Fast QuadTree-Based Pose Estimation for Security Applications Using Face Biometrics

Paola Barra[1], Carmen Bisogni[1], Michele Nappi[1], and Stefano Ricciardi[2(✉)]

[1] Department of Informatics, University of Salerno, Fisciano, Italy
barra90@gmail.com, {cbisogni,mnappi}@unisa.it
[2] Department of Biosciences, University of Molise, Campobasso, Italy
stefano.ricciardi@unimol.it

Abstract. Face represents a convenient contactless biometric descriptor, currently exploited in a wide range of security applications, though its performance may be considerably affected by subject's pose variations with respect to enrolment pose. This issue is particularly challenging whether the face image is acquired in uncontrolled conditions, or it is extracted from video sequence, the latter representing a more and more frequent case given the huge diffusion of audiovisual content on the internet. To this regard, in this paper, a pose estimation method aimed at rapidly evaluating face rotations is presented. The proposed approach exploits a novel adaptation of quad-tree data structure to achieve an approximate estimate of face's yaw/pitch angles, enabling to select the face image most compliant to the stored template. Preliminary results confirm the efficiency of the proposed method, that provides a more than halved computing time with respect to the state of the art with further improvement margins.

1 Introduction

Nowadays, the use of biometrics as a reliable way to authenticate a person based on the "something you are" paradigm, has spread from typical access control applications to a growing number of transaction authorization procedures, part of which performed through the internet. In this context, face notoriously represents one of the most diffused biometric descriptors, thanks to its good distinctiveness along with high acceptability resulting from its contactless acquisition. Nevertheless, face's reliability may be undermined by ample posing variations, typically induced by acquisition performed under uncontrolled or loosely controlled conditions. The more the angular distance (measured with regard to three degrees of freedom) between the captured face and the reference template, the more the expected impact on the verification accuracy. This performance degradation can be somewhat mitigated by specifically designed feature extraction/matching algorithms, but as the rotation increases the recognition error increases as well. This situation may easily happen whenever a mobile device camera, either in still or video mode, is used for face capture, a kind of person-authentication

M. H. Au et al. (Eds.): NSS 2018, LNCS 11058, pp. 160–173, 2018.
https://doi.org/10.1007/978-3-030-02744-5_12

modality that is becoming more and more common given the large number of apps requiring some form of user authorization. Another typical context To this regard, the growing networks of surveillance cameras diffused throughout buildings and cities provide multiple face capture opportunities from different perspectives which could be used for this purpose. Paradoxically, face capture performed in video mode provide a large number of frames in which subject's face is recorded in slightly different poses (depending on the frame-rate and the subject's head motion with respect to the camera's frustum) some of which could be close to the neutral pose captured during enrollment. Being able to rapidly detect which frame in a sequence is the best match (rotation-wise) to the template image would minimize the posing issue, thus increasing the verification accuracy. To this regard, this paper describes a pose estimation method aimed at rapidly evaluating face rotations, to determine the frame in a video sequence which is the best candidate for subject authentication or identification. The proposed approach exploits a specifically designed adaptation of well-known quad-tree data structure to achieve an approximate estimate of face's yaw/pitch angles, enabling to select the face image most compliant to the stored template. Experiments confirm both the effectiveness and the efficiency of the proposed method, that is able to estimate face rotations in a fraction of the computing time required by the state of the art.

The remainder of the paper is organized as follows. Section 2 resumes main works and methods related to the topic of face pose estimation and normalization. Section 3 describes the proposed system in detail. Section 4 presents the results of the experiments conducted so far. Finally, Sect. 5 concludes, providing directions for future research.

2 Related Works

The problem of head/face pose estimation and the related topic of face "frontalization" (i.e. face normalization according to its rotation axes) have been investigated by a number of works [1], both combined together and dealt with separately. Among the methods treating pose estimation and frontalization as a single problem, some makes use of 3D models, as in [2], in which a dense grid of 3D facial landmarks is projected to each 2D face image, enabling feature extraction in a pose adaptive fashion. In the subsequent step, for the local patch around each landmark, an optimal warp is estimated through homography, to correct texture deformation caused by pose variations.

The authors of [3] focus their attention to the role of occlusions in frontalization, by using Facial Feature Detection to obtain a set of landmarks to be compared to 3D model's landmarks. Other works exploit the distances between key points in face [4], though, these metrics could be affected by considerable angular error which should not be underestimated in pose estimation.

Methods aimed uniquely at pose estimation are more specific and they may be applied in several different contexts. A single face's range-image is sufficient to estimate the 3D pose of a previously unseen subject, in [5]. This approach is based on a novel shape signature to identify noses in range images. The GPU based algorithm generates candidates for their positions, and then generates and evaluates many pose

hypotheses in parallel. A novel error function that compares the input range image to precomputed pose images of an average face model is also proposed.

Face depth data, captured through a depth sensing device, are used in [6] to achieve pose estimation as a regression problem through a random forest framework. Since the regressor needs to be trained on labeled data, the method solves this problem by training only on synthetic data, generating an arbitrary number of training examples without the need of laborious and error-prone annotations.

In [7], a 3D pose-estimate algorithm based on central profile is proposed. The central profile is a unique curve on a 3D face surface that starts from forehead center, goes down through nose ridge, nose tip, mouth center, and ends at a chin tip. Based on the properties of the central profile, Hough transform is applied to determine the symmetry plane by invoking a voting procedure. An objective function maps the central profile to an accumulator cell with the maximal value. It detects the nose tip on the central profile and estimates the pitch angle.

The authors of [8] propose a pose classification framework based on dictionary-learning and sparse representation-based classifier (SRC). They implemented a Gabor feature vector after Gaussian weighted pre-processing as the face pose images' feature and used factors analysis in dictionary training. A specifically built dictionary of face occlusion helps solving the estimation problem when a face is occluded.

In [9], the aim of precisely estimating face rotation angles is achieved by means of a multi-level structured hybrid forest. The head contour is derived from patches, which are either head region or the background. Subsequently, randomly selected patches sub-regions are used to develop the MSHF for head pose estimation. This approach features an average head detection and pose estimation time of about 0.44 s.

Another relevant category of methods focuses on head pose estimation in uncontrolled environments and rely on neural networks.

A convolutional network is used in [10] to map images of faces to points on a low-dimensional manifold parametrized by pose, and images of non-faces to points far away from that manifold. Given an image, detecting a face and estimating its pose is viewed as minimizing an energy function with respect to the face/non-face binary variable and the continuous pose parameters.

Four different convolutional neural networks architectures are described in [11] and compared to evaluate the best pose estimate performance on in-the-wild face datasets. They investigate the use of dropout and adaptive gradient methods and show that the results achieved joining CNNs and adaptive gradient methods lead to the best results.

A method called Hyperface, for simultaneous face detection, landmarks localization, pose estimation and gender recognition, is proposed in [12]. The method works by fusing the intermediate layers of a deep CNN using a separate CNN followed by a multi-task learning algorithm operating on the fused features. This architecture exploits the synergy among the tasks which boosts up their individual performances.

In [13] a multi-task learning deep neural network is applied to a small grayscale face image. The network jointly detects multi-view faces and estimates head pose even under poor environment conditions such as illumination change, vibration, large pose change, and occlusion. The method performs face detection, bounding box refinement, and head pose estimation by using shared features learned through multi-task learning. Other methods focus on the problem of face pose estimation in uncontrolled conditions.

The authors of [14] address this challenge as a continuous regression problem on real images with large variations in background, illumination and expression. To this aim, they propose a probabilistic framework with a general representation not based on locating facial features. Face is represented with a non-overlapping grid of patches, instead. This representation is used in a generative model for automatic estimation of head pose in images taken in uncontrolled environments.

Finally, in [15] a unified model for face detection, pose estimation, and landmark estimation in cluttered images is proposed. This approach is based on a mixture of trees with a shared pool of parts each model by a facial landmark and it uses global mixtures to capture topological changes due to viewpoint.

Compared to the approaches resumed above, the pose estimation method proposed in this paper provides the following contributions:

1. It does not have any of the practical issues involved with 3D models.
2. It does not require neural networks and the related training.
3. It is fast since it is simply based on face landmarks accumulation through the efficient quad-tree data structure [16], disregarding illumination, color or background and returning discrete estimates for both for yaw and pitch angles and it could also estimate the roll (though this third face's degree of freedom is statistically less relevant and would significantly increase computing time).
4. Finally, it can be applied regardless of the context, since it is not dependent on any training set or database.

3 Method Description

The proposed method involves the use of facial landmarks and quad-tree decomposition to estimate face's pose and is structured in three main steps resumed below and depicted in Fig. 1:

- Step 1, *Face detection and facial landmarks localization*;
- Step 2, *Quad-tree decomposition* to obtain a sparse matrix representing the facial features;
- Step 3, *Pose estimation*, transforming the sparse matrix in a tree-array and comparing it to 35 angular references procedurally generated from a 3D synthetic face model through Hamming distance metric.

More in detail, given an input image a face localization algorithm is applied to detect key facial structures on the region of interest. Viola-Jones algorithm [17] is used to detect the presence and the location of the face in the image, resulting in a square region containing the face, referenced through the coordinates of the upper left corner and the size of the sides. After the face detection stage, facial landmarks are used to localize and represent salient regions of the face, such as eyes, nose, eyebrows, mouth and jawline.

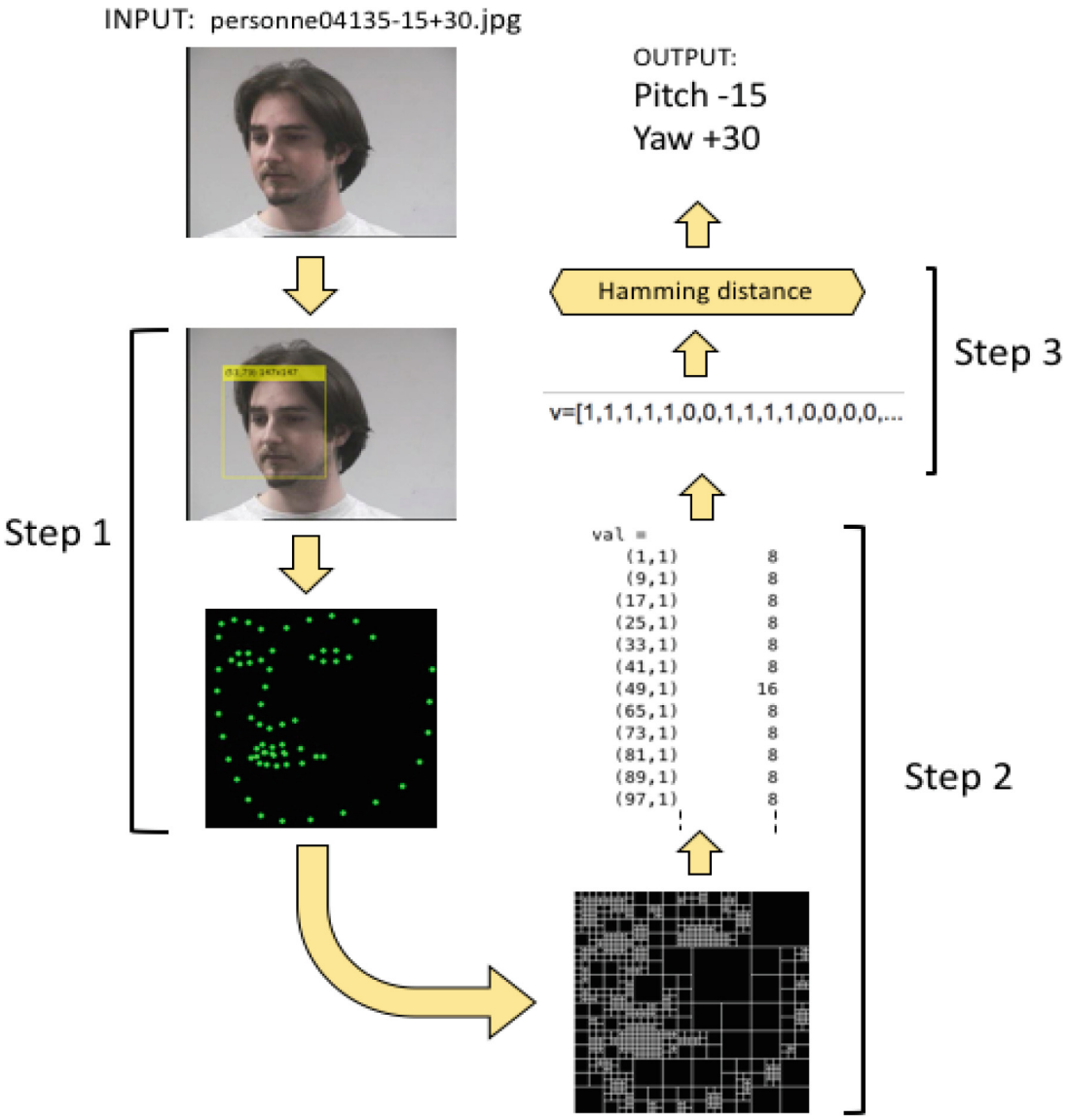

Fig. 1. Steps of the proposed method.

This procedure, based on [18] requires:

1. a training set of labeled facial landmarks on an image, these images are manually annotated, specifying 2D coordinates of regions surrounding each main facial feature.
2. the probability of distance between pairs of input pixels.

Given this training data, an ensemble of regression trees is trained to estimate the facial landmark positions directly from pixel intensity values. The final result of this process is a facial landmark detector that can be used to find facial landmark in a small time with high accuracy. More precisely, this landmarks detector is used to estimate the

location of 68 2D coordinates referencing relevant facial features (shown in Fig. 2) and resulting in a 68 × 2 array organized as follow:

- rows [1–17] contains jawline landmarks coordinates;
- rows [18–22] contains left eyebrow landmarks coordinates;
- rows [23–27] contains right eyebrow landmarks coordinates;
- rows [28–36] contains nose landmarks coordinates;
- rows [37–42] contains left eye landmarks coordinates;
- rows [43–48] contains right eye landmarks coordinates;
- rows [49–68] contains mouth landmarks coordinates.

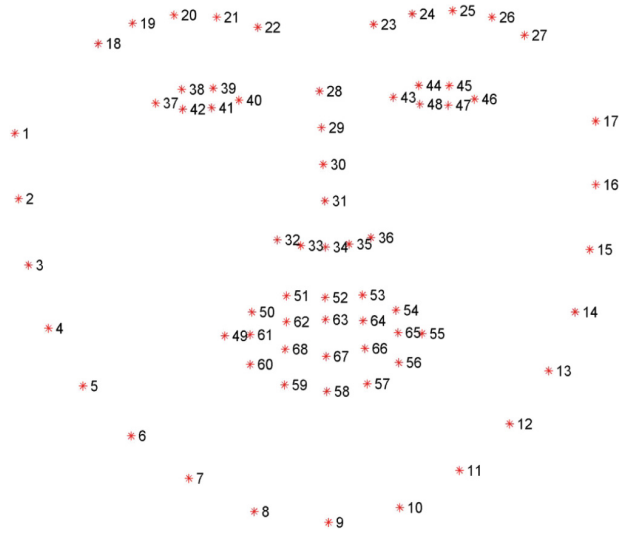

Fig. 2. The indexes of the 68-coordinates corresponding to the facial landmarks selected.

For the sake of balancing efficacy and efficiency, it is crucial to choose the right number of landmarks. A number of landmarks too small, indeed, may lead to insufficient precision in determining the pose angles. On the other hand, using too many landmarks may be counterproductive in terms of computing time, considerably impacting the practical usefulness of the whole approach.

Whenever landmarks have been identified (refer to Fig. 3 for examples), other details as background, color, illumination, hair, makeup and glasses, are no longer relevant for pose estimation and there is no need to consider them in the following steps. This makes possible working on more compact and manageable data further increasing the efficiency of the proposed algorithm. The image is cropped around the outermost landmarks and then converted to black and white to obtain a binary matrix (see Fig. 4). Quad-tree decomposition is then applied to this matrix to obtain the quad-tree. A quad-tree is a very specific tree in which every parent node may have four child nodes or none, and consequently it is a complete tree if every node has four child nodes except leaf nodes. In this case the binary matrix of landmark is the root of the tree and each decomposition makes a child node of the previous node, in other words on each near pass, image is split into four equally-sized smaller sections.

Fig. 3. The output of the facial landmarks algorithm.

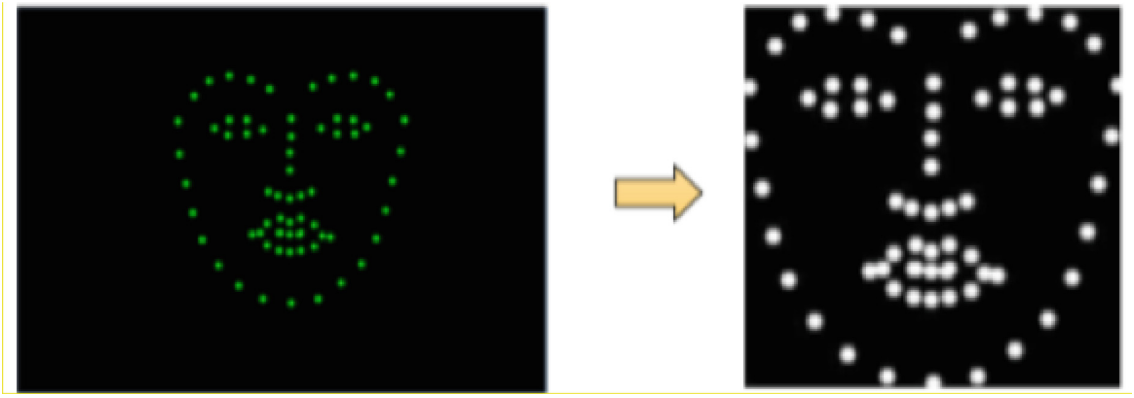

Fig. 4. Face-landmarks image cropping and binarization.

At each step, quad-tree decomposition tests each block to see if it meets some criterion of homogeneity. If a block meets the criterion, it is not divided any further. If it does not meet the criterion, it is subdivided again into four blocks, and the test criterion is applied to those blocks. This process is repeated iteratively until each block meets the criterion or equals the minimum set size. The decomposition process described above represents the region quad-tree, typically used for image processing applications such as image union, intersection and connected component labelling. In this work, the potential advantages of this technique have been brought to the problem of pose estimation by conveniently modifying its parameters and imposing some constraints. In this case, indeed, a block is split if the maximum value of the block elements minus the minimum value of the block elements is greater than the median of the binary landmarks matrix. Moreover, a minimum size has been chosen to avoid generation of blocks smaller than four pixels on each side. This criterion has been preferred to limit decomposition, this threshold can be further increased to make decomposition as quickly as possible, provided that the minimum size must be a power

of 2. By proceeding in this way, the final result of quad-tree decomposition is the original image split into various block of different sizes. The method resizes always the original face-landmarks image in 256 × 256 pixels, consequently, the block size must be between 4 pixels and 256 pixels on each side. In our case there won't be any block bigger than 32 × 32 pixels, because decomposition will be done at least twice.

As it is visible in the Fig. 5, the higher the number of blocks, the closer are the landmarks. The numerical result of this procedure is a sparse matrix which contains in the upper left corner of each block its size, and zeros in the pixels that make up the block. This data structure is particularly suited to create a tree in the form of a binary array, which is computationally inexpensive. The algorithm transforms the sparse matrix resulting from the quad-tree decomposition into an array representing a tree (Fig. 6).

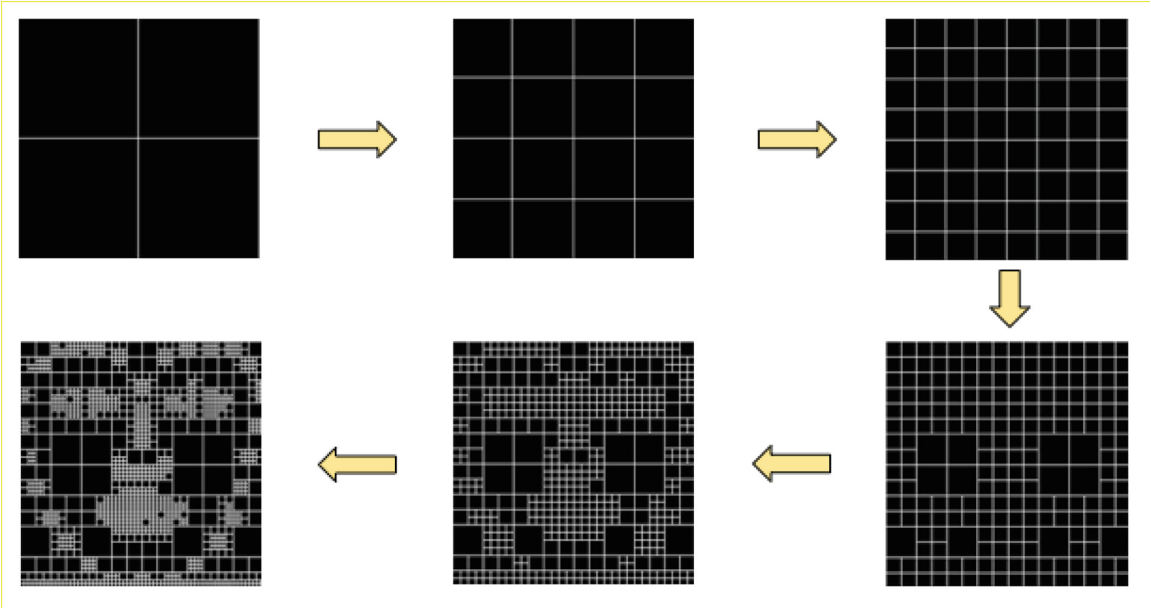

Fig. 5. The consecutive steps in quad-tree decomposition, from the upper left-hand corner to lower left corner: first decomposition, block of size 128 × 128; second decomposition, block of size 64 × 64; third decomposition, block of size 32 × 32; fourth decomposition, block of size 16 × 16; fifth decomposition, block of size 8 × 8; sixth decomposition, block of size 4 × 4.

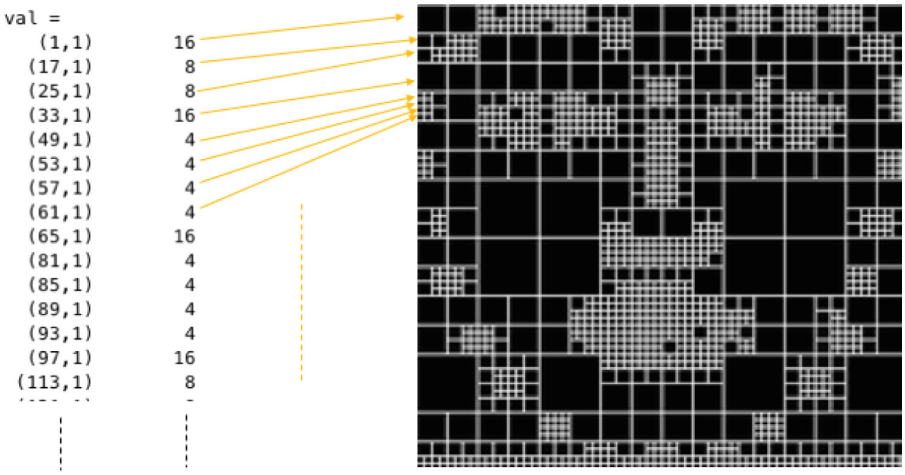

Fig. 6. Sparse matrix with coordinates and size of each block, ordered by column.

This tree is a complete tree of depth 6, in which each node has 4 children, accounting for 1365 nodes in total. Consequently, the array representing the tree has 1365 elements and each element corresponding to a node has value 1 if the node exists, 0 otherwise. Initially, the array is initialized with all values equal to 0. The tree is built with the recursive algorithm summarized in Fig. 7 below:

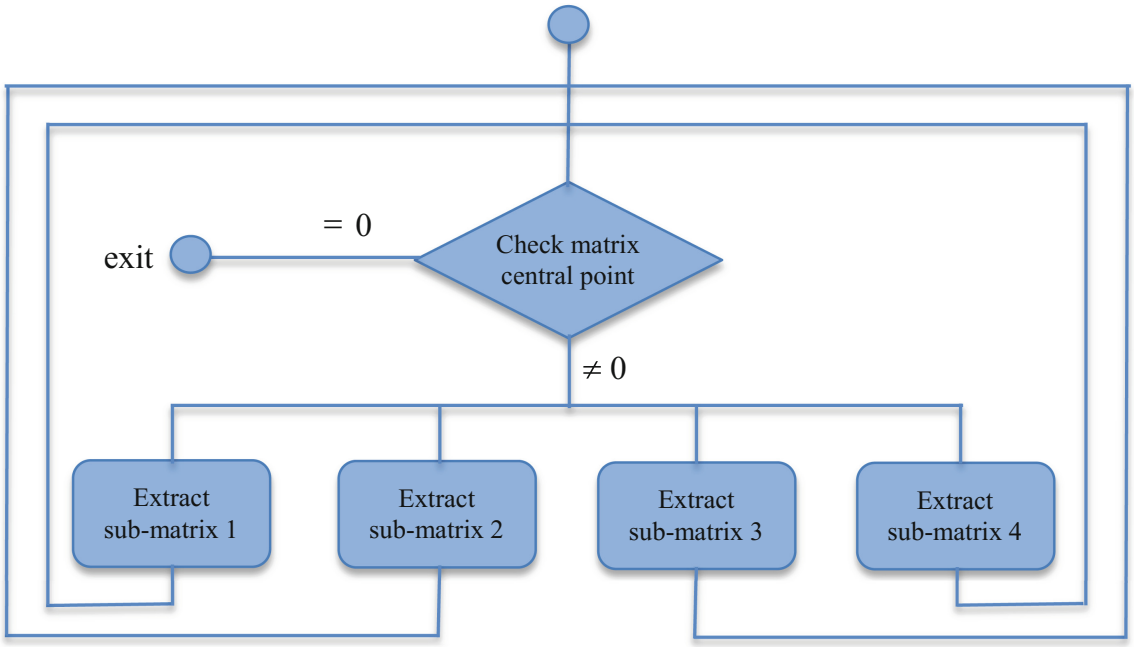

Fig. 7. Given an nxn matrix, if the central point has a value other than 0 then it means that the node has four children and therefore the four matrices originating from that point are recursively processed.

For instance, the first matrix, with 256 × 256 dimension that originates at point (1, 1) represents the root of the tree. To understand if the root has children, a check for a value other than 0 in the central point of the matrix, i.e. the point at (129,129) is performed. If so, then this means that the root has 4 children with 128 × 128 dimension respectively in points (1, 1) (1, 129) (129, 1) (129.129) and the algorithm is executed recursively in each submatrix. For each node that has generated children, a value 1 is assigned to the array representing the tree. The new tree is compared to each

Table 1. Rotation values (pitch, yaw) for all the 35 poses considered. Angular range: pitch [−30°, 30°], yaw [−45°, 45°]

(30, −45)	(30, −30)	(30, −15)	(30, 0)	(30, 15)	(30, 30)	(30, 45)
(15, −45)	(15, −30)	(15, −15)	(15, 0)	(15, 15)	(15, 30)	(15, 45)
(0, −45)	(0, −30)	(0, −15)	(0, 0)	(0, 15)	(0, 30)	(0, 45)
(−15, −45)	(−15, −30)	(−15, −15)	(−15, 0)	(−15, 15)	(−15, 30)	(−15, 45)
(−30, −45)	(−30, −30)	(−30, −15)	(−30, 0)	(−30, 15)	(−30, 30)	(−30, 45)

of the 35 trees corresponding to 35 reference poses featuring the rotation values (pitch, yaw) indicated in Table 1. These 35 reference poses were obtained by renderings of a 3D face model created through MakeHuman [17], an open source application designed for parametric modeling of humanoid figures (see Fig. 8).

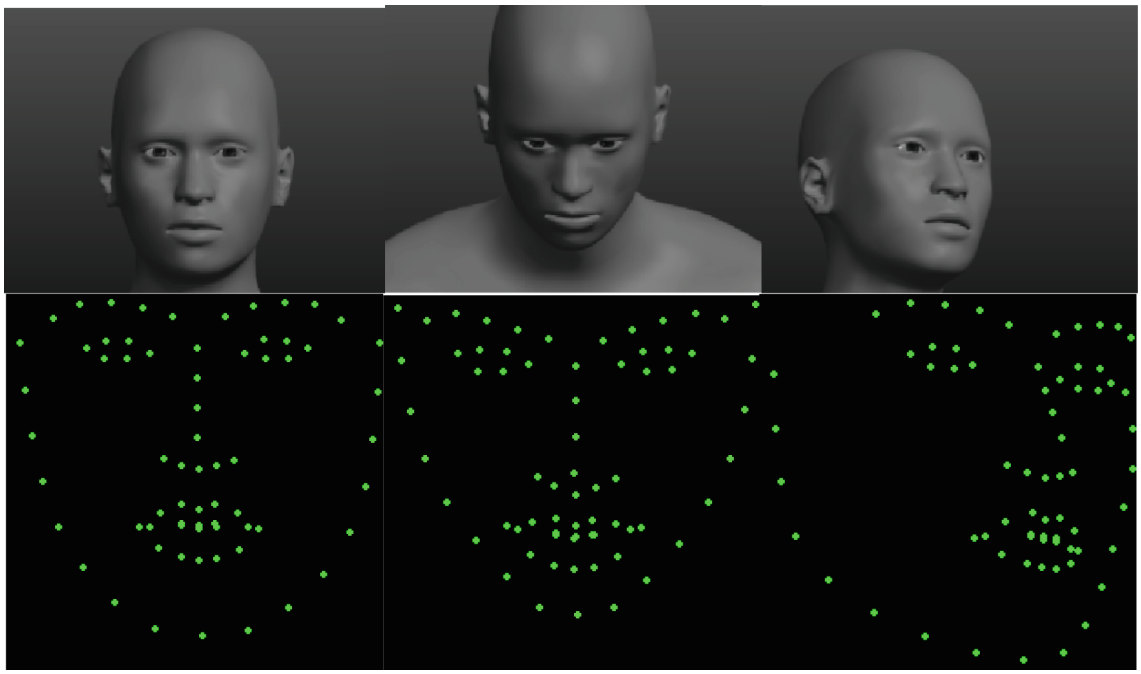

Fig. 8. Examples of synthetic face renderings (top half) and the corresponding landmarks (bottom half) for pitch and yaw values of (0, 0) (−30, 0) (15, −30)

Finally, a comparison is performed through the Hamming distance metric. The input tree is compared with each of the 35 poses. The tree corresponding to the lowest Hamming distance determines the pose of the input tree with regard to a couple of pitch and yaw values.

4 Experiments

The proposed method has the advantage of not having a training phase, therefore it can be applied to any image source. All the experiments conducted have been performed on the public database Pointing '04 [19] consisting of 16 sets of images (samples shown in Fig. 9).

The first set contains 30 frontal images of 15 people, with/without glasses and different ethnicity/skin-color. The remaining sets contain each one a person in different poses. The database contains image with varies head pose, determined by 2 angles (pitch, yaw) varying from −90° to +90°, including the combination of angles. That accounts for 93 images of the same person at different poses, for a total of 2790 image.

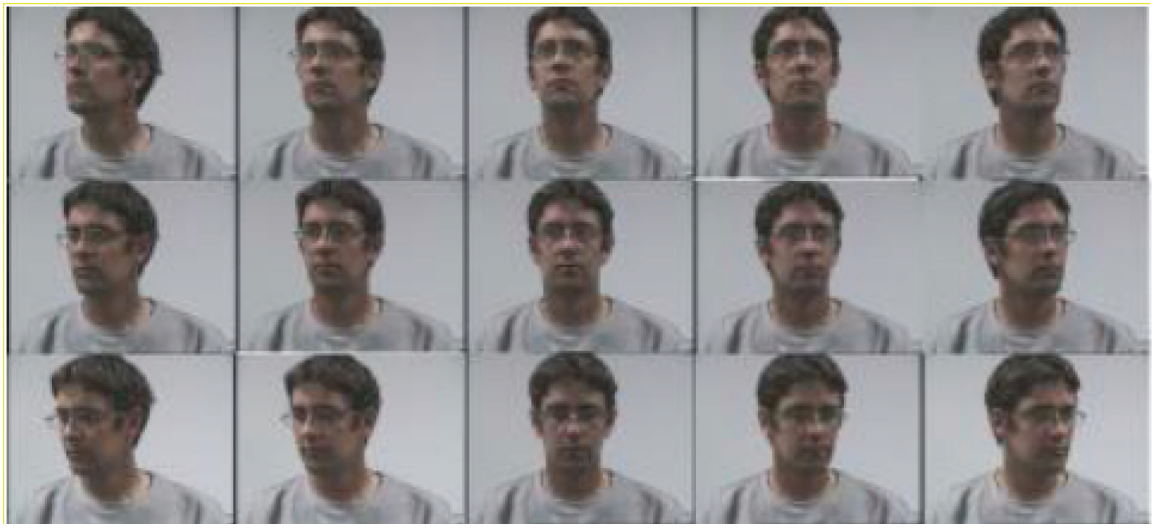

Fig. 9. Example of a subject in different poses from Pointing '04 database.

In this work a subset of the original database has been selected for the experiments according to the following criteria:

- The final aim of the method is to be part of a face recognition pipeline aimed at security applications, therefore, rotation angle exceeding 30° for pitch and 45° for yaw have not been considered. The simple reason is that in case of extreme rotations, face recognition algorithms become unreliable.
- Since the proposed method is based on facial landmarks not affected by variations such as makeup, glasses, hair, etc., images featuring these variations have been discarded.

In the end, the final gallery included 30 frontal images of 15 person, and 15 sets of 35 images each from 15 different subjects, for a total of 555 images. However, 37 images (6% of gallery) with maximum pitch and/or values have been excluded due to some inconsistency with actual rotation values (refer to Fig. 10), resulting in 518 valid images on which the pose estimation method has been applied to.

Fig. 10. subject 162 (+15, +45) compared to synthetic reference with corresponding angular values. It is possible to note an inconsistency between the actual subject's head pose and the annotated rotation values.

For each of the faces in the database pitch and yaw rotations were correctly estimated with a maximum angular error within 15°, that represents the discretization step adopted for the synthetic reference dataset used for comparison. With regard to proposed method's efficiency, the overall time required for pose estimate since input image is provided amounts to approximately 0.22 s on average. More in detail, landmarks detection required the greatest fraction of total processing time (74% accounting for 0.16 s), quad-tree decomposition only required 8% and 0.017 s, while pose estimation was performed in 0.04 s on average (18%).

In other terms, most of the time is used for the landmarks detection step. This could mean that an improvement of this step can make the algorithm substantially faster. For example, the number of landmark required to ensure an accurate (application-wise) estimate of the face pose, could possibly be reduced. Nevertheless, the proposed method results twice as fast as the state of the art [6], though its angular accuracy appears to be lower (Fig. 11).

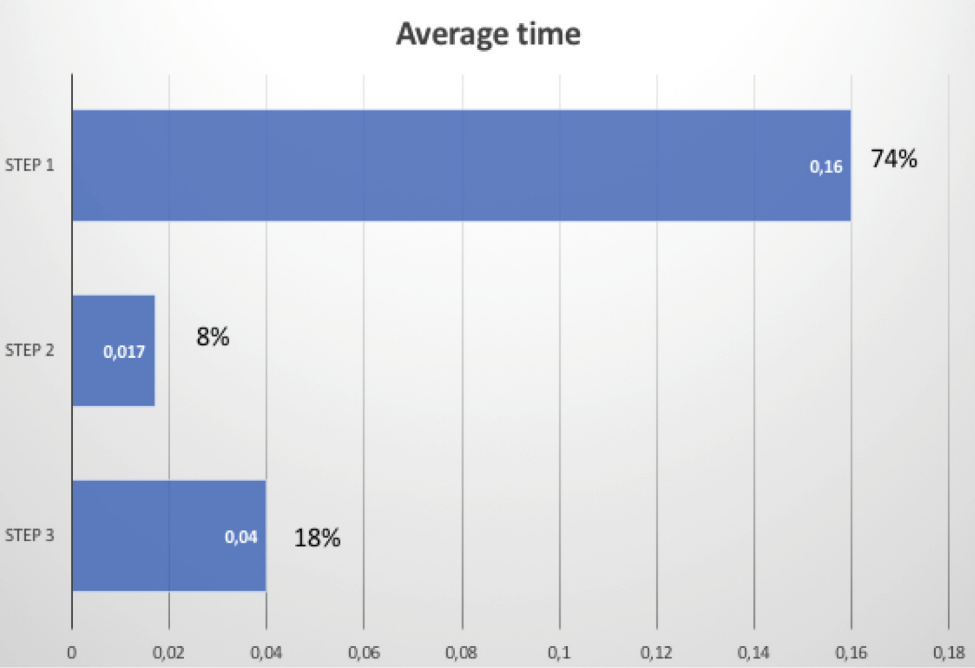

Fig. 11. Comparison of time required by main steps of the proposed method.

However, for the main application context to which this method is targeted, this lower accuracy could be negligible, since the main purpose is to determine which frame results the most promising for face recognition in the shortest possible amount of time. This is particularly true for the frequent case of video sequences which are becoming even more diffused than still images for security applications. As shown in Fig. 12, we successfully tested the proposed algorithm on video content in order to rapidly determine the best candidate (i.e. the most frontal frame) for face recognition.

Fig. 12. An example of application on a short video of an interview (from YouTube). The yellow box highlights the from the most frontal frame in the sequence, according to the proposed method. (Color figure online)

5 Conclusions

Face pose may relevantly affect the accuracy and therefore the reliability of face biometrics in the context of security applications. We presented a novel method to estimate face pose through a simple yet computationally inexpensive approach based on quadtree representation of facial landmarks. The experiments conducted so far on a dataset featuring 500+ images, confirm the effectiveness and the efficiency of this solution, that is capable of an approximate estimate of face's yaw and pitch angles in roughly 0,22 s, with a speed increment of about 200% over state of the art algorithms (without training). We plan to further improve both the accuracy and the speed of this approach by reducing both the discrete angular step considered in this work and the number of facial landmarks.

References

1. Murphy-Chutorian, E., Trivedi, M.M.: Head pose estimation in computer vision: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **31**(4), 607–626 (2009)
2. Ding, C., Tao, D.: Pose-invariant face recognition with homography-based normalization. Pattern Recogn. **66**, 144–152 (2017)
3. Çelık, A., Arica, N.: Occlusion analysis for face frontalization. In: 2016 4th International Symposium on Digital Forensic and Security (ISDFS)
4. Kavitha, J., Mirnalinee, T.T.: Automatic frontal face reconstruction approach for pose invariant face recognition. Procedia Comput. Sci. **87**, 300–305 (2016)

5. Breitenstein, M.D., Kuettel, D., Weise, T., Van Gool, L., Pfister, H.: Real-time face pose estimation from single range images. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8. IEEE, June 2008
6. Fanelli, G., Gall, J., Van Gool, L.:. Real time head pose estimation with random regression forests. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 617–624. IEEE, June 2011
7. Lia, D., Pedrycz, W.: A central profile-based 3D face pose estimation. Pattern Recogn. **47**(2), 525–534 (2014)
8. Liao, H., Shejie, L., Wang, D.: Tied factor analysis for unconstrained face pose classification. Optik Int. J. Light Electron Opt. **127**(23), 11553–11566 (2016)
9. Liu, Y., Xie, Z., Yuan, X., Chen, J., Song, W.: Multi-level structured hybrid forest for joint head detection and pose estimation. Neurocomputing **266**, 206–215 (2017)
10. Osadchy, M., Cun, Y.L., Miller, M.L.: Synergistic face detection and pose estimation with energy-based models. J. Mach. Learn. Res. **8**(May), 1197–1215 (2007)
11. Patacchiola, M., Cangelosi, A.: Head pose estimation in the wild using Convolutional Neural Networks and adaptive gradient methods. Pattern Recognit. **71**, 132–143 (2017)
12. Ranjan, R., Patel, V.M., Chellappa, R.: Hyperface: a deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. IEEE Trans. Pattern Anal. Mach. Intell. (2017)
13. Ahn, B., Choi, D.-G., Park, J., Kweon, I.S.: Real-time head pose estimation using multi-task deep neural network. In: Robotics and Autonomous Systems, vol. 103, pp. 1–12, May 2018
14. Aghajanian, J., Prince, S.: Face pose estimation in uncontrolled environments. In: BMVC, vol. 1, no. 2, p. 3, September 2009
15. Zhu, X., Ramanan, D.: Face detection, pose estimation, and landmark localization in the wild. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2879–2886. IEEE, June 2012
16. Samet, H.: The quadtree and related hierarchical data structures. ACM Comput. Surv. (CSUR) **16**(2), 187–260 (1984)
17. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, vol. 1, pp. I-I. IEEE (2001)
18. Kaemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
19. http://www.makehuman.org
20. Gourier, N., Hall, D., Crowley, J.L.: Estimating face orientation from robust detection of salient facial features. In: Proceedings of Pointing 2004, ICPR, International Workshop on Visual Observation of Deictic Gestures, Cambridge, UK