LOCATING FACIAL LANDMARKS WITH BINARY MAP CROSS-CORRELATIONS

Jérémie Nicolle *Kévin Bailly* *Vincent Rapp* *Mohamed Chetouani*

Univ. Pierre & Marie Curie, ISIR - CNRS UMR 7222, F-75005, Paris - France
{*nicolle, bailly, rapp, chetouani*}*@isir.upmc.fr*

ABSTRACT

Precise facial landmark localization in still images is a key step for many face analysis applications, such as biometrics or automatic emotion recognition. In this paper, we propose a framework for facial point detection in frontal and near-frontal images. We introduce a new appearance model based on binary map cross-correlations that efficiently uses LBP and LPQ in a localization context. Inclusion of shape-related constraints is performed by a nonparametric voting method using relational properties within triplets of points, designed to correct outliers without losing precision for accurately detected points. We tested our system's performance on the widely used as benchmark BioID database obtaining state-of-the-art results. We also discuss evaluation metrics used to compare facial landmarking systems and which have been mixed up in recent literature.

***Index Terms* —** LBP, LPQ, facial landmarks, shape model, binary maps.

1. INTRODUCTION

The goal of facial landmarking is to precisely locate a set of key points in faces, delimiting the eyes, the eyebrows, the nose and the mouth. This task is particularly challenging because of the variations in head pose, morphology, expression and illumination. Most state-of-the-art methods combine appearance-based information with shape-related constraints to increase robustness. However, important differences exist among those methods, concerning features, image exploration techniques and shape-related constraints.

Feature choice is a key point in localization methods. A balanced trade-off must be found between performance and computation time. Particularly fast-to-compute features can be used (Milborrow and Nicolls [1] use an Active Shape Model based on gray levels and Cootes *et al.* [2] use Haar-like features). More discriminant features, like dense SIFT used by Belhumeur *et al.* [3], can lead to very precise results but are time-consuming. Because of their low computation time and their robustness towards illumination and blur, we chose to use LBP and LPQ and to include them in a new detection-oriented framework (details in §2.1).

Various image exploration techniques for point regression have been used in literature. Some methods estimate probability maps on large areas and predict point locations within these areas (dense exploration techniques). To estimate these probability maps, generative methods can be used (for example using a distance to a manifold estimated by PCA [1] or cross-correlating gray level mean patches [4]), as well as discriminant methods (for instance using a distance to an hyperplane estimated by SVM [5]). Other methods use more local areas to estimate point locations, that are potentially outside the areas used for feature extraction (sparse exploration techniques), as in [6] where the authors use SVR, or in [2, 7, 8] where random forests are used and lead to very fast algorithms. Nevertheless, sparse exploration based methods have the disadvantage of highly depending on initialization. To avoid this issue and because of the robustness it induces, we opted for a generative dense exploration technique.

Combining shape-related constraints with appearance-based detection has proven to increase robustness. A commonly used approach for modeling these constraints is to perform PCA to learn admissible deformations and optimize a cost function in the space of the parameters controlling these global deformations [9]. However, in order to stay within the learned manifold, many accurately located points may be displaced. We propose a nonparametric voting method based on relational properties within triplets of points that lets us introduce more local constraints correcting outliers without losing precision for accurately detected points (details in §2.2).

In this paper, we detail our facial feature detection algorithm and present our results on the well-know BioID database. Different evaluation metrics for facial feature localization algorithms can be used: one represents the cumulative distribution of image mean errors and the other the cumulative distribution of landmark errors. These curves have been mixed up in recent literature and have led to questions raised in [2] concerning the lack of distinctive "S" shape of some result curves. We propose a discussion about these evaluation metrics in §3.1.

2. FRAMEWORK

In our method, appearance-based regression is first performed cross-correlating LBP and LPQ binary maps with mean patches calculated on the learning database. Then, we iteratively correct potential outliers with shape-related constraints based on relational properties within triplets of points.

2.1. LBP-LPQ based probability maps

Local Binary Patterns (LBP) and Local Phase Quantization (LPQ) have proven their efficiency to characterize appearance [10, 11], mainly because of their robustness towards illumination and blur. Most facial analysis methods involving LBP or LPQ use them by computing histograms on different areas within the images [10, 11]. Histograms are commonly used because finding a relevant distance between LBP (or LPQ) values is not straightforward (appearance of pixels coded by close values can be very different). However, histograms do not keep information about the spatial distribution of the appearance within the areas of computation. Moreover, reducing the size of these areas can increase precision but raises the issue of finding an appropriate distance for sparse histograms. Thus, using them for precise localization can be difficult.
We propose a solution to efficiently use these features for precise point detection. In our method, we learn a mean patch for each point and each LBP (and LPQ) value and calculate probability maps by cross-correlating a few selected mean patches with the corresponding LBP (or LPQ) binary maps extracted from the test images.

2.1.1. LBP-LPQ mean patch learning

For each image of our learning database, we compute 2^8 binary LBP-maps and 2^8 binary LPQ-maps. The b^{th} map takes the value 1 for the pixels coded by the LBP (or LPQ) value b. We extract binary patches centered on each landmark and average them over all the images to obtain our mean patches. These mean patches give information about the probability of presence of pixels coded by each LBP or LPQ value. This way, we extract illumination and blur invariant features characterizing appearance around each landmark keeping precise spatial distribution related information that would have been lost by histogram computation. Figure 1 illustrates the extraction of LBP and LPQ mean patches for an area centered on the right eye.

2.1.2. Feature selection and weighting

In order to select the maps that are relevant for each of the landmarks and weight them appropriately, we calculate an accuracy score for all the 2^9 weak regressors on the learning set. Let $\{\mathbf{P}^{k,b}, k \in [\![1, n_p]\!], b \in [\![1, 2^9]\!]\}$ be the previously learned mean patches for each of the n_p landmarks and $\{\mathbf{M}^b(l)\}$ be

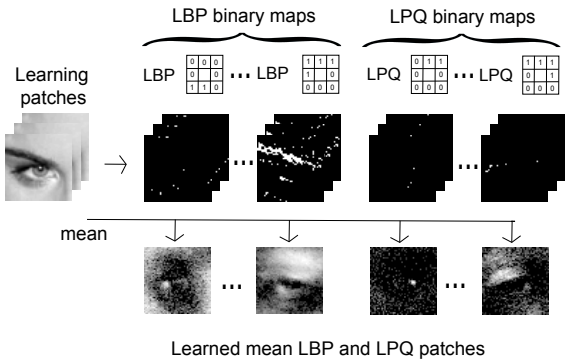

Fig. 1. Mean patch calculation process.

the binary maps for image l. Each patch gives an estimation of the location of the k^{th} landmark using:

$$\mathbf{p}_e^{k,b}(l) = argmax(\mathbf{M}^b(l) * \mathbf{P}^{k,b})$$

where $*$ denotes a normalized cross-correlation. We compute a response map for each landmark and each patch on the whole training database following:

$$R^{k,b} = \sum_{l=1}^{n_l} \delta_{(\mathbf{p}_e^{k,b}(l) - \mathbf{p}_t^k(l))}$$

where n_l is the number of images in the learning set, $\delta_{\mathbf{a}}(x, y)$ takes the value 1 when $(x, y) = \mathbf{a}$ and $\mathbf{p}_t^k(l)$ is the true location of the k^{th} landmark on image l. Then, we calculate the accuracy scores as: $S^{k,b} = \iint R^{k,b}.G$ where G is a gaussian with zero-mean, thus according more weight to the weak regressors that have often been placing the landmark close to its true location in the learning images. We use these accuracy scores to select the more relevant weak regressors for each point and appropriately weight them.

2.1.3. Probability map calculation

We perform these previous steps for two different sizes of mean patches. Large patches aim at roughly locating points using information about areas that can be relatively far and small patches aim at placing points more precisely, only using local information. Using these two sets of patches and their associated accuracy scores, we define our probability maps for the test images as follows:

$$\mathbf{J}^k(l) = \mathbf{I}^k_{large}(l) + \alpha.\mathbf{I}^k_{small}(l)$$

where

$$\mathbf{I}^k(l) = \sum_{j=1}^{n_k} S^{k,\mathbf{sel}^k(j)}.(\mathbf{M}^{\mathbf{sel}^k(j)}(l) * \mathbf{P}^{k,\mathbf{sel}^k(j)})$$

The parameter α sets the relative impact of the two different sizes of patches. The vectors **sel** contain the indices of

the previously selected maps and n_k is their length. These appearance-based maps give information about the probability of presence of each landmark and will be combined with attraction maps calculated using our shape model to make the final algorithm.

2.2. Triplet-based shape model

The purpose of the shape model in landmark localization is to correct potential outliers. We learn relational properties (ratios of distances and angles) within all triplets of points and select for each point the more stable triplets. During the test phase, we use a k-nearest neighbor algorithm to obtain a similar model and generate attraction maps used to correct, step by step, potential outliers.

2.2.1. Triplet model

For each triplet of points $\mathbf{t}_{k_1k_2k_3} = (\mathbf{p}_{k_1}, \mathbf{p}_{k_2}, \mathbf{p}_{k_3})$ of a learning image l, we calculate the ratio of distances and the angle between the vectors $\mathbf{v}_{k_2k_3} = \mathbf{p}_{k_3} - \mathbf{p}_{k_2}$ and $\mathbf{v}_{k_2k_1} = \mathbf{p}_{k_1} - \mathbf{p}_{k_2}$ to form

$$f^l(\mathbf{t}_{k_1k_2k_3}) = \frac{\|\mathbf{v}_{k_2k_1}\|}{\|\mathbf{v}_{k_2k_3}\|}.e^{i(\mathbf{v}_{k_2k_3}\widehat{,}\mathbf{v}_{k_2k_1})}$$

that indicates the location of $\mathbf{p}_{k_1}$ relatively to $\mathbf{p}_{k_2}$ and $\mathbf{p}_{k_3}$. For each point, we select the more stable triplets on the learning database.

2.2.2. Attraction map calculation

During the test phase, for a configuration of points $C = \{\mathbf{p}_k, k \in [\![1, n_p]\!]\}$, we define an attraction map for each point, which is computed using a voting method and the previously selected triplets. First, we use a k-nearest neighbor algorithm to obtain a similar model s using f as features. With enough variety in the learning database, we can find a model with close head pose, expression and morphology that will let us appropriately constrain our local detector responses. Then, each selected triplet (k_1, k_2, k_3) votes for a location for point $\mathbf{p}_{k_1}$ using points $\mathbf{p}_{k_2}$ and $\mathbf{p}_{k_3}$ of configuration C. Each vote is a gaussian centered at location:

$$\mathbf{p}_{k_1}^{k_2k_3} = \mathbf{p}_{k_2} + \mathbf{v}_{k_2k_3}.f^s(\mathbf{t}_{k_1k_2k_3})$$

The accumulation of these votes for all selected triplets gives an attraction map $\mathbf{A}^k$ for each point. We weight probability maps with these attraction maps to obtain model-weighted probability maps (with $\circ$ the Hadamard product) :

$$\mathbf{W}^k = \mathbf{A}^k \circ \mathbf{J}^k$$

An illustration is given figure 2. Our approach introduces local constraints that let us keep the accurately located points in place, as opposed to a global and stronger constraint, for instance forcing the point configuration to stay within a manifold learned via PCA.

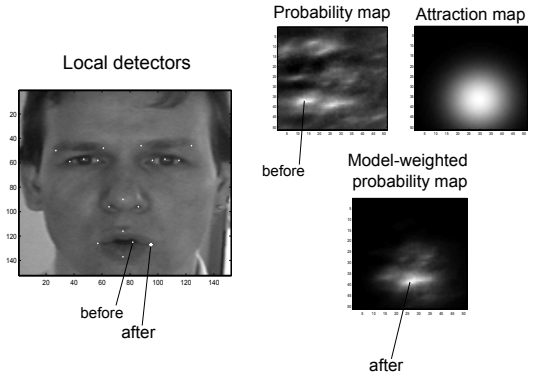

Fig. 2. Outlier correction.

2.2.3. Iterative correction

We present the final algorithm of our landmark detector, which iteratively corrects outliers, and finally selects the likeliest configuration (algorithm 1). We define the likelihood function $L(C)$, based on the shape model, as follows:

$$\frac{1}{L(C)} = \sum_k \sum_{j=1}^{m_k} H(d_{k,j} - \beta)$$

with $d_{k,j}$ the distance between $f^C(\mathbf{t}_j^k)$ and $f^s(\mathbf{t}_j^k)$, m_k the number of selected triplets for point k, H the Heaviside step function and β an acceptance threshold. We use a step function in order to let all admissible locations unpenalized. Thus, the inverse likelihood indicates an estimation of the number of triplets that appear to contain an outlier.

start
initialize a configuration C_0 by calculating probability maps $\mathbf{J}^k$
$C_0 = \{\mathbf{p}_k\}$ with $\mathbf{p}_k = argmax(\mathbf{J}^k)$
for *step u* **do**
  calculate $f^{C_{u-1}}$
  generate similar model via k-NN
  calculate attraction maps $\mathbf{A}^k$ for C_{u-1}
  calculate model-weighted maps using $\mathbf{W}^k = \mathbf{A}^k \circ \mathbf{J}^k$
  estimate the new configuration C_u
  $C_u = \{\mathbf{p}_k\}$ with $\mathbf{p}_k = argmax(\mathbf{W}^k)$
  calculate the likelihood of C_u
  $l(u) = L(C_u)$
end
find the best configuration
$u_{final} = argmax(l)$
$C_{final} = C_{u_{final}}$
stop

Algorithm 1: Binary Map based Point Localization (BiM-PoL)

3. RESULTS

In this section, we first discuss evaluation metrics used to compare landmarking systems before comparing our results with recent state-of-the-art methods.

3.1. Evaluation metrics

Two different kinds of evaluation metrics have been used in recent literature in the domain, leading to relevant questions raised in [2]. One is based on the m_{e17} measure proposed in [1] and represents the cumulative distribution of the image errors (a mean error is calculated for each image and the curve indicates the proportion of images whose mean errors are inferior to a certain threshold). The other represents the cumulative distribution of the landmark errors (indicating the proportion of landmarks whose errors are inferior to a certain threshold). These curves have been mixed up in a lot of recent papers and are definitely not alike (as shown comparing figures 3 and 4). Considering that the predictions follow normal distributions centered on the true landmark locations, then the error obtained for one landmark follows half-normal distribution, which is why the intermediate mean operation and the number of landmarks used for this mean calculation has influence on the shape of image errors cumulative distribution. The starting threshold and the slope increase with the number of landmarks, explaining the differences between figure 3 (image mean errors with 17 landmarks leading to a distinctive "S" shape) and figure 4 (landmark errors).

3.2. Performance evaluation

In this paragraph, we present our results and compare them to other recent state-of-the-art systems (RFRV [2], STASM [1], Cons [3]) with the evaluation metrics used in respective papers. The different parameters of our algorithm (number of selected mean patches, number of triplets used for shape-related constraints inclusion...) have been optimized in cross-validation on the learning database. We learned our system on 500 images of LFPW database (proposed in [3]) that includes interesting variability in illumination, morphology or head pose, and tested it on the well-known BioID database. For testing, we used the 1083 images on which Viola-Jones face detector gave a relevant response. Because of the different point annotations between the learning and the test database, constant biases have been introduced as in [3]. Our results are shown in figure 3 in terms of proportion of images whose m_{e17} errors are inferior to a certain threshold. We obtain results slightly better than the recent regression forest approach proposed by Cootes *et al.* in [2]. Figure 4 shows our results in terms of proportion of landmarks whose errors are inferior to a certain threshold. Our results are equivalent to the recent consensus of exemplars approach proposed by Belhumeur *et al.* in [3].

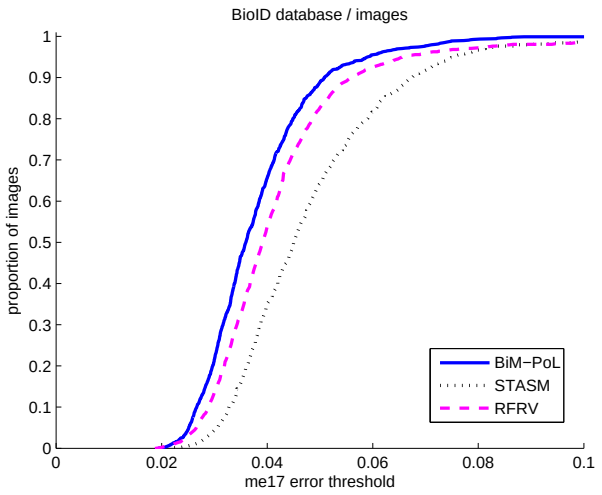

Fig. 3. Cumulative distribution by image errors for BioID database.

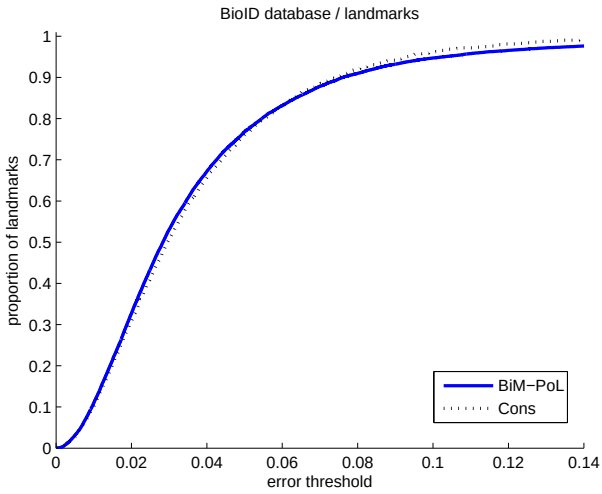

Fig. 4. Cumulative distribution by landmark errors for BioID database.

4. CONCLUSION

We presented in this paper a new method for facial landmark localization in frontal and near-frontal images leading to state-of-the-art results. We proposed a new appearance model for using LBP and LPQ in the context of detection by using binary map cross-correlations to estimate probability maps. In our method, we include shape-related constraints via a voting method using relational properties within triplets of points. This shape model lets us introduce more local constraints than using the widely used global PCA approach and avoid small displacements for accurately located points. This paper also intends to clarify evaluation metrics that have often been mixed up in recent literature in the domain.

5. REFERENCES

[1] S. Milborrow and F. Nicolls, "Locating facial features with an extended active shape model," in *European Conference on Computer Vision (ECCV), 2008*, 2008, pp. 504–513.

[2] T. Cootes, M. Ionita, C. Lindner, and P. Sauer, "Robust and accurate shape model fitting using random forest regression voting," in *European Conference on Computer Vision (ECCV), 2012*, 2012, pp. 278–291.

[3] P.N. Belhumeur, D.W. Jacobs, D.J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, June 2011, pp. 545 –552.

[4] J. M. Saragih, S. Lucey, and J. Cohn, "Face alignment through subspace constrained mean-shifts," in *International Conference on Computer Vision (ICCV), 2009*, Sep. 2009.

[5] V. Rapp, T. Senechal, K. Bailly, and L. Prevost, "Multiple kernel learning svm and statistical validation for facial landmark detection," in *Automatic Face Gesture Recognition and Workshops (FG), 2011 IEEE International Conference on*, Mar. 2011, pp. 265 –271.

[6] B. Martinez, M. Valstar, X. Binefa, and M. Pantic, "Local evidence aggregation for regression based facial point detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2012.

[7] M. Dantone, J. Gall, G. Fanelli, and L. Van Gool, "Real-time facial feature detection using conditional regression forests," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012.

[8] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, June 2012, pp. 2887 –2894.

[9] K. Bailly, M. Milgram, P. Phothisane, and E. Bigorgne, "Learning global cost function for face alignment," in *International Conference on Pattern Recognition (ICPR), 2012*, 2012.

[10] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 915–928, Jun. 2007.

[11] T. Ahonen, E. Rahtu, V. Ojansivu, and J. Heikkil, "Recognition of blurred faces using local phase quantization," in *International Conference on Pattern Recognition (ICPR), 2008*, 2008.