ESTIMATING THE QUALITY OF FACE LOCALIZATION FOR FACE VERIFICATION

Yann Rodriguez *Fabien Cardinaux* *Samy Bengio* *Johnny Mariéthoz*

IDIAP
CP 592, rue du Simplon 4
1920 Martigny, Switzerland
{rodrig,cardinau,bengio,marietho}@idiap.ch

ABSTRACT

Face localization is the process of finding the exact position of a face in a given image. This can be useful in several applications such as face tracking or person authentication. The purpose of this paper is to show that the error made during the localization process may have different impacts depending on the final application. Hence in order to evaluate the performance of a face localization algorithm, we propose to *embed* the final application (here face verification) into the performance measuring process. Moreover, in this paper, we estimate this embedding using either a multilayer perceptron or a K nearest neighbor algorithm in order to speedup the evaluation process. We show on the BANCA database that our proposed measure best matches the final verification results when comparing several localization algorithms, on various performance measures currently used in face localization.

1. INTRODUCTION

Face localization is the process of finding the exact position of a face in a given image. It is generally used as an important step in several applications such as face tracking or person authentication. Unfortunately, analyzing the quality of a face localization algorithm is not straightforward, and no universal criterion has been acknowledged in the literature for this purpose. We argue that such a criterion does not exist and propose instead the use of a criterion specific for each application the localization algorithm is designed for. More precisely, this paper concentrates on a face verification task. In that context, a good localization algorithm is the one that minimizes the number of errors made by the verification algorithm. Knowing that verification in itself is not error-free, we propose here a methodology to estimate the verification errors given the errors made by the localization algorithm. We then propose to estimate this measure using either a multilayer perceptron or a K nearest neighbor algorithm.

We present here the results of several experiments conducted on the benchmark BANCA database [1], comparing three different face localization algorithms in the context of a face verification task, using the same verification algorithm. We will show that our proposed measure best matches the final verification performance induced by several localization algorithms.

The paper is organized as follows. Section 2 presents classical measures used in the literature in order to evaluate the quality of a face localization algorithm. Section 3 presents our idea, which consists in estimating the error made by the verification process given the error made by the localization process. Section 4 presents the framework (database, face verification and face localization systems) used to evaluate our proposed method. Section 5 presents the results of the experiments, and finally Section 6 concludes the paper.

2. PERFORMANCE MEASURES FOR FACE LOCALIZATION

Direct comparison of face localization systems is a very difficult task, mainly because there is no clear definition of what a good face localization means. Most of the papers found in the literature generally only provide localization and error rates, but rarely mention the way they count a correct/incorrect hit to compute these rates. Furthermore, when reported, this criterion is usually not clearly described. Sometimes, faces are even identified manually by humans [2]. This lack of uniformity makes results difficult to compare and reproduce.

Recently, Jesorsky *et al.* [3] introduced a relative error measure based on the distances between the detected and the expected (ground-truth) eye center positions. Let C_l (respectively C_r) be the true left (resp. right) eye coordinate position and let $\tilde{C}_l$ (resp. $\tilde{C}_r$) be the left (resp. right) eye position estimated by the localization module. Jesorsky's measure can be written as

$$d_{eye} = \frac{\max(d(C_l, \tilde{C}_l), d(C_r, \tilde{C}_r))}{||C_l - C_r||} \tag{1}$$

where $d(a, b)$ is the Euclidean distance between positions a and b. A successful localization is accounted if $d_{eye} < 0.25$ (which corresponds approximately to half the width of an eye).

One drawback of this measure is that it is not possible to differentiate errors in translation, rotation and scale. More recently, Popovici *et al.* [4] proposed a new parametric scoring function which overcomes limitations of Jesorsky's measure. Parameters can be tuned to more precisely penalize each type of error. Let $\overrightarrow{dx_l}$ (resp. $\overrightarrow{dx_r}$) be the x translation of the obtained left (resp. right)

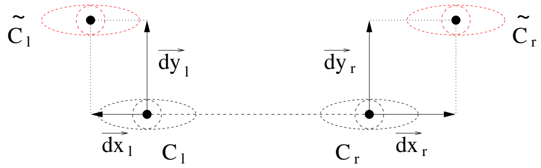

Fig. 1. Summary of current basic measurements made in face localization.

eye position, and let $\overrightarrow{dy_l}$ (resp. $\overrightarrow{dy_r}$) be the y translation of the obtained left (resp. right) eye position. Popovici *et al.* then define three[1] basic Δ measures representing the difference in x translation, y translation, and scaling, as follows:

$$\begin{aligned}\Delta_x &= \frac{(\overline{dx_l} + \overline{dx_r})}{2 \cdot ||C_l - C_r||} \ , \\ \Delta_y &= \frac{(\overline{dy_l} + \overline{dy_r})}{2 \cdot ||C_l - C_r||} \ , \\ \Delta_s &= \frac{||\tilde{C}_l - \tilde{C}_r||}{||C_l - C_r||} \ .\end{aligned}$$

where $\overline{dx}$ is the algebraic measure of vector $\overrightarrow{dx}$. All these measures are summarized in Figure 1. Note that both the choice of Jesorsky's threshold (0.25) and Popovici's parameters still remain subjective.

In this paper, we argue that a universal objective measure to evaluate face localization does not exist. A given localized face may be correct for the task of initializing a face tracking system, but may not be accurate enough for a face verification system. We therefore think that there is no absolute definition of what a *good face localization* is. We rather suggest to look for an application-dependent measure representing the final task.

Moreover, in the context of face verification, there have been several empirical evidences [5] showing that the verification score obtained with a perfect (manual) localization is significantly better than the verification score obtained with a not-so-perfect (automatic) localization, which shows the importance of measuring accurately the quality of a face localization algorithm for verification.

3. APPROXIMATE FACE VERIFICATION PERFORMANCE

As explained in Section 2, instead of searching for a universal cost function assessing the quality of a face localization algorithm, we propose to estimate a specific cost function adapted to the target task. We here concentrate on the task of face verification, hence a good face localization algorithm in that context is a module which produces a localization such that the expected error of the face verification module is minimized. More formally, let $\mathbf{x}_i$ be the input vector describing the face of an access i (defined more precisely in Section 4), $\mathbf{y}_i = \mathrm{FL}(\mathbf{x_i})$ be the output of a face localization algorithm applied to $\mathbf{x}_i$ (generally in terms of eye positions), $z_i = \mathrm{FV}(\mathbf{y}_i)$ be the decision taken by a face verification algorithm (generally accept or reject the access) and $\epsilon = \mathrm{Error}(z_i)$ be the error generated by this decision. The ultimate goal of a face localization algorithm is thus to minimize the following cost function:

$$\mathrm{Cost} = \sum_i \mathrm{Error}(\mathrm{FV}(\mathrm{FL}(\mathbf{x}_i))) \ . \tag{2}$$

One solution could thus be to embed all subsequent functions (FV and Error) into a single box and estimate this box using some universal approximator:

$$\mathrm{Cost} = \sum_i f(\mathrm{FL}(\mathbf{x}_i); \theta) \tag{3}$$

[1] In fact, Popovici *et al.* define a fourth measure for rotation, but we will assume in this paper that the eyes have been perfectly aligned horizontally, i.e. $\overrightarrow{dy_l} = \overrightarrow{dy_r}$.

where $f(\cdot; \theta)$ is a parametric function that would replace the rest of the process following localization using parameters θ. In this paper, we consider two such functions $f(\cdot)$: a multilayer perceptron (MLP) and a K nearest neighbor (KNN) algorithm [6]. In order to be independent of the precise localization of the eyes, we have in fact slightly modified this approach by changing function $f(\cdot)$ inputs to instead contain the error made by the localization algorithm in terms of very basic measures: Δ_x, Δ_y and Δ_s, as described in Section 2. Let $\mathrm{GT}(\mathbf{x}_i)$ be the ground-truth eyes position of $\mathbf{x}_i$ and $\mathrm{Err}(\mathbf{y_i}, \mathrm{GT}(\mathbf{x_i}))$ be the function that produces the face localization error vector; we thus have

$$\mathrm{Cost} = \sum_i f(\mathrm{Err}(\mathrm{FL}(\mathbf{x}_i), \mathrm{GT}(\mathbf{x_i})); \theta) \ . \tag{4}$$

In order to train such function $f(\cdot)$, we used the following methodology. First of all, in order to cover the space of localization errors, we create artificial examples based on all available training accesses. The training examples of $f(\cdot)$ are thus uniformly generated by adding small perturbations (localization errors) bounded by a reasonable range. For each generated example, a verification is performed and a corresponding target value of 1 (respectively 0) is assigned when a verification error appears (respectively does not appear).

Preliminary experiments using this setup revealed some difficulties in the successful training of the MLP, mainly due to the generated examples being noisy (for the same localization perturbation, but a different access, the verification system was not consistent, yielding an inconsistent expected output of the MLP as well). In order to solve this problem, the localization error space was partitioned into several smaller subspaces into which the expected output was computed as the average of the original outputs of each example of the subspace and the input was replaced by the center of the subspace. This corresponds to a kind of smoothing of the expected output in order to remove the inherent noise of the data.

Using KNN to estimate $f(\cdot)$ did not yield any problem apart from the fact that it is significantly slower than MLPs during testing (although many times faster than using the actual face verification system).

Finally, in order to obtain an estimation of the expected classification error of a given face localization algorithm, we simply average the output of $f(\cdot)$ for all test examples.

4. BASELINE SYSTEM

In this section, we describe the environment that was used to evaluate the quality of our system. We first describe the database, then the verification system, and finally three different localization systems that were compared in this paper.

4.1. The BANCA Database

The BANCA database [1] was designed to test multi-modal identity verification with various acquisition devices under several scenarios (controlled, degraded and adverse). In our experiments we use face images from the French and English sections, each containing 52 subjects.

Each subject participated in 12 recording sessions in different conditions and with different cameras. Each of these sessions contains two video recordings: one true client access and one impostor attack. Five "frontal" face images have been extracted from each video recording. Following the "BANCA Experimental Protocol", these five images should be considered as a single access; however, in order to estimate and test our cost function, we used each image as an independent access. According to [1], we decided to follow protocol P, which appears to be the most realistic one.

4.2. The Face Verification System

A face verification system (FV) usually consists in image normalization and feature extraction followed by classification. In this study we use a FV based on local features and Gaussian Mixture Models (GMMs) [5], briefly described as follows.

First, a 80×64 (rows $\times$ columns) face window is cropped out (based on the result of the face localization process); each face window contains the face area from the eyebrows to the chin; moreover, the location of the eyes is the same in each face window (via geometric normalization). Histogram equalization is used afterward to normalize the face images photometrically. We then extract *DCTmod2* features vector $\mathbf{X}$ from each face image [7]. The resulting feature vectors are 18-dimensional for each local block, and there are 17x13=221 overlapping blocks per image.

The face verification system was implemented using a Gaussian Mixture Model (GMM) technique similar to those used in text-independent speaker verification systems. A generic GMM is trained with the features computed on several faces (non-client specific), in order to maximize $p(\mathbf{X}|\Omega)$, the likelihood of a face $\mathbf{X}$ given the generic GMM Ω, for all $\mathbf{X}$ of the training database. This GMM is then adapted for each client i in order to produce a model of $p(\mathbf{X}|C_i)$, the likelihood of a face $\mathbf{X}$ given the client model C_i. The ratio between these likelihoods represents the score of the verification model, which is then compared to a threshold in order to take a final decision.

4.3. The Face Localization Systems

We compare here three different face localization systems. The first one (hereafter called *Weak Boosting*) is based on the well-known Viola-Jones [8] face detector, based on boosting and fast-to-compute Haar-like features. For comparison purposes (we wanted this detector to be not so accurate, in order to span a large range of localization performances), we used a very basic cascade architecture of only height stages.

The second localization algorithm (hereafter called Boosting) is based on a cascade of boosted classifiers using an extended set of Haar-like features [9]. The source code is part of OpenCV, available publicly at http://sourceforge.net/projects/opencvlibrary/.

The third localization algorithm (*MLP*) is composed of two sequentially connected neural networks. It is mainly inspired by Rowley's face detector [10].

5. EXPERIMENTS AND RESULTS

We used the French part of the BANCA database to generate examples used to estimate $f(\cdot)$, the verification errors given localization errors. For each of the 2730 available images, 50 (10 horizontal and vertical shifts, at 5 different scales) localization errors were generated randomly in a predefined interval: $[-1, 1]$ for Δ_x and Δ_y (position errors) and $[0.5, 1.5]$ for Δ_s (scaling error). The total number of generated examples is thus 136500. For the MLP, the space is then divided into 729 regions (9 per each of the 3 dimensions) in order to compute average (smoothed) targets. The

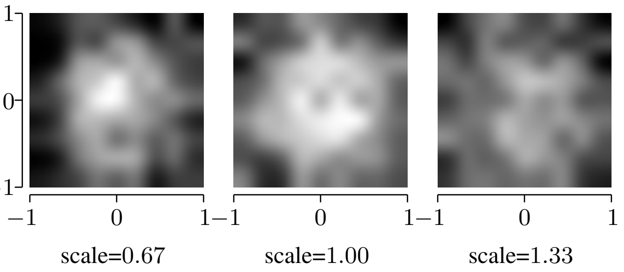

Fig. 2. Verification error with respect to the localization error, on the BANCA French dataset, protocol P. Each point in these subfigures represents a verification error (white represents a small error, while black is the highest error) for a given translation error. Three different scaling errors are represented.

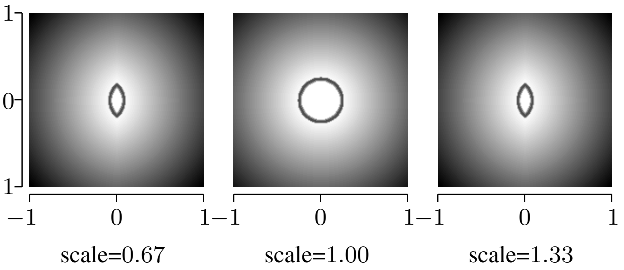

Fig. 3. Jesorsky's measure. Compare this with the actual verification errors shown in Figure 2. The black circle represents the decision threshold chosen by Jesorsky.

capacity of the models is tuned using a K-fold cross-validation procedure on the training set.

Figure 2 shows the verification error with respect to the localization error, for three different scales (the color represents the probability of verification error with respect to the input, for each translation error) using the BANCA French dataset, protocol P. We can observe that the verification error is not linear, nor uniform, with respect to the inputs, as implicitly supposed in Jesorsky's measure. Moreover, this shows that the verification system, based on a GMM approach, is quite robust to translation errors.

Table 1 compares the localization measures (the smaller the better for all compared measures) obtained using Jesorsky's method, the proposed method using either the MLP or KNN, and the actual verification error rate[2], on all the accesses of the BANCA English section (protocol P).

All measure techniques (Jesorsky, the MLP and the KNN) failed to correctly rank the *MLP* and *Boosting* localization algorithms. Scores obtained by our proposed method is however very similar to the ones obtained by the true verification rate. Our estimate is quite realistic.

Table 2 shows the mean absolute error (MAE) between the actual verification error rate and the values obtained by each of the three localization measures using three different face localization algorithms as well as assuming a perfect localization (ground-truth). We see that our method easily outperforms Jesorsky's abil-

[2]the number of verification errors divided by the total number of accesses.

Localizators	Jesorsky	MLP	KNN	Verification
ground-truth	0	22.1	20	23
MLP	5.4	24.45	28.26	28.15
Boosting	4.6	23.8	26.9	28.8
Weak-Boosting	44.7	31.5	31.4	33.7

Table 1. Comparison of three performance measure methods (in terms of error rate) for three localization systems as well as for a perfect localization (ground-truth). The last column contains the actual verification error rate.

Jesorsky	MLP	KNN
26.6	2.95	1.82

Table 2. The *mean absolute error* (MAE) between three performance measure methods and the actual verification error rate, averaged over the three localization systems and the ground-truth.

ity to estimate the quality of a localization algorithm for the task of face verification.

In order to understand why Jesorsky's measure performed so badly, we show in Figure 3 the score computed by Jesorsky's measure for the same translation and scale errors as those of Figure 2. Moreover, we show in black the decision boundary ($d_{eye} < 0.25$) used by Jesorsky to accept or reject a localization. Comparing these figures, we see that Jesorsky basically fails to represent correctly errors due to scaling.

6. CONCLUSION

In this paper, we have proposed a novel methodology to compare face localization algorithms in the context of a particular application, namely face verification. We have proposed a method to estimate the verification errors induced specifically by the use of a particular face localization algorithm. This measure can then be used to compare more precisely several localization algorithms. We tested our proposed measure using the BANCA database on a face verification task, comparing three different face localization algorithms. Results show that our measure does indeed capture more precisely the differences between localization algorithms (when applied to verification tasks), which can be useful to select an appropriate localization algorithm.

In fact, one can view the process of training a localization system as a selection procedure where one simply selects the best localization algorithm according to a given criterion. In that respect, an interesting future work could concentrate on the use of such a measure to effectively *train* a face localization system for the specific task of face verification.

7. ACKNOWLEDGMENTS

This research has been partially carried out in the framework of the Swiss NCCR project (IM)2. It was also supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778, funded in part by the Swiss Federal Office for Education and Science (OFES). This publication only reflects the authors' views. All experiments were done using the *Torch* package [11]. The authors would like to thank S. Marcel for providing his face detector and C. Sanderson for useful suggestions.

8. REFERENCES

[1] E. Bailly-Baillière, S. Bengio, F. Bimbot, M. Hamouz, J. Kittler, J. Mariéthoz, J. Matas, K. Messer, V. Popovici, F. Porée, B. Ruiz, and J.-P. Thiran, "The BANCA database and evaluation protocol," in *4th International Conference on Audio- and Video-Based Biometric Person Authentication, AVBPA*, Guilford, UK, 2003.

[2] M.-H. Yang and D. Roth and N. Ahuja, "A SNoW-Based Face Detector," in *Advances in Neural Information Processing Systems, S. A. Solla, T.K. Leen and K.-R. Muller (eds), pp. 855-861, MIT Press*, 2000.

[3] O. Jesorsky, K. Kirchberg, and R. Frischholz, "Robust face detection using the Hausdorff distance," in *Proceedings of Audio and Video based Person Authentication*, 2001, pp. 90–95.

[4] V. Popovici, Y. Rodriguez, J.-P. Thiran, and S. Marcel, "On performance evaluation of face detection and localization algorithms," Technical Report 03-80, IDIAP, Martigny, Switzerland, 2003.

[5] F. Cardinaux, C. Sanderson, and S. Marcel, "Comparison of MLP and GMM Classifiers for Face Verification on XM2VTS," in *4th International Conference on Audio- and Video-Based Biometric Person Authentication, AVBPA*, Guilford, UK, 2003, pp. 911–920.

[6] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.

[7] C. Sanderson and K. K. Paliwal, "Fast features for face authentication under illumination direction changes," *Pattern Recognition Letters*, vol. 24, no. 14, pp. 2409–2419, 2003.

[8] P. Viola and M. Jones, "Robust Real-time Object Detection," in *IEEE ICCV Workshop on Statistical and Computational Theories of Vision*, 2001.

[9] R. Lienhart and J. Maydt, "An Extended Set of Haar-like Features for Rapid Object Detection," in *Proceedings of the IEEE Conference on Image Processing*, 2002, pp. 900–903.

[10] H. Rowley, S. Baluja and T. Kanade, "Rotation Invariant Neural Network-Based Face Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1998, pp. 38–44.

[11] R. Collobert, S. Bengio, and J. Mariéthoz, "Torch: a modular machine learning software library," Technical Report 02-46, IDIAP, Martigny, Switzerland, 2002.