and J. Kittler (Eds.): CIARP 2015, LNCS 9423, pp. 271–278, 2015.
DOI: 10.1007/978-3-319-25751-8_33

In order to remove possible false detections many approaches has been proposed; for example, Li and Da [14] fitted a nose template to each nose candidate. Bronstein et al. [4] impose geometric relations between the candidates (e.g. that the nose apex is located between the two eyes, above the nose tip and within certain distance intervals). In the work of Faltemier et al. [10] the nose tip is found as the highest z value after the face surface is aligned using ICP to a template face. Chang et al. [6] impose that the eyes regions must present similar y and z value, and that the nose tip is found starting between the eyes landmark and moving *down* (i.e. along the x direction) along the face. Another example is the work of Colombo et al. [7] in which the idea is to look from all the nose and eyes candidates, all the possible triangles formed by one nose candidate and two eyes candidates. Then, each triangle is described by the distances between the three regions composing it; finally, triangles with abnormal distances are rejected.

While some of the previously described methods make hypothesis regarding the position or orientation of the face (i.e. assuming than the nose is down the two eyes), others are computationally expensive (i.e. requiring ICP registration or checking all possible combination of nose and eyes triangles). But what these methods have in common is that they use the explicit three-dimensional representation of the face, in the present work we aim to bypass the 3D mapping of the face and instead take advantage of the information available in the Depth gradient map of the face. To this end we will introduce a simple false positive rejection approach -that do not need the explicit 3D mapping of the face- and instead restricts the distance (in the image plane) between the eyes, and between the eyes and the nose.

2 Proposed Approach

2.1 Find Nose and Eyes Candidates

Let S be the surface defined by a twice differentiable real valued function $D : \Omega \to \mathbb{R}$, defined on an open set $\Omega \subseteq \mathbb{R}^2$:

$$S = \{(x, y, z) \,|\, (x, y) \in \Omega;\ z \in \mathbb{R}; D(x, y) = z\}. \tag{1}$$

For every point $(x, y, D(x, y)) \in S$ we consider two different curvature measures, the Mean (H) and the Gaussian (K) curvature defined as [5,7]:

$$H(x, y) = \frac{(1 + D_y^2)\, D_{xx} - 2 D_x D_y D_{xy} + (1 + D_x^2)\, D_{yy}}{2\left(1 + D_x^2 + D_y^2\right)^{3/2}}, \tag{2}$$

$$K(x, y) = \frac{D_{xx} D_{yy} - D_{xy}^2}{\left(1 + D_x^2 + D_y^2\right)^2}. \tag{3}$$

Following the procedure described in [9] we compute the first derivatives of the scene depth. After that, to calculate $H(x, y)$ and $K(x, y)$ it is only necessary to compute the second order derivatives (D_{xx}, D_{yy} and D_{xy}) e.g. using finite differences.

Once the Mean and Gaussian curvatures are computed, it is possible to classify the different areas of the face according to its shape [2]. Depending on H and K values, points on the surface are classified. To remove smooth regions from the areas of interest a thresholding approach was followed[7,11], and then those points with high absolute value of K and H were isolated. For those points with $K > 0$ and $H > 0$ the eye candidate label was assigned, while those with $K > 0$ and $H < 0$ were selected as nose candidates. Figure 1 illustrates some of the nose and eyes candidates obtained.

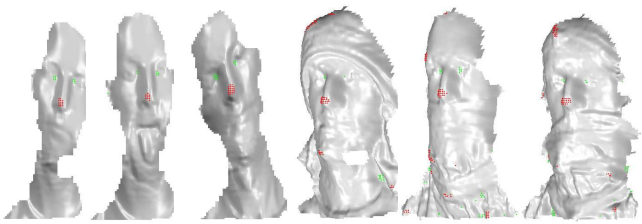

Fig. 1. Examples of nose (red dots) and eyes (green dots) candidates point detected.

2.2 Remove False Positive Detections

In first place, we estimate the distance between the eyes and between each eye and the nose tip using a training set. We approximate each distance distributions by a Gaussian distributions i.e.

$$d_{ee}(u) = \frac{1}{\sigma_{ee}\sqrt{2\pi}} e^{-\left(\frac{(u-\mu_{ee})^2}{2\sigma_{ee}^2}\right)} \quad (4)$$

$$d_{ne}(u) = \frac{1}{\sigma_{ne}\sqrt{2\pi}} e^{-\left(\frac{(u-\mu_{ne})^2}{2\sigma_{ne}^2}\right)}. \quad (5)$$

where d_{ee} and d_{ne} denote the distance between eyes and the distance between the nose and each eye, respectively. The mean and variance parameters (μ and σ^2) were estimated by fitting the distributions given by Eqs. (4) and (5) to the distributions obtained from a given training set.

Once distance distributions are obtained, they are used to *propagate* where it is feasible to find the nose/eyes considering each of the others nose and eyes candidates.

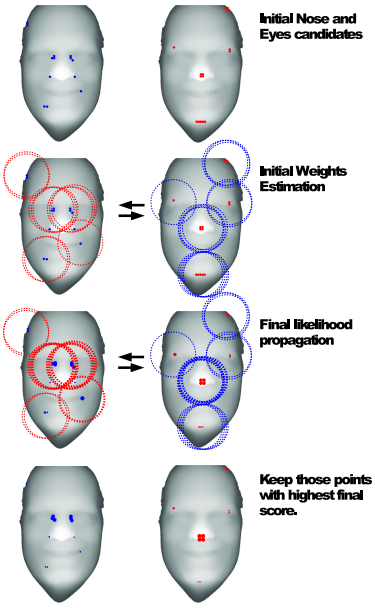

Fig. 2. Illustration of the steps followed to remove false positive detections. The size of each candidate point illustrates the likelihood of that point at each step.

$$\mathcal{K}_{ne}(x,y) = d_{ne}\left(\sqrt{x^2+y^2}\right) \quad \text{and} \quad \mathcal{K}_{ee}(x,y) = d_{ee}\left(\sqrt{x^2+y^2}\right). \tag{6}$$

The next step consists in estimating in which areas is more suitable to have the nose and eyes candidates so we can keep the set of points with higher probability to be the true ones. For this we follow four basic steps -illustrated in Fig. 2-:

1. Estimate initial *nose-likelihood* ($\mathcal{N}$) and *eye-likelihood* ($\mathcal{E}$) functions:

$$\mathcal{N}_0(x,y) = \frac{1}{\#e}\sum_{i=1}^{\#e} \delta_{x_i,y_i} * \mathcal{K}_{ne}(x,y) \tag{7}$$

$$\mathcal{E}_0(x,y) = \left(\frac{1}{\#e}\sum_{i=1}^{\#e} \delta_{x_i,y_i} * \mathcal{K}_{ee}(x,y)\right) \cdot \left(\frac{1}{\#n}\sum_{j=1}^{\#n} \delta_{x_j,y_j} * \mathcal{K}_{ne}(x,y)\right) \tag{8}$$

   where $\#e$, $\#n$ is the number of eyes and nose candidates respectively, δ_{x_i,y_i} is the Dirac delta function at each nose/eye candidate location (x_i, y_i); and $\mathcal{K}_{ne/ee}$ the Kernels defined above.
2. Compute the nose-likelihood and eyes-likelihood distributions as,

$$\mathcal{N}(x,y) = \frac{1}{\#e}\sum_{i=1}^{\#e} \mathcal{E}_0(x_i,y_i)\delta_{x_i,y_i} * \mathcal{K}_{ne}(x,y) \tag{9}$$

$$\begin{aligned}\mathcal{E}(x,y) = &\left(\frac{1}{\#e}\sum_{i=1}^{\#e} \mathcal{E}_0(x_i,y_i)\delta_{x_i,y_i} * \mathcal{K}_{ee}(x,y)\right) \\ &\cdot \left(\frac{1}{\#n}\sum_{j=1}^{\#n} \mathcal{N}_0(x_i,y_i)\delta_{x_j,y_j} * \mathcal{K}_{ne}(x,y)\right).\end{aligned} \tag{10}$$

   This second step is illustrated in the third row of Fig. 2, the size of each nose/eye candidate point illustrates the value of $\mathcal{N}_0/\mathcal{E}_0$ and the information of each candidate will be propagated with different weight.
3. Finally, the nose/eyes candidate points with higher $\mathcal{N}/\mathcal{E}$ are kept as the true nose/eyes locations.

Figure 3 shows an example of the input image (a), depth gradient field (b-c) [estimated following [9]], eyes candidate points (display as blue dots) overlapped to the computed $\mathcal{E}(x,y)$ (d), nose candidate points (display as blue dots) overlapped to the computed $\mathcal{N}(x,y)$ (e), and finally (f) the output nose and eyes locations obtained .

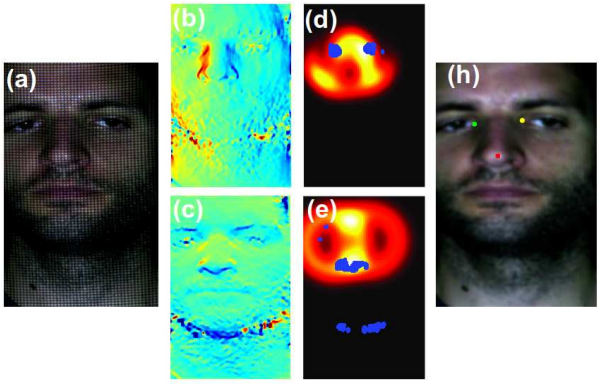

Fig. 3. This figure illustrates the nose and eyes detection procedure. (a) Input image. (b-c) x and y depth partial derivatives obtained by measuring fringes' deformation. (d-e) eyes and nose candidates obtained by curvature thresholding (as blue dots) overlapped with the estimated functions $\mathcal{E}(x, y)$ and $\mathcal{N}(x, y)$ respectively. (h) Nose and eyes located in the image.

3 Experiments and Evaluation

In this section, we perform a set of experiments using the Texas 3D Face Recognition Database [12,13]. This Database contains 1149 pairs of high resolution, pose normalized, preprocessed, and perfectly aligned color and range images of 118 subjects. Additionally, it includes the locations of manually marked anthropometric facial fiducial points which will be used as ground truth. For the experiments, we split the database in two sets: Train and Test sets. The first one was used to estimate the parameters, and the test set was used for evaluation once all the parameters were trained.

In a first experiment, we want to evaluate the robustness of the proposed methodology when we have different power of noise in the input gradient field. Figure 5 shows the accuracy obtained (over the test set) for different levels of noise . The noise added to the input gradient field was Gaussian with zero mean; as we can see, when the variance of noise distribution was below the 20% of the maximum of the signal, the error in both nose and eyes detection was below the 10%.

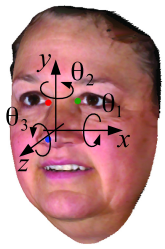

Fig. 4. Definition of θ_1, θ_2 and θ_3.

In a second experiment, we want to test the robustness of the proposed technique when the faces are in a non-frontal position with respect to the camera. We define the angles θ_1, θ_2 and θ_3 as illustrated in Figure 4. Recall that we are measuring the projected euclidean distance in the 3D space over the image plane, because of this it is expected that for databases

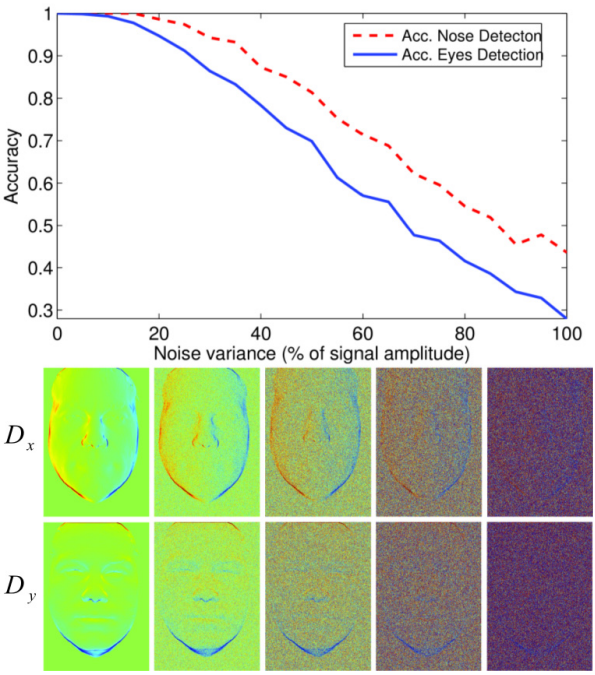

Fig. 5. Result obtained over the test set varying the power of the noise added to the input gradient field.

with larger pose variations, K_{ne} and K_{ee} kernels will become wider and hence more errors should be expected[1].

A new dataset was generated by randomly rotating faces (of the Texas Database) and projecting them back to the image plane. Again this database was spitted in train and test sets to avoid over fitting. The test set was divided according to faces' position, hence we were able to measure the accuracy for different face orientation (i.e. different values of θ_1, θ_2 and θ_3). Figure 6 shows the accuracy for test sets with faces in different positions. Accuracies on eyes and nose recognition are displayed for $|\theta_i| \in [0^o, 55^o]$ $i = 1..3$, the range images below the x axis illustrates the pose obtained with the corresponding value of θ_i. As we can see the proposed approach is independent with respect to the value of θ_3 while accurate results can be achieved when we restrict θ_1 and θ_2 to the interval $[-30^o, 30^o]$, for larger values of θ_1 and θ_2 the performance drops significantly.

[1] In this direction we think there is room for interesting future work. For example, an improvement could be to estimate the distance in the 3D space from the distances in the image plane plus the information available in the Depth gradient field (D_x and D_y).

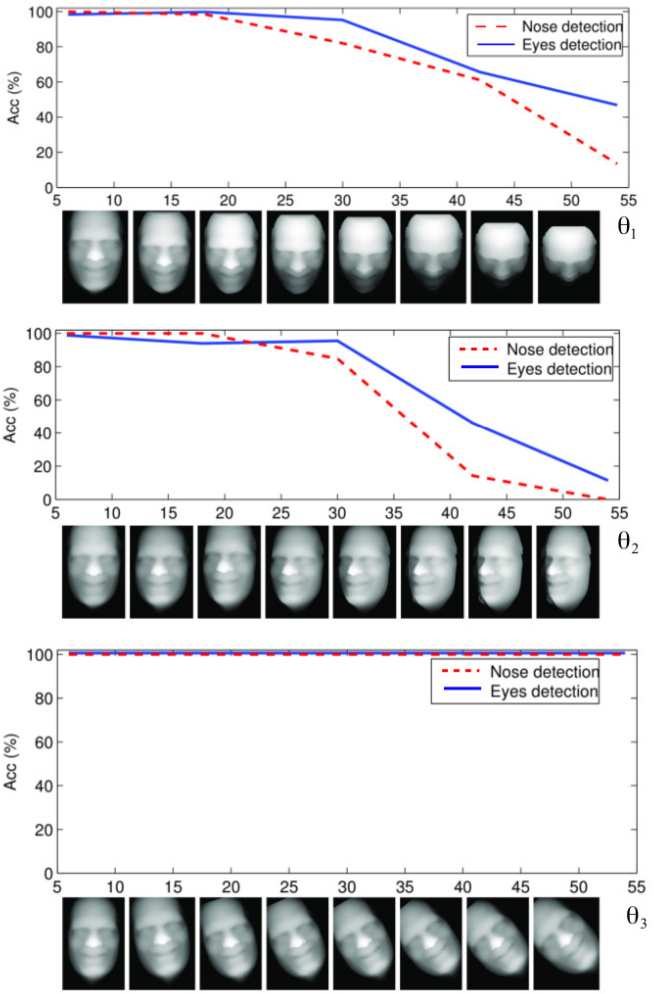

Fig. 6. Results when varying face position.

4 Conclusions

This work proposes an efficient approach to find the location of the eyes and nose position using curvature analysis but bypassing the explicit 3D mapping of the face. In addition a method for removing the false candidates was proposed, which constrain the inter-distances between the set of candidate point to find a robust solution. The proposed method was tested in a public database; in particular the effect of noise in the input gradient field and the position of the face on system performance was studied. In future work we plan to include depth gradient information in order to achieve a more robust estimation of the distance

between candidate point, and hence improve the robustness of the method under arbitrary rotations of the face. Other interesting line of research is to apply the proposed framework for tracking some facial landmarks which could be analyzed e.g. for gesture recognition.

Acknowledgments. The authors thank PEDECIBA, CSIC, and ANII for their financial support.

References

1. Abate, A.F., Nappi, M., Riccio, D., Sabatino, G.: 2D and 3D face recognition: A survey. Pattern Recognition Letters **28**(14), 1885–1906 (2007)
2. Besl, P.J., Jain, R.C.: Invariant surface characteristics for 3D object recognition in range images. Computer Vision, Graphics, and Image Processing **33**(1), 33–80 (1986)
3. Bowyer, K.W., Chang, K., Flynn, P.: A survey of approaches and challenges in 3D and multi-modal 3D+2D face recognition. Computer Vision and Image Understanding **101**(1), 1–15 (2006)
4. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Three-Dimensional Face Recognition. International Journal of Computer Vision (IJCV) **64**(1), 5–30 (2005)
5. do Carmo, M.: Differential Geometry of Curves and Surfaces. Pearson Education Canada (1976)
6. Chang, K.I., Bowyer, K.W., Flynn, P.J.: Multiple nose region matching for 3D face recognition under varying facial expression. IEEE Transactions on Tattern Analysis and Machine Intelligence **28**(10), 1695–1700 (2006)
7. Colombo, A., Cusano, C., Schettini, R.: 3D face detection using curvature analysis. Pattern Recognition **39**(3), 444–455 (2006)
8. Di Martino, J.M., Fernández, A., Ferrari, J.A.: 3D curvature analysis with a novel one-shot technique. IEEE International Conference on Image Processing (ICIP2014) (2014)
9. Di Martino, J.M., Fernández, A., Ferrari, J.A.: One-shot 3D gradient field scanning. Optics and Lasers in Engineering **72**, 26–38 (2015)
10. Faltemier, T.C., Bowyer, K.W., Flynn, P.J., Member, S.: A Region Ensemble for 3-D Face Recognition **3**(1), 62–73 (2008)
11. Gordon, G.G.: Face recognition based on depth maps and surface curvature. SPIE 1570, Geometric Methods in Computer Vision, pp. 234–247, Sep 1991
12. Gupta, S., Castleman, K., Markey, M., Bovik, A.: Texas 3d face recognition database. In: 2010 IEEE Southwest Symposium on Image Analysis Interpretation (SSIAI), pp. 97–100, May 2010
13. Gupta, S., Markey, M., Bovik, A.: Anthropometric 3D face recognition. International Journal of Computer Vision **90**(3), 331–349 (2010)
14. Li, X., Da, F.: Efficient 3D face recognition handling facial expression and hair occlusion. Image and Vision Computing **30**(9), 668–679 (2012)
15. Mahoor, M.H., Abdel-Mottaleb, M.: Face recognition based on 3D ridge images obtained from range data. Pattern Recognition **42**, 445–451 (2009)