Benchmarking 3D pose estimation for face recognition

Pengfei Dou, Yuhang Wu, Shishir K. Shah, and Ioannis A. Kakadiaris
Computational Biomedicine Lab, University of Houston, TX, USA
{pengfei,yuhang}@cbl.uh.edu, {sshah,IKakadia}@central.uh.edu

***Abstract*—3D-Model-Aided 2D face recognition (MaFR) has attracted a lot of attention in recent years. By registering a 3D model, facial textures of the gallery and the probe can be lifted and aligned in a common space, thus alleviating the challenge of pose variations. One obstacle preventing accurate registration is the 3D-2D pose estimation, which is easily affected by landmarks. In this work, we present the performance that state-of-the-art pose estimation algorithms could reach using state-of-the-art automatic landmark localization methods. We generated an application-specific dataset with more than 59,000 synthetic face images and groundtruth camera pose and landmarks, covering 45 poses and six illumination conditions. Our experiments compared four recently proposed pose estimation algorithms using 2D landmarks detected by two automatic methods. Our results highlight one near-real-time landmark detection method and a highly accurate pose estimation algorithm, which would potentially boost the 3D-Model-Aided 2D face recognition performance.**

I. Introduction

Most of the existing face verification methods use solely 2D information. To address pose variation, methods based on deformable 2D shape models [1] have been proposed and extended [2]–[4]. However, as pose variations are intrinsically caused by rigid face motion, using 3D data could be beneficial. In addition, using 3D data can also help handle more general and continuous view changes, illumination variance, and unexpected occlusions, examples of which may not be included in the gallery [5,6].

Even though multiple studies have shown that using 3D data in both enrollment and authentication phases would offer state-of-the-art face recognition performance [7,8], it is much more practical to use 2D data. In addition, most of the online data and images are still 2D. One solution proposed to solve this problem is using a 3D Morphable Model (3DMM) [9], built using a large number of 3D facial scans. Although 3DMM can be used to estimate the pose and illumination through shape and texture decomposition, it requires detailed and accurate point-wise correspondence between the 2D image and 3D model as well as (time consuming) parameter tuning. Another promising solution is fitting a general 3D model (reference model) to the 2D face image [5,9] or the 3D face model (if available) [10] of each subject in the gallery to obtain both the fitted 3D face model and the aligned 2D face image during enrollment. In the authentication phase, the fitted 3D model is rotated and translated to fit the 2D probe image and lift a reprojected probe image so that corresponding features between enrollment and authentication are compared in the same normalized 2D space [10]. These approaches fall under the umbrella of 3D-Model-Aided 2D Facial Recognition (MaFR).

Fig. 1. The 45 poses we used to evaluate the pose estimation algorithm.

One of the critical obstacles that prevents applying MaFR under an unconstrained environment is the fragility and sensitivity of the 3D-2D pose estimation algorithm to landmark localization error. When a portion of the landmarks are annotated/detected on the wrong pixel, or completely missing in some cases, the projection matrix of pose estimation will change dramatically. As the projection of 2D probe face is determined solely by this projection matrix, the features extracted from the reprojected 2D image are extremely sensitive to pose estimation perturbations. However, among previous work related to MaFR [5,10], almost all of them circumvent the problem by assuming that the landmarks are correct or annotated manually. With perfect landmarks, almost all pose estimation algorithms could easily reach 100% accuracy. In the real world, however, we can hardly expect landmarks to be perfect, even with manual annotations. The question arises: What is the expected face recognition performance using state-of-the-art landmark detection and pose estimation methods? Reviewing the recent publications on the 3D-2D pose estimation (or Perspective-n-Point, PnP) problem, we observed that, although comparisons were performed in almost every work between the newly proposed algorithm and its predecessors, the datasets they used all fell into the same category: a 3D model and synthesized 2D landmarks perturbed by Gaussian noise (usually less than 10 pixels) [11]–[14]. In this work, we aim to highlight the importance of pose estimation in MaFR and provide a systematical evaluation of state-of-the-art pose estimation algorithms. Our contributions are:

i) We used an application-specific and a very challenging dataset to benchmark four recently proposed 3D-2D pose estimation algorithms.

ii) Instead of synthesized 2D landmarks, we applied two state-of-the-art landmark detection methods to localize the 2D reference points, and used a post-processing step to refine these automatic landmarks.

II. 3D-2D Pose Estimation Algorithms

In this paper, we selected four recently published 3D-2D pose estimation algorithms and benchmarked their performance on our synthetic face dataset. As a baseline, we formulated the 3D-2D pose estimation as a least-square minimization problem and solved it using the Levenberg-Marquardt algorithm (LM-LSM) [15].

A. LHM: An iterative solution to PnP

LHM is an iterative pose estimation algorithm proposed by Lu *et al.* [11]. LHM explicitly incorporates a weak-perspective assumption and solves the pose estimation problem by minimizing an object-space collinearity error, thus guaranteeing global convergence.

Assume that the image point $v_i = (u_i, v_i)^T$ is the projection of a 3D point p_i on the normalized image plane. Under a weak perspective model, v_i, p_i, and the projection center are collinear, thus satisfying the collinearity equation $Rp_i + t = V_i(Rp_i + t)$, where $V_i = \frac{v_i v_i^t}{v_i^t v_i}$ is the line-of-sight projection matrix which projects the 3D point orthogonally to the line of sight determined by the image point. Based on this collinearity equation, the object-space collinearity error could be defined as $e_i = (I - \hat{V}_i)(Rp_i + t)$, where $\hat{V}_i$ is the observed line-of-sight projection matrix. Given a set of 3D-2D correspondences, LHM seeks to minimize the sum of squared object-space collinearity error: $E(R,t) = \sum_{i=1}^{n} \|e_i\|^2 = \sum_{i=1}^{n} \|(I - \hat{V}_i)(Rp_i + t)\|$.

By adopting an Expectation-Maximization (EM) framework, LHM evolves successively by first improving an estimation of the rotation and then estimating the associated translation. Like every iterative method, LHM also requires an initialization to start. In our benchmarking experiments, we initialized with a coarse pose estimated via Direct Linear Transformation (DLT) [16].

B. EPnP: A non-iterative solution to PnP

EPnP was proposed by Lepetit *et al.* [12] as a non-iterative solution to the 3D-2D pose estimation problem. Compared with earlier methods, EPnP achieved a linear computational complexity with respect to the number of reference points. This computation efficiency was achieved by representing each of N points as the weighted sum of four (non-planar case) or three (planar case) virtual control points, based on which 3D-2D pose could then be estimated by solving a quadratic equation system with constant size.

By expressing each 3D reference point p_i as the weighted sum of four virtual control points $c_j, (j = 1, 2, 3, 4)$, the 3D-2D pose estimation problem could be formulated as solving the following linear system for virtual control point coordinates in a Camera Coordinate System (CCS):

$$\forall i, \omega_i \begin{bmatrix} u_i \\ 1 \end{bmatrix} = K \cdot p_i^c = K \cdot \sum_{j=1}^{4} \alpha_{ij} c_j^c \ , \tag{1}$$

where w_i are scalar perspective parameters which could be eliminated by further transforming the linear equations, u_i is the projection of the i^{th} 3D reference point on the image plane, K is the camera intrinsic parameter matrix, and α_{ij} are coefficients w.r.t. each virtual control point, which could be easily computed in the World Coordinate System (WCS) where the 3D model is defined.

C. RPnP: A robust solution to PnP

RPnP is another non-iterative algorithm proposed by Li *et al.* [13]. It is based on their pioneering work [17], in which the classical P3P problem was reformulated to reduce the number of unknown parameters by perspective similar triangle (PST), thus achieving better stability. By dividing n reference points into $n-2$ triplets, $n-2$ fourth-order polynomials could be derived. Instead of solving this equation system directly, RPnP further transformed it into an eighth-order cost function F by summing up the square of each polynomial. With root-finding techniques, the minima of this cost function could be solved. As F is an eighth-order polynomial, multiple minima could be found. Among them, RPnP chooses the one with the least reprojection residual (similarly to EPnP).

D. OPnP: An optimal non-interative solution to PnP

In OPnP [14], rotation is parameterized as a non-unit quaternion (2), where $s = a^2 + b^2 + c^2 + d^2$ is the scale of the quaternion, thus transforming the pose estimation problem into an unconstrained minimization problem (4).

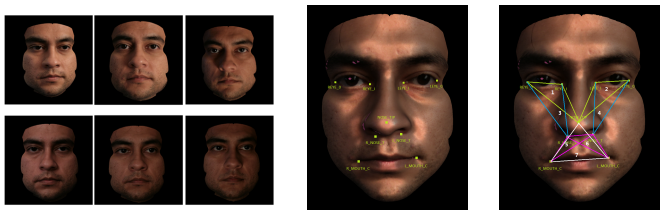
$$R = \frac{1}{s} \cdot \begin{bmatrix} r_1^T \\ r_2^T \\ r_3^T \end{bmatrix} \ , \tag{2}$$

$$\begin{aligned} r_1^T &= \begin{bmatrix} a^2+b^2-c^2-d^2 & 2bc-2ad & 2bd+2ac \end{bmatrix} \\ r_2^T &= \begin{bmatrix} 2bc+2ad & a^2-b^2+c^2-d^2 & 2cd-2ab \end{bmatrix} \\ r_3^T &= \begin{bmatrix} 2bd-2ac & 2cd+2ab & a^2-b^2-c^2+d^2 \end{bmatrix} \end{aligned} \ , \tag{3}$$

By replacing s with the reciprocal of the average depth of 3D reference points $\frac{1}{\lambda}$, the perspective projection between 3D reference points $q_i, (i = 1, 2, \cdots, n)$ and their corresponding 2D points on image plane $[u_i, v_i]^T$ could then be reformulated into a linear system with $2n$ equations:

$$(1 + r_3^T \hat{q}_i) \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} r_1^T \\ r_2^T \end{bmatrix} q_i + \begin{bmatrix} \hat{t}_1 \\ \hat{t}_2 \end{bmatrix}, i = 1, 2, \cdots, n \ , \tag{4}$$

where $\begin{bmatrix} \hat{t}_1 & \hat{t}_2 & \hat{t}_3 \end{bmatrix}^T = \frac{1}{\lambda} t$ and $\hat{q}_i$ is the 3D reference coordinates after centralization to center of the 3D model. By summing up the square of each equation, 3D-2D pose estimation was transformed into an unconstrained minimization problem (5), which was then solved using the Grobner Basis (GB) Solver:



(a) (b) (c)

Fig. 2. (a) Six illuminations of the synthetic dataset, (b) definition of the nine landmarks, and (c) the seven triplets used to average the automatic landmarks.

$$min_{a,b,c,d} f(a,b,c,d) = \|M\alpha\|_2^2 = \alpha^T M^T M \alpha \ , \quad (5)$$

where $\alpha = [1, a^2, ab, ac, ad, b^2, bc, bd, c^2, cd, d^2]^T$, and M could be constructed using p_i and q_i. Please refer to [14] for more details.

III. Experimental Setup and Evaluation Protocol

This section describes the dataset and protocol used to benchmark the selected 3D-2D pose estimation algorithms. The dataset consists of a large collection of synthesized faces with arbitrary poses uniformly distributed in pan $[-45°, +45°]$ and tilt $[-25°, +25°]$ space. To detect 2D landmarks, we apply two recently published automatic landmark detection algorithms. Based on the automatic 2D landmarks and the manual 3D landmarks on the model, 3D-2D facial pose is estimated for comparison with groundtruth, which is computed during the process of creating synthetic data.

A. Dataset

We decided to use synthetic 2D face data generated using 3D faces for the following reasons:

- Instead of the rigid cube model used for evaluation in recent 3D-2D pose estimation research, we use the face model which is application-specific.
- Instead of using Gaussian noise corrupted 2D landmarks, we apply facial landmark detectors and retrieve automatic 2D landmarks, as is the case in practical face recognition systems.
- When using synthetic images, groundtruth pose information in terms of rotation, translation, and camera parameters is manually assigned and thus can be controlled more precisely for analysis.

Although there exist several face pose datasets (*e.g.*, [18]), they only cover discrete pose space. Our synthetic face dataset, however, by sampling randomly within each predefined pose, covers the whole pose space. Our synthetic face dataset consists of 11 subjects from the UHDB11 dataset [19], and covers 45 poses and six different illumination conditions (Fig. 1 and Fig. 2(a)). Within each pose, we generated 20 synthetic faces for each subject and each illumination condition. For each synthetic face image, we computed the groundtruth pose, the camera intrinsic/extrinsic parameters, and the groundtruth 2D landmarks.

B. 2D Landmark detection

To retrieve 2D reference landmarks, we applied two recently proposed open-source state-of-the-art landmark detection algorithms [4,20].

Zhu *et al.* [4] proposed a landmark detection algorithm based on multi-view trees with a shared pool of part models. The main idea of the algorithm is to represent the gradient of a local region of landmarks in a part model and use a mixture of trees to capture global topological changes due to viewpoint. By using a pool of shared part models and rearranging the mixture of part models into a tree structure, the model fitting was performed efficiently with dynamic programming. We use the term "UCI landmarks" in the following sections to represent the landmarks detected with this algorithm.

Xiong *et al.* [20] proposed a supervised descent method (SDM). In training, SDM learns a sequence of descent directions that gradually update the shape parameters to minimize the difference between appearance at the candidate location and ground-truth. In testing, SDM retrieves the descent direction to update the shape model based on the learned Hessian and Jacobian matrix of Newton method in iterations. As indicated in [20], this method achieved real-time performance. We use the term "CMU landmarks" in the following sections to represent the landmarks detected with this algorithm.

C. Benchmarking protocol

In our experiment, we used nine reference points located at the right/left eye outer and inner corner, nose tip, right/left nostrils, and right/left mouth corner, as illustrated in Fig. 2(b). These landmarks are selected because (i) they are stable during facial movement, (*e.g.*, facial expressions) and (ii) they have distinct appearance features, which is desirable for automatic landmark detection.

We manually annotated the 3D landmarks on the 3D models and applied the aforementioned two algorithms to retrieve the 2D landmarks on the synthetic facial images. Anticipating that automatic landmarks would contain large deviations from the groundtruth, we applied a post-processing step to refine the automatic 2D landmarks by averaging them in seven triplets (Fig. 2(c)), which resulted in seven refined landmarks. Comparison with groundtruth indicated that this post-processing step reduced the error of the landmark detection (Fig. 4). For more details, please refer to the next section. Based on these two sets of 2D landmarks, we applied each pose estimation algorithm to our synthetic dataset. We first compared their accuracy across 45 facial poses in terms of two metrics:

- Rotation error: Given the true camera rotation R and the estimated $\hat{R}$, the rotation error measures the maximum error between Euler angles corresponding to these two rotations.
- Translation error: Given the true camera translation t, translation error measures the relative error of estimated translation $\hat{t}$ as $E_t = \frac{\|t-\hat{t}\|_2}{\|t\|_2}$.

To statistically evaluate the pose estimation algorithms' sensitivity to noise, we added Gaussian noise to the true 2D

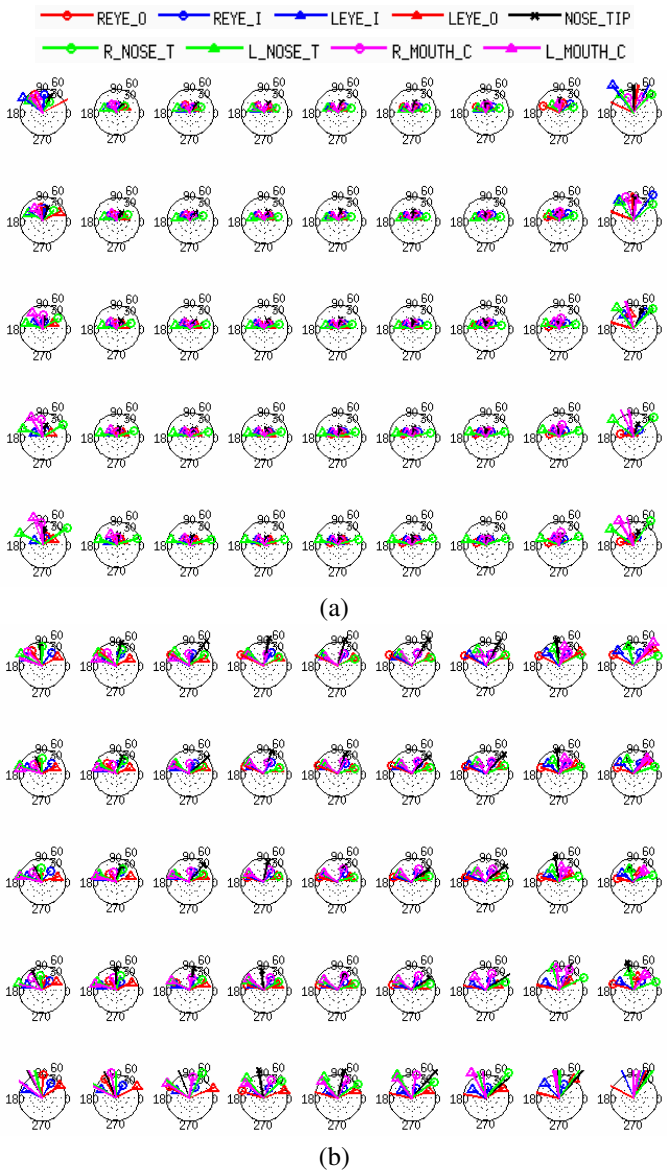


Fig. 3. Errors of CMU and UCI automatic landmarks in 45 poses. Bar length indicates the error magnitude and bar orientation indicates the error orientation relative to the groundtruth (red square/triangle: right/left eye outer-corner; blue square/triangle: right/left eye inner-corner; black line: nose tip; green square/triangle: right/left nostril; pink square/triangle: right/left mouth corner). (a) CMU automatic landmarks and (b) UCI automatic landmarks.

landmarks and tested, for each algorithm, the probability that it could guarantee a predefined level of accuracy under a certain level of noise. For each synthetic face in a subset of our dataset, we ran pose estimation 300 times with upper-bounded Gaussian noise and observed how many times the rotation and translation error fell below our predefined thresholds.

IV. Results

This section presents the experimental results using the aforementioned 3D-2D pose estimation algorithms. In Sec. IV.B and Sec. IV.C we report and analyze the pose estimation accuracy for different automatic landmarks. In Sec. IV.D we present the sensitivity analysis results. Before that, however, we first analyze the accuracy of the detected landmarks with respect to the groundtruth.

A. Error analysis of the landmark detection

First, we present the error of the automatic landmarks detected. In the polar coordinate system (Fig. 3), each line corresponds to a 2D facial landmark. The horizontal orientation is set to be the same as the 2D vector from the Right Eye Inner Corner to the Left Eye Inner Corner ($\vec{v_0} = (REYE\_I - LEYE\_I)$). In each subgraph, the length of the line indicates the average distance between the automatic landmark (A_i) and groundtruth (G_i), and the orientation shows the intersection angle between two vectors $\vec{v_i} = A_i - G_i$ and $\vec{v_0}$.

Within the pose space covering pan $[-35°, +35°]$ and tilt $[-25°, +25°]$, CMU landmarks exhibit an average error less than 30 pixels. Within challenging poses covering pan $[-45°, -35°]$ and $[+35°, +45°]$, however, many false detections were observed. However, UCI landmark detection exhibits a much higher rate of successful detection for challenging poses, but lower accuracy, especially on detecting nose tip. We also observed that all of the landmark errors have a global offset towards the angle of $90°$. One of the possible reasons is that the semantic positions of true landmarks are slightly different from the positions showing the most distinct appearance features. However, since all of the following results are evaluated with the same landmarks, we could neglect the influence of this global error.

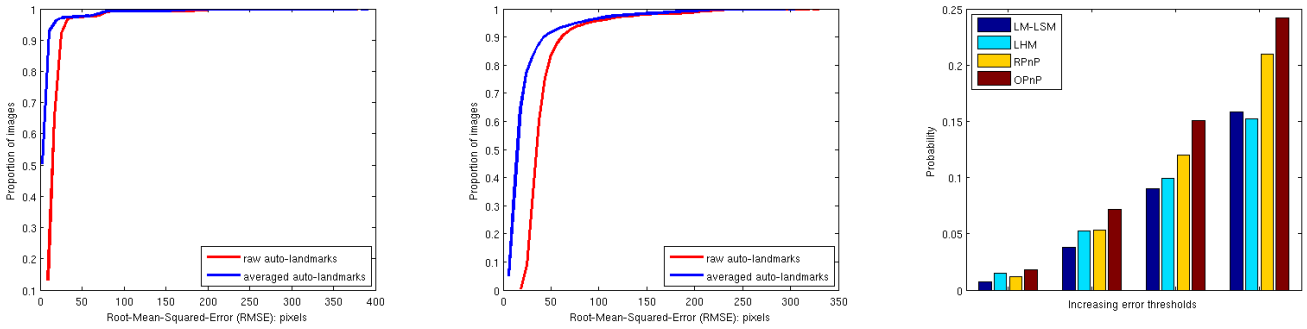
To reduce detection error, we used the mean of seven triplets of landmarks, as illustrated in Fig. 2(c). Error statistics w.r.t. true landmarks in Fig. 4 indicate that the seven landmarks obtained after post-processing are more accurate.

B. Results using the CMU landmarks

With CMU landmarks, we applied the pose estimation algorithms to our synthetic dataset. The accuracy results (Fig. 5) illustrate that OPnP performed better w.r.t. rotation error (less than $5°$) and translation error (less than *5%*) than the other algorithms in most of the 45 poses, except for those covering pan $[-15°, +15°]$ and tilt $[+5°, +25°]$ where OPnP closely followed LHM. For several poses that cover pan $[+35°, +45°]$, all these algorithms performed poorly. This is mainly because the automatic landmarks here (Fig. 3) have larger errors. We also observed that, in almost all of these poses, EPnP performed very poorly. This was expected, as, according to the results in [13,14], EPnP's pose estimation accuracy degenerates quickly with increasing noise added to the synthetic 2D landmarks.

C. Results using the UCI landmarks

The pose estimation results with UCI landmarks are illustrated in Fig. 6. As shown in the Fig. 3, UCI landmarks had larger errors compared with CMU landmarks. As a result, pose estimation accuracy exhibited a small degradation. In most of the poses, OPnP still outperformed the other algorithms. While within poses covering tilt $[-5°, +15°]$ and pan $[-25°, +25°]$, LHM exhibited better accuracy. As EPnP performed poorly in these two experiments, we excluded it from the next experiment.



(a) (b) (c)

Fig. 4. (a) Post-processing reduces the error of CMU landmarks (red line: raw auto-landmarks; blue line: refined auto-landmarks), (b) post-processing reduces the error of UCI landmarks (red line: raw auto-landmarks; blue line: refined auto-landmarks), and (c) sensitivity analysis results: four groups of bars correspond to increasing error thresholds so that the rotation error is smaller than $4°$, $8°$, $12°$, and $16°$ and the translation error is smaller than 5%, 10%, 15%, 20%.

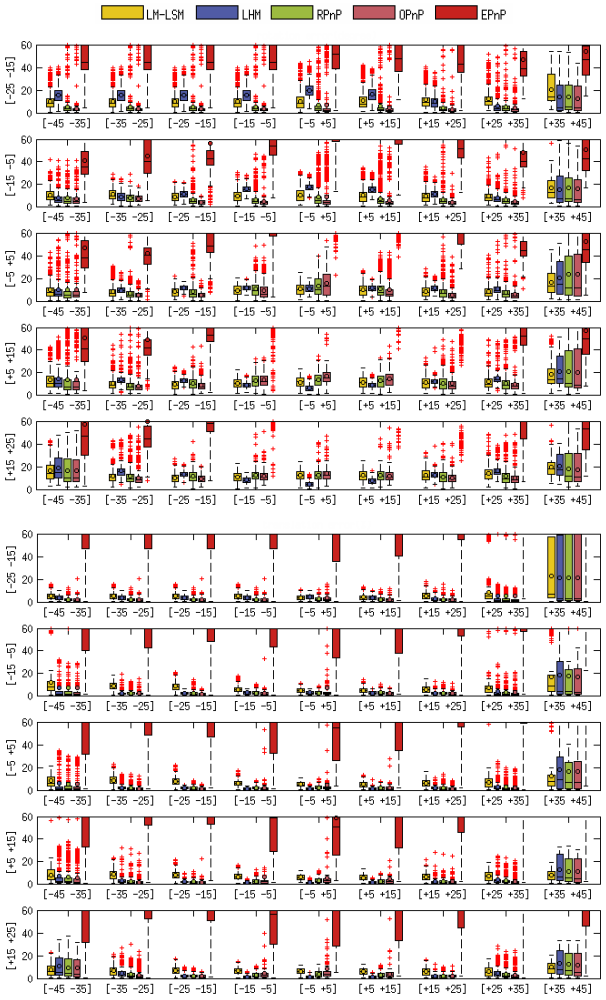


Fig. 5. Boxplot of the pose estimation accuracy with CMU landmarks (circles indicate the mean of error). (T) Rotation error with CMU landmarks and (B) translation error with CMU landmarks.

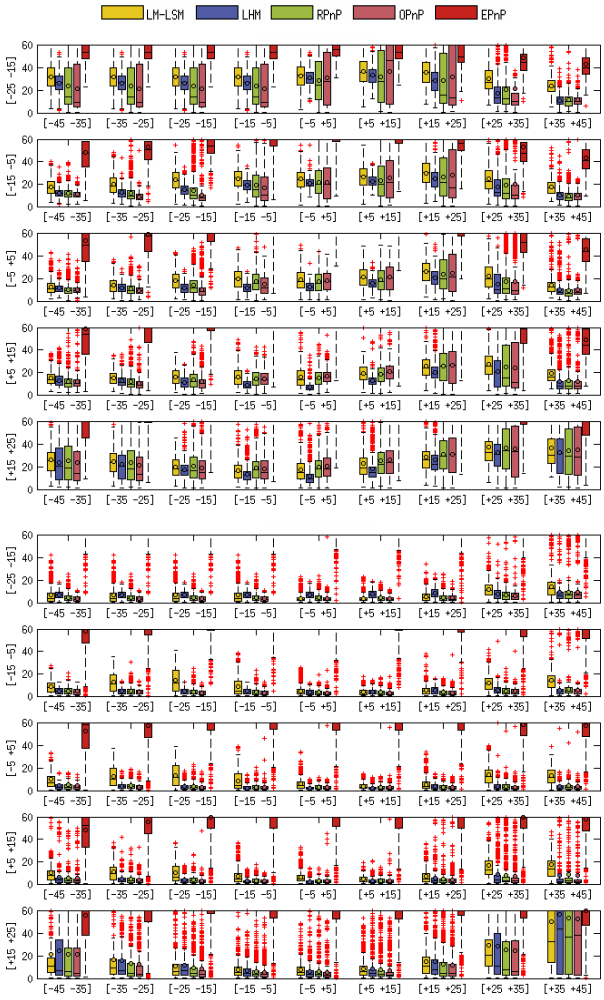


Fig. 6. Boxplot of the pose estimation accuracy with UCI landmarks (circles indicate the mean of error). (T) Rotation error with UCI landmarks and (B) translation error with UCI landmarks.

D. Sensitivity analysis for varying noise levels

We also evaluated the sensitivity of each algorithm with respect to varying noise levels. With true 2D landmarks available, we could simulate different noise levels by adding well-controlled Gassian noise. In our experiment, we added Gaussian noise to the true 2D landmarks and assigned multiple tolerance bounds to measure if an algorithm could guarantee a predefined level of accuracy under the noise. We selected from

our dataset a subset consisting of one image for each subject, illumination, and pose. For each synthetic face in the subset, we ran the pose estimation algorithms with Gaussian noise corrupted landmarks 300 times and computed the probability that rotation and translation errors fell below four predefined tolerance bounds. We set the mean of Gaussian noise to be zero and variance to be 10, which is larger than that used by Li *et al.* [13,14]. The thresholds of rotation error were set to 4°, 8°, 12°, and 16° respectively and the thresholds for translation error were 5%, 10%, 15%, and 20%. We observed that there was no evident correlation between the pose estimation accuracy and the facial pose when synthetic landmarks were used. This was reasonable as none of our 45 predefined facial poses fell into the quasi-singular case [13], which would usually cause severe degeneration to pose estimation accuracy. As a result, we could summarize all the pose estimation results covering 45 poses and compute the possibility distribution (Fig. 4(c)). Our experiments indicated that OPnP is the most robust algorithm as, under the same noise level, OPnP exhibited much higher probability to guarantee better accuracy. The second robust algorithm is RPnP, which exhibited slightly lower probability than OPnP in guaranteeing high (less than 16° in rotation error and less than 20% in translation error) accuracy.

V. Conclusion

In this paper, we systematically evaluated four recently proposed pose estimation algorithms for estimating 3D facial pose. Using our synthetic dataset, we reported the current state-of-the-art performance for 3D facial pose estimation using landmarks localized automatically.

Based on CMU landmarks, OPnP exhibited the best accuracy with an average of less than 5° in rotation error and less than 5% in translation error for the majority of the 45 poses covered in our dataset. For the remaining poses covering tilt $[+5^\circ, +25^\circ]$ and pan $[-15^\circ, +15^\circ]$ LHM outperformed OPnP. This implies that we should not expect a sole algorithm to cover all facial poses. For example, OPnP might guarantee high pose estimation accuracy with mugshot face images. For surveillance face images, however, LHM might be more promising.

With respect to the degeneration observed in the UCI landmarks when compared with the CMU landmarks, pose estimation accuracy exhibited only linear degradation. This implies the robustness and stability of LHM, RPnP, and OPnP. EPnP, however, exhibited very poor performance on both CMU and UCI landmarks.

VI. Acknowledgment

This research was funded in part by the US Army Research Lab (W911NF-13-1-0127) and the UH Hugh Roy and Lillie Cranz Cullen Endowment Fund. All statements of fact, opinion or conclusions contained herein are those of the authors and should not be construed as representing the official views or policies of the sponsors.

References

[1] T. Cootes and C. Taylor, "Active shape models: Smart snakes," in *Proc. British Machine Vision Conference*, Leeds, UK, Sep. 22-24 1992, pp. 266–275.

[2] S. Milborrow, "Locating facial features with active shape models," Master's thesis, University of Cape Town, 2007.

[3] M. Rogers and J. Graham, "Robust active shape model search," in *Proc. European Conference on Computer Vision*, London, UK, May 27-31 2002, pp. 517–530.

[4] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, June 16–21 2012, pp. 2879–2886.

[5] G. Hsu and H. Peng, "Face recognition across poses using a single 3D reference model," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Columbus, Ohio, June 23-28 2013, pp. 869–874.

[6] D. Yi, Z. Lei, and S. Z. Li, "Towards pose robust face recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, Ohio, June 23-28 2013, pp. 3539–3545.

[7] I. Kakadiaris, G. Passalis, G. Toderici, M. Murtuza, Y. Lu, N. Karampatziakis, and T. Theoharis, "Three-dimensional face recognition in the presence of facial expressions: An annotated deformable model approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 640–649, 2007.

[8] X. Lu and A. K. Jain, "Deformation modeling for robust 3D face matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 8, pp. 1346–1357, 2008.

[9] I. Kemelmacher-Shlizerman and R. Basri, "3D face reconstruction from a single image using a single reference face shape," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, pp. 394–405, 2011.

[10] G. Toderici, G. Passalis, S. Zafeiriou, G. Tzimiropoulos, M. Petrou, T. Theoharis, and I. Kakadiaris, "Bidirectional relighting for 3D-aided 2D face recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, June 13-18 2010, pp. 2721–2728.

[11] C.-P. Lu, G. D. Hager, and E. Mjolsness, "Fast and globally convergent pose estimation from video images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 6, pp. 610–622, 2000.

[12] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem," *International Journal of Computer Vision*, vol. 81, no. 2, pp. 155–166, 2009.

[13] S. Li, C. Xu, and M. Xie, "A robust O(n) solution to the perspective-n-point problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1444–1450, 2012.

[14] Y. Zheng, Y. Kuang, S. Sugimoto, K. Astrom, and M. Okutomi, "Revisiting the PnP problem: A fast, general and optimal solution," in *Proc. IEEE International Conference on Computer Vision*, Sydney, Australia, Dec. 1-8 2013, pp. 2344–2351.

[15] K. Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quarterly Journal of Applied Mathmatics*, vol. II, no. 2, pp. 164–168, 1944.

[16] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2000.

[17] S. Li and C. Xu, "A stable direct solution of perspective-three-point problem," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 25, no. 05, pp. 627–642, 2011.

[18] N. Gourier, D. Hall, and J. L. Crowley, "Estimating face orientation from robust detection of salient facial structures," in *Proc. International Workshop on Visual Observation of Deictic Gestures*, Cambridge, England, UK, Aug. 23-26 2004, pp. 1–9.

[19] G. Toderici, G. Evangelopoulos, T. Fang, T. Theoharis, and I. Kakadiaris, "UHDB11 database for 3D-2D face recognition," in *Proc. Pacific-Rim Symposium on Image and Video Technology*, Guanajuato, Mexico, Oct. 28-Nov. 1 2013, pp. 73–86.

[20] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Portland, Oregon, June 25-27 2013, pp. 532–539.