Multiview Facial Landmark Localization in RGB-D Images via Hierarchical Regression With Binary Patterns

Zhanpeng Zhang, *Student Member, IEEE*, Wei Zhang, *Member, IEEE*, Jianzhuang Liu, *Senior Member, IEEE*, and Xiaoou Tang, *Fellow, IEEE*

***Abstract*— In this paper, we propose a real-time system of multiview facial landmark localization in RGB-D images. The facial landmark localization problem is formulated into a regression framework, which estimates both the head pose and the landmark positions. In this framework, we propose a coarse-to-fine approach to handle the high-dimensional regression output. At first, 3-D face position and rotation are estimated from the depth observation via a random regression forest. Afterward, the 3-D pose is refined by fusing the estimation from the RGB observation. Finally, the landmarks are located from the RGB observation with gradient boosted decision trees in a pose conditional model. The benefits of the proposed localization framework are twofold: the pose estimation and landmark localization are solved with hierarchical regression, which is different from previous approaches where the pose and landmark locations are iteratively optimized, which relies heavily on the initial pose estimation; due to the different characters of the RGB and depth cues, they are used for landmark localization at different stages and incorporated in a robust manner. In the experiments, we show that the proposed approach outperforms state-of-the-art algorithms on facial landmark localization with RGB-D input.**

***Index Terms*— Facial landmark localization, gradient boosting decision tree, random binary pattern, random forest (RF).**

I. Introduction

FACIAL landmark localization is a fundamental problem in many computer vision applications, including 3-D face modeling, face tracking, cartoon generation, and face recognition. It has been widely studied for the past two decades [1]–[6]. Many recent works have shown impressive localization results on near-frontal face images [7], [8]. However, the large variations of face appearance caused by illumination, expression, and out-of-plane rotation make the robust and accurate localization in real-world applications still a challenging task.

Given a parametric face model S, the goal of facial landmark localization with an input image I is to estimate the model parameters S_I, which minimizes the difference between the estimation and the ground truth $\widehat{S_I}$ (i.e., $S_I = \arg\min_S ||\widehat{S_I} - S||_2$). Existing image-based methods can be generally divided into two categories: 1) detection-based and 2) regression-based approaches. Detection-based approaches [3], [9]–[12] usually extract local appearance for each facial landmark. The localization procedure can be conducted with a searching window over the image. To handle the ambiguous or occluded cases, these approaches usually need the landmark distribution constraints as the shape prior to refine the result. However, the exhaustive searching process is time consuming and hard to reach interactive performance.

Recently, many image-based approaches tend to locate the landmarks by updating the initialization guess iteratively with the regression technique. The active appearance model (AAM) [13], [14], as a classical face representation method, encodes the face shape and global appearance together into a statistical model in the training phase. For a new input face image, the difference between the face appearance and the synthesized model is used to drive a parameter update procedure. References [4] and [15] propose to find the best estimation with subsequential nonlinear regression steps to update the AAM parameters. Meanwhile, local features, such as scale-invariant feature transform [16] and binary features [17]–[21], are popular in the regression-based approaches [22], [23] and achieve impressive results. In [1] and [24], the algorithms cast the votes for facial landmark location with random forest (RF) regression on local patches. Similarly, deep convolutional networks are used to detect major facial points by multilevel regression [25]. These approaches present impressive results on near-frontal faces. However, facial landmark localization in the multiview condition is still challenging, as the face texture shows substantial changes in large head rotation and

Manuscript received August 14, 2013; revised November 15, 2013 and December 31, 2013; accepted February 17, 2014. Date of publication February 26, 2014; date of current version August 31, 2014. This work was supported in part by the Natural Science Foundation of China under Grant 61201443; in part by the Science, Industry, Trade, Information Technology Commission of Shenzhen Municipality, China, under Grant JC201005270378A; and in part by the Guangdong Innovative Research Team Program under Grant 201001D0104648280. This paper was recommended by Associate Editor L. Zhang. *(Corresponding author: Wei Zhang.)*

Z. Zhang and X. Tang are with the Shenzhen Key Lab of CVPR, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518000, China, and also with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: zz013@ie.cuhk.edu.hk; xtang@ie.cuhk.edu.hk).

W. Zhang is with the Media Laboratory, Huawei Technologies Company Ltd., Shenzhen 518129, China (e-mail: zhangwei@siat.ac.cn).

J. Liu is with the Media Laboratory, Huawei Technologies Company Ltd., Shenzhen 518129, China, and also with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: liu.jianzhuang@huawei.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2014.2308639

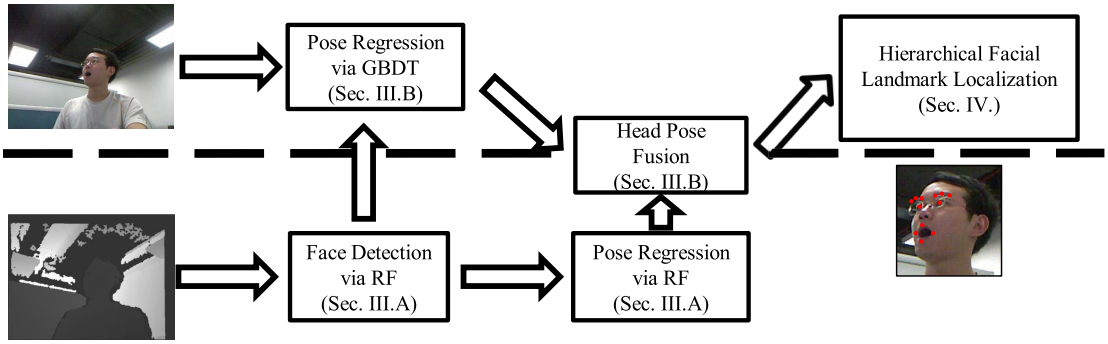


Fig. 1. Framework of our multiview facial landmark localization in RGB-D images.

the performance of these algorithms may drop. Furthermore, it is a nontrivial problem to set the region of the local landmarks for the regression task. A narrow region leads to results that are sensitive to initial landmark configuration, while a wide region increases appearance modeling difficulty.

Fortunately, depth sensors provide valuable 3-D spatial cues to resolve the ambiguity in the 2-D image data, which is insensitive to the illumination change. The wide influential work on body localization [26] has demonstrated its effectiveness. In [27] and [28], high-resolution depth data are used for nonrigid face tracking and 3-D pose estimation. Meanwhile, as the RGB and depth cues can give complementary descriptions of the scene, the combination of the two cues has also been proven to be a promising approach for many vision tasks. The commercial depth and RGB cameras, such as Microsoft Kinect, are used in [29] for facial cartoon animation. Cai *et al.* [30] develop a deformable model fitting algorithm for face tracking in depth and RGB sequences. In [31], depth input is used together with an RGB image for 3-D face alignment. However, this approach combines depth and RGB cues in RF straight for face alignment, which needs high-resolution depth input from 3-D scanners. In addition, the RF-based alignment algorithm may cause the overfitting problem, which limits its extensibility, especially in cases where the training and testing data are heterogeneous.

In this paper, we propose a multiview facial landmark localization system with RGB-D inputs. The system can handle noisy depth input, such as the input from a Microsoft Kinect camera, and achieve real-time performance. The facial landmark location distribution is treated as a head pose conditional model and our system estimates both the 3-D head pose and 2-D landmark location. In the algorithm, we extract random local binary patterns of different scales, and estimate the facial parameters with hierarchical regression techniques. Fig. 1 shows the flow diagram of the proposed system. The depth channel and RGB channel are used at different stages: the depth input is fed to the RF for face detection and pose estimation at the beginning stage; the RGB input is fed to gradient boosted decision trees (GBDTs) for head pose and hierarchical facial landmark location regression when the face is available. The pose estimation results from the depth and RGB inputs are weighted and combined to improve the precision.

The contributions of this paper include: we propose a multiview facial landmark localization method with a new framework for the combination of the RGB and depth cues, which improves the accuracy of head pose and facial landmark localization. In addition, we take a hierarchical regression approach to locate the facial landmarks. The hierarchical configuration models the facial appearance variation in different levels, which lowers the learning difficulty for the regression framework and achieves a high generalization ability.

The remainder of this paper is organized as follows. The related work and building block algorithms are presented in Section II. In Section III, we first show the pose estimation in depth and RGB inputs individually, followed by the pose fusion from these two observations. After that, the multiview facial landmark localization in the RGB channel is formulated into a regression framework and the detailed GBDT-based regression algorithm is given in Section IV. The system implementation and experimental results on different databases are shown in Section V. Finally, Section VI concludes this paper.

II. Related Work and Preliminaries

In this section, we give a brief review of the randomized binary features and pattern regression techniques used in the following sections.

A. Comparison-Based Randomized Binary Pattern

The feature descriptors based on image intensity comparison are widely used for vision tasks, such as visual correspondence and object matching. BRIEF [17], ORB [19], BRISK [20], and FREAK [21] are some examples. The core idea of these features is that a binary string derived from simply comparing pairs of image intensities can efficiently describe a keypoint. For the purpose of scale and rotation invariances, different random sampling patterns are used in ORB and BRISK. The key advantage of these binary descriptors is that the usage of the Hamming distance can efficiently replace the Euclidean distance, which makes these features suitable for applications with low memory and real-time requirement. In the remaining part of this paper, we take a similar binary descriptor with the following definition:

$$\mathbf{u}^k(I) = [\mathbf{u_0}(I), \ldots, \mathbf{u_{k-1}}(I)] \tag{1}$$

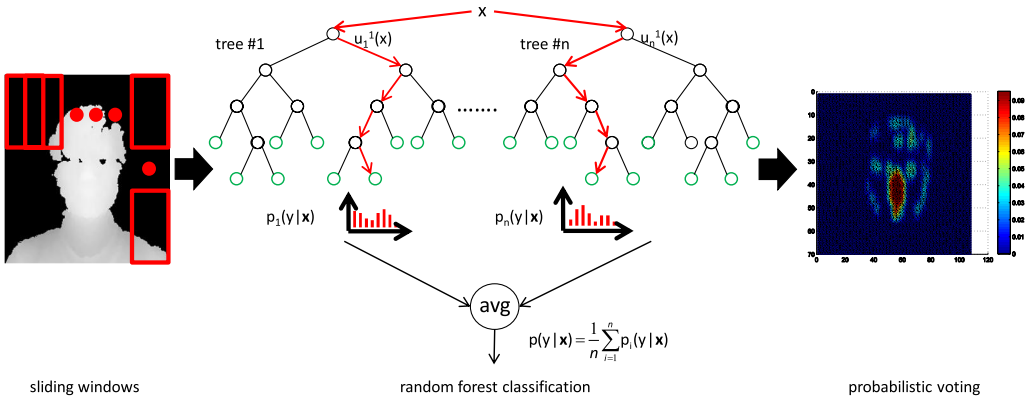


Fig. 2. Object detection via Hough forest.

where

$$\mathbf{u_i}(I) = \begin{cases} 1, & \mathbf{g_i}\left(I, p_i^0, p_i^1\right) > \tau_i \\ 0, & \text{else} \end{cases}. \tag{2}$$

In (2), $\mathbf{u_i}(I)$ is the binary function defined on the input pattern I and a pairwise independent sampling function $\mathbf{g_i}(I, p_i^0, p_i^1)$ with sampling pair $\{p_i^0, p_i^1\}$ and threshold τ_i. Note that the combination of $\{p_i^0, p_i^1, \tau_i\}$ defines a binary feature.

The main difference of the proposed binary descriptor $\mathbf{u}$ with previous mentioned binary descriptors is in the definition of the input pattern sampling region Ω. Since we use $\mathbf{u}$ for the tasks of facial landmark regression, the correspondent pattern sampling region is determined by the parameterized regression target. At the coarse level of the facial parameter regression, the pattern sampling region covers the whole head region. When the regression goes further to the subtle facial landmark location, the sampling pattern is restricted to be in the neighbor of the facial components. Ω keeps a reasonable scale with the correspondent regression target, such that we can do hierarchical sampling and localization in a coarse-to-fine manner.

B. RF for Object Detection and Pose Regression

RF [32], [33] is a simple but powerful learning tool widely used for the classification as well as regression tasks. A typical RF consists of a set of decision trees. Each nonleaf node of a decision tree contains a binary test that partitions the input data. At the training stage, the input data are recursively partitioned via hierarchically building the binary test for each nonleaf node. Each tree is grown until a predefined stopping criterion (e.g., the maximum tree depth or the number of minimum samples falling to a leaf node) is reached. Each leaf contains the information about the training samples reaching it, e.g., the class distribution in the case of classification or the predicted outcome in the case of regression. At the test stage, a sample is passed down to the leaf node and takes the predicted output value or class label with maximum posterior probability.

An ensemble of these trees is assembled and trained in a randomized way to achieve better generalization and stability, compared with a single tree. The randomization is obtained by the following rules: training each tree with a random subset of training samples or constructing binary tests and randomly selecting a subset at each nonleaf node. In our experiments, these two rules are used together for RF construction. We use the 1-D random binary pattern $\mathbf{u}^1$ to construct the random binary test set.

Hough forest [34] is a special type of RF used for object detection. Similar to the generic RF training process, a set of labeled samples $P = \{\mathbf{u}^k(x_i), c_i, d_i\}$ are fed to build the random decision trees recursively, where $\mathbf{u}^k(x_i)$ is the random binary pattern of input x_i, and $\{c_i, d_i\}$ is the label pair, which indicates the existence of the object and the offset to the object center. Two types of measurement are used to evaluate the quality of node splitting during the training. Classification uncertainty is used to measure the impurity of the object and nonobject splitting. Regression uncertainty corresponds to the uncertainty of object specified sample offset. The two types of measurements are used together for best splitting determination in the training process. The test stage is shown in Fig. 2. Given an input image, a set of samples is generated via sliding the detection window. The nonobject samples are filtered out via going through the RF. The remaining object candidates are left and used for predicting the object location via Hough transform.

C. GBDTs for Pose Regression and Landmark Localization

Gradient boosting [35] is another widely used regression/classification technique for extensive vision tasks, such as object localization [4], [7] and pose estimation [36]. It achieves the state-of-the-art result [23] in the task of face alignment. The gradient boosting method consists of an ensemble of weak prediction models, which are additively combined to output the final prediction. GBDT is a specific version of gradient boosting where the weak prediction model is a decision tree.

Fig. 3 shows our GBDT scheme. A GBDT that takes random binary patterns for weak prediction model learning is used for the regression task. Here, we use depth-k decision trees in the GBDT. Each tree can be represented as a prediction function $h(\mathbf{u}^k(x))$. $\mathbf{u}^k(x)$ is a k-dimensional random binary pattern extracted from the input sample x. $h(\mathbf{u}^k(x))$

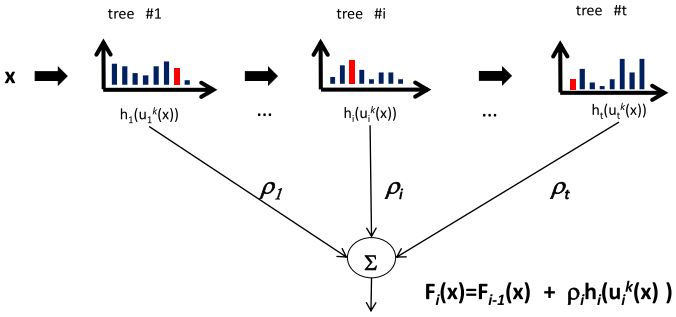


Fig. 3. GBDTs for regression.

partitions the sample space to 2^k disjoint sets. Given the $L2$ loss objective function, a decision tree represents a piecewise prediction function, which corresponds to the 2^k output of leaf nodes.

The boosting method fits the facial landmark localization problem very well since it provides an efficient way to select the random binary pattern features. Specifically, in the GBDT training process, the goal is to find a regression function $F(x)$ that maps an input feature vector x to a target value y, while minimizing the expected value of the $L2$ loss function $\Psi(y, F(x))$. $F(x)$ is the sum of T stages of weak regression functions

$$F(x) = \sum_{t=1}^{T} \rho_t h_t\left(\mathbf{u}_t^k(x)\right) \tag{3}$$

where ρ_t is the learning rate for each stage t and is given a fixed value in our experiments, and $h_t(\mathbf{u}_t^k(x))$ is the regression function of a k-dimensional random binary pattern $\mathbf{u}_t^k(x)$. For simplicity, the outputs correspondent to all the 2^k enumerative values of $\mathbf{u}_t^k(x)$ are not identified explicitly in the equation. The GBDT learning is a greedy stage-wise approach. At each training stage t, we select a weak regressor $h^*(\mathbf{u}_t^{k*}(x))$ from a large random binary feature pool $\{\mathbf{u_t}^k(x)\}$ that maximally decreases the total loss for N training samples

$$h^*\left(\mathbf{u}_t^{k*}(x)\right) = \underset{\{h(\mathbf{u_t}^k(x))\}}{\arg\min} \sum_{i=1}^{N} \Psi(y_i, F_{t-1}(x_i) + h\left(\mathbf{u}^k(x_i)\right). \tag{4}$$

A steepest descent step is then applied to the minimization problem of (4). It is infeasible to apply gradient descent on $h(\mathbf{u}^k(x_i))$ since the weak regressor represents a piecewise constant function. So, at each stage t, we compute the pseudoresiduals [35] by

$$\tilde{y}_i = -\left[\frac{\partial \Psi\left(y_i - F(x_i)\right)}{\partial F(x_i)}\right]_{F(x_i)=F_{t-1}(x_i).} \tag{5}$$

In our implementation, we use the least squares for the loss function $\Psi(y, F(x))$ and then $\tilde{y}_i = y_i - F_{t-1}(x_i)$. The problem thus becomes

$$h^*\left(\mathbf{u}_t^{k*}(x)\right) = \underset{\{h(\mathbf{u_t}^k(x))\}}{\arg\min} \sum_{i=1}^{N} \left\|\tilde{y}_i - h_t\left(\mathbf{u_t}^k(x)\right)\right\|^2. \tag{6}$$

Given a random binary pattern $\mathbf{u}^k(x)$, $h(\mathbf{u}^k(x))$ can be naturally solved in (6), since the its output is the mean of $\tilde{y}_i$ of the samples that fall into the corresponding leaf node.

Algorithm 1 Training Process for Gradient Boosting Random Binary Pattern Regression

Input: $\{x_i\}_1^N$, $\{y_i\}_1^N$.
Output: $F(x) = \sum_{t=1}^{T} \rho_t h_t(\mathbf{u}_t^k(x))$.
$F_0(x)$ =mean$\{y_i\}_1^N$.
Randomly generate a pool of binary patterns $\{\mathbf{u^k}\}_1^K$.
for t = 1 to T **do**
  for i = 1 to N **do**
    $\tilde{y}_i = y_i - F_{t-1}(x_i)$.
  end for
  $\mathbf{u}_t^{k*}(x) = \arg\min_{\mathbf{u_t}^k(x)} \sum_{i=1}^{N} \|\tilde{y}_i - h(\mathbf{u_t}^k(x_i))\|^2$.
  $F_t(x) = F_{t-1}(x) + \rho_t h^*(\mathbf{u}_t^{k*}(x))$.
end for

We simply choose the suitable $\{\mathbf{u_t}^k(x)\}$ in the training process. The pseudocode of our GBDT regression is described in Algorithm 1.

III. Face Detection and Pose Estimation in RGB-D Images

At the beginning stage of the proposed facial landmark localization framework, the face is detected first in the depth input, followed by the pose estimation in the RGB and depth channels, respectively. Specifically, we first obtain a prediction of face detection and pose estimation in the depth input using a random decision forest-based approach proposed in [27]. With the estimated head position, we can crop the corresponding RGB image and get the face, in which we run a GBDT regression to obtain another prediction of the head rotation. The final result is obtained by fusing the two predictions.

This approach performs 3-D face detection and pose estimation in the depth input via Hough forests (Section II-B). The annotated depth images are provided to build the training patch set $\mathcal{P}$. Each training patch $P_i \in \mathcal{P}$ is denoted as $\{D_i, \lambda_i, \upsilon_i, \theta_i\}$, where D_i denotes the cropped patch from the depth input. The class label $\lambda_i \in \{-1, 1\}$ is a negative/positive sample indicator. υ indicates the offset from the current patch center to the object center. θ denotes the head rotation represented by Euler angle. In the training process, we need to learn a binary test for each internal node using a random binary pattern $\mathbf{u^1}(D_i : p_i^0, p_i^1, \tau_i)$ defined as

$$\mathbf{u^1}(D_i) = \begin{cases} 1, & \text{avg}\left\{D_{p_i^0}\right\} - \text{avg}\left\{D_{p_i^1}\right\} > \tau_i \\ 0, & \text{else} \end{cases} \tag{7}$$

where $\{p_i^0, p_i^1\}$ are sampling rectangles within the patch D_i and τ_i is the threshold.

At each internal node in the training process, we randomly generate a pool of random binary patterns $\{\mathbf{u}^1(\cdot)\}$ and select the one that produces the maximum discriminative performance.

For a patch set $\mathcal{P}$, we use the weighted entropy to measure the classification uncertainty U_c

$$U_c\left(\mathcal{P} : \mathbf{u}^1(\mathcal{D})\right) = -\sum_{\varphi \in \{L,R\}} w_\varphi \sum_{\lambda \in \{-1,1\}} p(\lambda|\mathcal{P}_\varphi) \ln(p(\lambda|\mathcal{P}_\varphi)) \tag{8}$$

where L and R denote the left child and the right child of the node, respectively, $w_{\varphi=L,R}$ is the percentage of patches falling into the left or the right child, and $p(\lambda|\mathcal{P}_\varphi)$ is the percentage of negative or positive patches in $\mathcal{P}_\varphi$.

A. Overview of Fanelli et al.'s [27] Approach

Similarly, the regression uncertainty U_r is defined on the head pose parameters (v, θ), which are modeled as two independent multivariate Gaussian distributions $v \sim N(\mu_v, \Sigma_v)$ and $\theta \sim N(\mu_\theta, \Sigma_\theta)$. The weighed differential entropy is used to define the regression uncertainty as

$$U_r\left(\mathcal{P} : \mathbf{u}^1(\mathcal{D})\right) = \sum_{\varphi\in\{L,R\}} w_\varphi \left(\log\left(|\Sigma_\varphi^v| + |\Sigma_\varphi^\theta|\right)\right) \tag{9}$$

where $|\Sigma|$ denotes the determinant of the covariance matrix Σ. Note that only positive patches are involved for U_r. We choose a best $\mathbf{u}^{1*}$ within the generated pool $\{\mathbf{u}^1\}$, which satisfies

$$\mathbf{u}^{1*} = \arg\min_{\mathbf{u}^1} \left(U_c\left(\mathcal{P} : \mathbf{u}^1(\mathcal{D})\right) + \left(1 - e^{-d/\omega}\right) U_r\left(\mathcal{P} : \mathbf{u}^1(\mathcal{D})\right)\right) \tag{10}$$

where d is the depth of the current node, and ω controls the steepness of the weighting function. A leaf of the tree stores: 1) the percentage of positive patches reaching the leaf $p(\lambda = 1|\mathcal{P})$ and 2) the multivariate Gaussian distribution $\{p(v|\mathcal{P}), p(\theta|\mathcal{P})\}$ for the head pose parameters of these positive training patches.

In the estimation process, patches are densely sampled from the depth input and are fed into the Hough forest. Only those valid patches that have a high probability to be positive $(p(\lambda = 1|P) > 0.95)$ are sent to voting for the face position v and pose parameter θ. The final detection and pose estimation result is obtained by leveraging all votes with the mean shift clustering algorithm.

B. Pose Estimation in the RGB Channel

Face pose estimation from the depth input is less sensitive to illumination change and clutter background. However, the depth input also has high noise levels compared with the RGB input, which degenerate the precision of pose estimation. To take advantage of the complementary observation from the RGB input, pose estimation is also performed in the RGB channel, where a GBDT regression is carried based on the randomized binary patterns. Mathematically, given the RGB input I

$$\theta(I) = \sum_{t=1}^{T} \rho h_t\left(\mathbf{u}_t^k(I)\right) \tag{11}$$

where ρ is the learning rate with a fixed value 0.1 for all the regression stages.

In particular, when the face position is available via face detection from the depth input, the corresponding RGB input is cropped, which contains only the face region. In the training process of head pose regression, the cropped face regions annotated with pose labels θ are collected, where $\theta = (\theta_{\text{pitch}}, \theta_{\text{yaw}}, \theta_{\text{roll}})$ represents the face rotation angles along the X, Y, and Z axes. We use the gray-scale input face patches and apply a global illumination normalization as a preprocessing step to reduce the effect of varying illumination. For the feature selection, we first generate a large pool of randomized binary patterns $\{\mathbf{u}_i^k(\cdot)\}_1^F$ (as shown in Fig. 4). Here, the size of the rectangles $\{p_i^0, p_i^1\}$ is randomized. In particular, the maximum width and height of the rectangles are kept less than a fixed scale (0.5 in our experiment) of the width and height of the cropped face region. At each stage of GBDT training, we extract the k-dimensional patterns $\mathbf{u}^\mathbf{k}$, as shown in Algorithm 1.

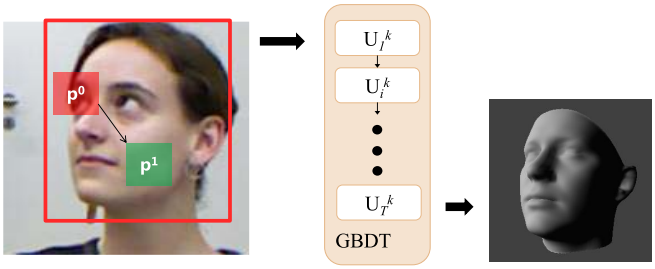


Fig. 4. Pose estimation in RGB input using GBDT.

In the test stage of the head pose estimation, there are only some patch comparison and lookup operations for a random binary pattern $\mathbf{u}^\mathbf{k}$, as shown in (11). They go through every stage in the GBDT extremely fast.

The prediction of the pose parameters in the RGB channel is combined with the one estimated in the depth channel by weighted summation. The weight of the prediction from the depth is defined as $w_d = \exp(-\text{trace}(|\Sigma_\theta|)/s)$, where $\text{trace}(|\Sigma_\theta|)$ represents the confidence of the prediction (defined in Section III-A) and s is a balance factor ($s = 900$ in our implementation). The weighted pose estimation takes the form

$$\theta = (1 - w_d)\theta_{\text{rgb}} + w_d\theta_d \tag{12}$$

where θ_{rgb} and θ_d are the predictions from the RGB and depth observations, respectively.

IV. Hierarchical Facial Landmark Localization

In our system, the facial landmark location is also estimated via regression. Specifically, we apply a hierarchical regression approach proposed next on the cropped face image. This facial landmark localization approach works on a cascaded random binary pattern regression framework, which consists of several groups of GBDTs, as shown in Fig. 5.

Due to the large changes of head pose and facial expressions, joint regression in the large target space is too difficult or needs a good initialization. We aim to reduce the image appearance variations gradually with hierarchical regression. A similar coarse-to-fine localization scheme is used in [3], where the rough facial component positions are first detected and further refined by a detailed face alignment procedure. There are two levels of facial landmarks in our proposed approach: 1) face component level and 2) facial landmark level, which are denoted as $\mathbf{s} = \{\mathbf{s}_c, \mathbf{s}_l\}$. In each level, we estimate the facial landmark locations using cascaded GBDTs.

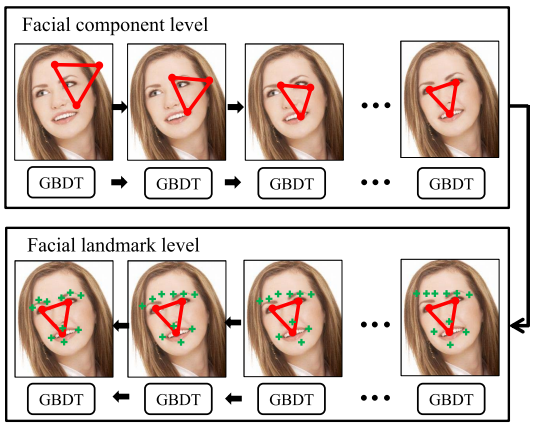
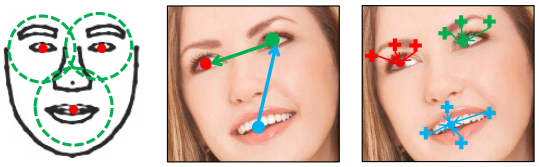


Fig. 5. Hierarchical regression working on a cascaded GBDT framework.



(a) (b)

Fig. 6. (a) Facial component level in the hierarchical regression. The red points are the positions. Green circles roughly indicate the sampling radius for the correspondent features. (b) Hierarchical configuration for the facial components and landmarks. A child is described by a displacement vector (the arrow) from it to its parent component.

A. Facial Component Level

The regression process first works on the component level, estimating the coarse locations of salient facial parts (e.g., eyes, nose, and mouth), as shown in Fig. 6(a). In particular, we define a hierarchical configuration for the facial components and landmarks, as shown in Fig. 6(b), instead of estimating all the facial landmark locations directly.

A parent component from the component level is assigned to each landmark based on the spatial distribution. We also define a parent for each facial component, except for the root. A child is described by a displacement vector from it to its parent component, so we need to estimate the displacement via the GBDT regression. We denote the facial component parts with the parameter $\mathbf{s}_c = \{o, v_1, v_2\}$, where o is the location of the root component (the left eye in our experiments), v_1 is the displacement from the right eye to the left eye, and v_2 is the displacement from the mouth to the right eye [Fig. 6(b)]. The landmark-level facial parameter $\mathbf{s}_l$ is defined similarly.

With a hierarchical facial landmark configuration, the sampling region of the random binary patterns is constraint to within the whole face region, as shown by the green circles in Fig. 6(a). Using of the whole face region for the random binary patterns sampling makes the facial component regression be robust to the variation of initial landmarks estimation caused by the face detection from depth input. The motivation of using the displacement vectors is that the variations of the relative positions are much smaller and the shape constraint is encoded implicitly in this scheme.

Given an input image I and the initial component parameter estimation $\mathbf{s}_c^0$ derived from the training procedure, the GBDT regression estimates the component parameter increment $\Delta\mathbf{s}_c$, which is additively merged into $\mathbf{s}_c$, as shown in Fig. 5.

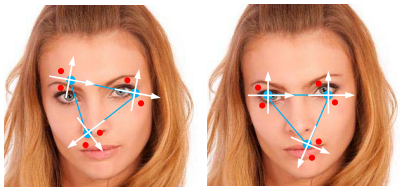


Fig. 7. Left and right images show the same parameter-indexed features. Red pixel pairs are indexed by homogeneous coordinates (white crosses) of current estimated components.

At the stage t of GBDT, the random binary patterns are sampled from the image I with the anchor pointed determined by the current facial component estimation $\mathbf{s}_c$. So, we have

$$\mathbf{s}_c^t = \mathbf{s}_c^{t-1} + h_t\left(u_t^k(I, \mathbf{s}_c^{t-1})\right), \quad t = 1, 2, \ldots, T. \tag{13}$$

In (13), the random binary pattern $u^k(I, \mathbf{s}_c)$ is slightly different from the previous random binary pattern $\mathbf{u}^k$, since a new control factor $\mathbf{s}_c$ from the previous stage of regression is introduced. The similar schemes are used in [23] and [36], where the pose/shape-indexed features are presented and used for regression. We define an associated homography matrix for each facial component and express the sampling pair $\{p^0, p^1\}$ of random binary patterns in homogeneous coordinates, as shown in Fig. 7.

We take a greedy approach in the training of the GBDT. At each stage t of the training process, each decision tree is sequentially trained while minimizing the parameter residual $\delta\mathbf{s}_c$ from the ground truth $\mathbf{s}_c^{\text{gt}}$ to the estimated value $\mathbf{s}_c^{t-1}$ from the previous training stage. The training process is described in the following steps.

1) Given the training images and their ground truth facial component locations, take the mean facial component location as the initial locations.
2) Randomly generate a pool of location-indexed binary features.
3) Train a GBDT as in Algorithm 1. The input is the pool of the generated random binary features $\{\mathbf{u}\}$ and the output is the facial component location residuals.
4) Update the current estimated parameter with the parameter increment predicted by the trained GBDT.
5) Repeat Steps 2–4 until the residual is unable to reduce or the maximum iteration is reached.

B. Facial Landmark Level

GBDT regression is also used in this stage. The training process for the GBDT in this level is similar to that in the upper level. The only difference is that the parameter-indexed random binary features are sampled within a smaller area (proportional to the distance between neighboring landmarks). This is to reduce the effect of nonrigid deformation and capture the features in a more detailed level. The whole procedure of localization at facial landmark level is shown in Fig. 5.

To get the initial facial landmark locations in a test face, we use the facial component locations estimated by the upper level and the mean displacement vectors in the training samples.

The testing instances go through the pretrained GBDTs and the landmark locations are obtained.

C. Pose Conditional GBDTs

With the estimated head pose θ, we can compute the conditional probabilities over the view space and estimate the landmark locations with conditional view-based GBDTs. We discretize the orientation space into disjoint sets $\{\Theta_i\}$. The Gaussian kernel is employed to estimate the distance $d(\theta, \Theta_i)$ between θ and Θ_i

$$d(\theta, \Theta_i) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{\|\theta - \omega_i\|^2}{2\sigma^2}\right) \tag{14}$$

where ω_i is the centroid of Θ_i and σ is the bandwidth parameter. To estimate the landmark locations, we use

$$u = \sum_{\Theta_i} u(i) P(\theta|\Theta_i) = \sum_{\Theta_i} u(i) \frac{d(\theta, \Theta_i)}{\Sigma_{\Theta_i} d(\theta, \Theta_i)} \tag{15}$$

where $u(i)$ is the landmark locations estimated by the GBDTs in the Θ_i view space.

V. Implementation and Experiments

We use various databases in the training and testing of the proposed system. To facility further discussion, we first introduce these databases briefly. After that, we perform the experiments in two parts: 1) head pose estimation and 2) facial landmark localization.

A. Databases

1) BIWI Kinect head pose database [27]: contains 24 sequences of different people recorded while sitting in front of a Microsoft Kinect camera roughly one meter away. These people rotate their heads to span all possible orientations. An offline template-based head tracker is used to label the 3-D head position and rotation. The range of the rotation angles is between $\pm 75°$ for yaw, $\pm 60°$ for pitch, and $\pm 50°$ for roll. The face region in a depth map can be obtained by projecting the 3-D template to the depth map with the pose parameters. In addition, depth values over 130 cm are set to zero.

2) Annotated facial landmarks in the wild (AFLW) database [38]: as its name implies, contains indoor and outdoor faces with large variations in head pose, lighting, and makeup. The images are collected from Flickr, an online photo sharing application. There are up to 21 annotated landmarks per face, and we choose 11 of them for our experiment. The landmarks and sample faces are shown in Fig. 8. We obtain the face bounding boxes from the annotated data and crop the face. The face images are then rescaled to 150×150 pixels.

3) EURECOM Kinect face database [37]: contains facial RGB-D images of 52 people obtained by Kinect. The data are captured in two sessions during different time periods. In each session, the facial images of each person are acquired in nine states of different expressions, lighting, and occlusion conditions. Fig. 9 shows some of these images and the labeled landmarks.

Fig. 8. Example faces of the training data for facial landmark localization. The left images show the labeled landmarks.

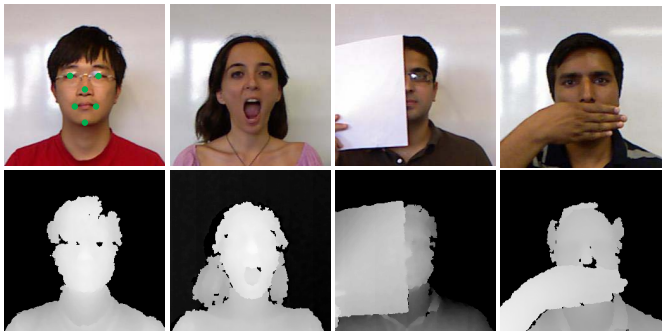


Fig. 9. Example RGB (top row) and depth (bottom row) images, as well as the labeled landmarks in the EURECOM Kinect face data set [37].

4) B3-D(AC)^2 face database [27]: consists of RGB-D face videos of 14 people, where each person utters a set of 40 sentences in front of a 3-D scanner, both in neutral and emotional tone. Over 100 K frames are fitted with a dense generic 3-D face model. By selecting a set of 13 landmarks on the generic template, we automatically obtain their 3-D locations in the depth images and the corresponding 2-D projections in the RGB images.

In our implementation, we use the BIWI Kinect head pose database [27] as our depth training data. The depth training process for the RF is the same as that in [27]. As for the GBDT-based head pose estimation, we use the RGB images of the above database. Before the RGB training process, we crop the face in an image based on the labeled head center. We apply some random displacements on the face locations to improve the robustness of the model. All the training images are rescaled to 150×150 pixels.

The parameters of a GBDT are: the number T of stages, the generated feature pool size F for training, the feature dimension k of each decision tree, and the R feature subsets from which we select the best in each stage. Here, we set $T = 5000$, $F = 10\,000$, $k = 5$, and $R = 20$ for the following experiments.

B. Pose Estimation in RGB-D Images

With the head position estimated in the depth map experiment, we crop the face images and run a GBDT to obtain a head rotation guess. Fig. 10 shows the mean errors of the three angles in different stages of the GBDT regression.

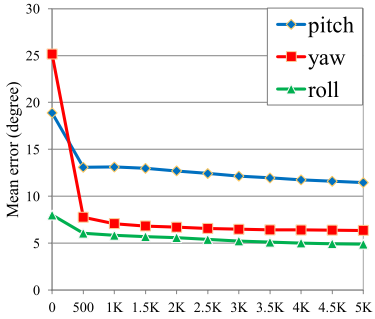


Fig. 10. Mean rotation errors with different number of random binary patterns in the GBDT regression.

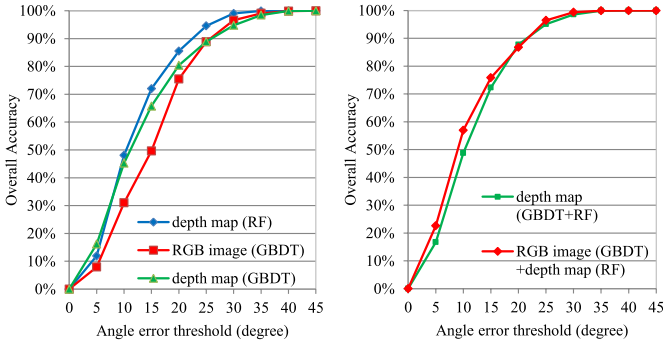


Fig. 11. Accuracy of overall head rotation estimation with different options. (a) Pose estimation from a single regression model. (b) Pose regression from multiple regression models fusion.

It shows that the errors converge quickly after passing through the 1000th random binary pattern. We compare the accuracy of the pose estimation results from the depth maps and the RGB images in Fig. 12. This is to show the performance of RGB and depth-based methods from different angles, and then demonstrate the effectiveness of our combination of RGB and depth input. In Fig. 12(a)–(c), we can observe that the results from the depth maps are less accurate than those from the RGB images for the yaw angle. The situation is reversed for the pitch angle. This is because the face texture decreases quickly as the pitch rotation increases. In this case, the depth cue is more useful. As for the yaw rotation, the depth cue does not have discrimination changes, especially for the noisy signal in our case. It shows that the results from RGB and depth input are complementary. In addition, both of the two methods give good results for the roll angle because the in-plane rotation can be well represented by both the RGB and depth cues. Then, we combine the two results and evaluate the overall accuracy in Fig. 12(d). Here, the rotation error is defined as the Euclidean distance of the three angle differences when compared with the ground truth. It shows that the accuracy of the RGB image-based method is lower than that of the depth map-based method. However, the result can be improved when we combine these two methods together.

We also measure the running time of the GBDT regression with C++ implementation on a desktop PC with a 3.2-GHz CPU and 4-GB RAM. It takes only 0.55 ms for an image. This extremely fast performance attributes to the comparison-based features and the decision tree structure. We should also note that the time consumption of this stage is negligible compared with that of the depth map-based approach, which costs about 30 ms in the same platform. That means, we improve the accuracy of pose estimation with just little extra cost.

To further clarify the effectiveness of the texture cue, we conduct the GBDT-based method on the depth maps. Similar to the experiment on the RGB images, we also need to crop the faces first. For comparison, Fig. 11(a) shows the results from different options. The GBDT regression on the depth maps provides more accurate result than on the RGB images. It is reasonable because the depth cue is more informative as in the previous experiment (Fig. 12). However, when combined with the RF approach, the comparison is reversed, as shown in Fig. 11(b). Note that the combination of the RF and the GBDTs for the depth map is similar to (12). So, the advantage of our combination of the RGB image and depth cues is further demonstrated.

C. Facial Landmark Localization in RGB-D Images

First, we give a qualitative evaluation on the generalization ability of the proposed facial landmark localization approach. We take the AFLW database [38] as the training data and test on the BIWI Kinect head pose database. The training configuration is as follows: we use ten GBDTs in each level; every GBDT contains 500 random binary patterns; the dimension of each random binary pattern is five. We divide the training samples into three subsets based on the yaw rotation and train three GBDTs for the pose conditional estimation.

Fig. 15 shows some of the results in the RGB images from the BIWI Kinect head pose database. We can see that the algorithm can deal with cases with different head rotations. In addition, we add some random occlusions in the frames and test the algorithm. In these cases, the location accuracy decreases. However, the landmarks can still be roughly located in the occlusion areas. This is because the facial shape prior is encoded in the cascaded regression process, and the partial occlusion does not destroy the global configuration. The running time for a face image in the test is about 20 ms. Together with the head pose estimation and other I/O operation cost, the system achieves 20 frames/s.

We conduct quantitative evaluation for the accuracy of facial landmark localization. First, we use the annotations created in [39], which covers 82 frames in the BIWI database. We compare our method with that of [1], which uses conditional RF regression to locate facial landmarks in RGB images. To be fair, here we use the same training data set as [1] and test on 10 facial landmarks including four eye corners, four mouth corners, and two nose strips. Fig. 13(a) shows the accuracy of the two methods. It is demonstrated that our algorithm outperforms this state-of-the-art technique. With the use of the depth information, we obtain the head pose information that can facilitate the landmark localization task. In Fig. 13(b), we show the performance over the ten GBDTs of facial landmark level. The accuracy converges with just a few iterations. In addition, to show the superiority of pose conditional GBDT, we compute the relative improvement over the single GBDT and list it in Fig. 13(c). We obtain 2%–16% improvement on the localization accuracy rate for the ten facial landmarks.

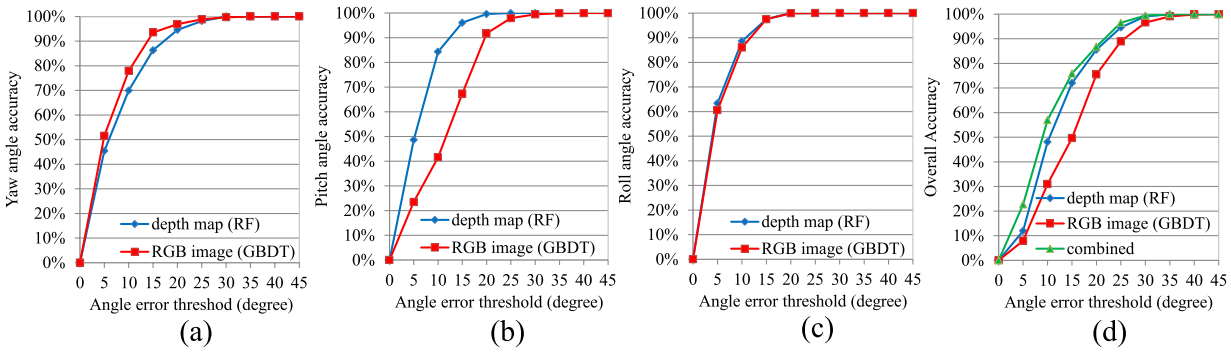


Fig. 12. (a)–(c) Accuracy of three head rotation estimation methods with different thresholds. (d) Accuracy of the overall head rotation estimation (we define the angle error as the Euclidean distance to the ground truth).

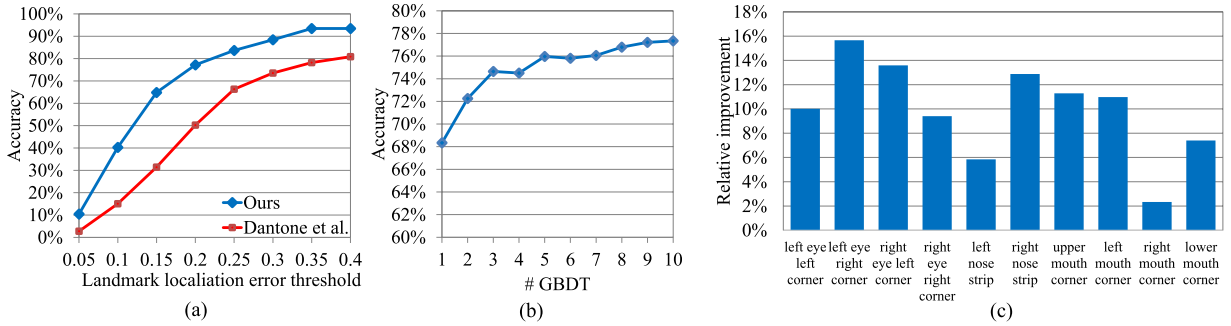


Fig. 13. (a) Comparison of the accuracy between our method with [1]. The thresholds for computing the accuracy are normalized by the interocular distance. (b) Accuracy in the facial landmark level over different stages of cascaded GBDTs. The threshold is set to 0.2. (c) Relative improvement of the localization accuracy with pose conditional GBDT over single GBDT. The improvement is computed as the reduced error normalized by the single-GBDT error.

The facial landmark localization experiment is also performed on B3-D(AC)^2. In the experiment, all RGB-D frames from 14 persons are used for the localization task. At the head detection and pose estimation stage, 124 520 frames are successfully detected and used for the facial landmark localization stage. Since our approach outputs the 2-D fiducial points, we need to project these locations back to 3-D depth space for a fair comparison with [31]. For each RGB and depth image pair, we use the depth and RGB calibration data provided in B3-D(AC)^2 to calculate a pixel-wise registration. For each localized facial landmark in the RGB image, we find its depth correspondence and project it back to the 3-D depth space.

The localization errors from fivefold cross validation are given in Table I. We notice that there is an obvious performance drop on the nose component localization. The main reason of this phenomenon is that many shadows around nose regions for most of the samples. Since less discriminative features can be found within the shadow region for precise nose localization, it is a good way to rely on the global face shape constraint for nose localization estimation, as those done in [31].

For comprehensiveness, we also use the EURECOM Kinect face database [37] to verify the effectiveness of our hierarchical method. We use the same training configuration as in the previous experiment, and compare the hierarchical and nonhierarchical versions of our algorithm. The nonhierarchical algorithm runs in the facial landmark level directly, without the facial component level. For fairness, the total numbers of the cascaded GBDTs in both the hierarchical and nonhierarchical versions are the same (i.e., 20 GBDTs). In Fig. 14, we can observe that the hierarchical approach outperforms the nonhierarchical one. In addition, because the regression dimension in the first hierarchical level is much lower than in the second level, the hierarchical method can also save much computation time.

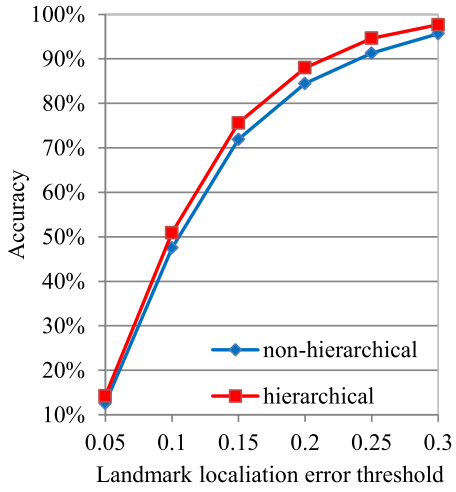


Fig. 14. Comparison of the accuracy between the hierarchical and nonhierarchical methods for facial landmark localization. The thresholds for computing the accuracy are normalized by the interocular distance.

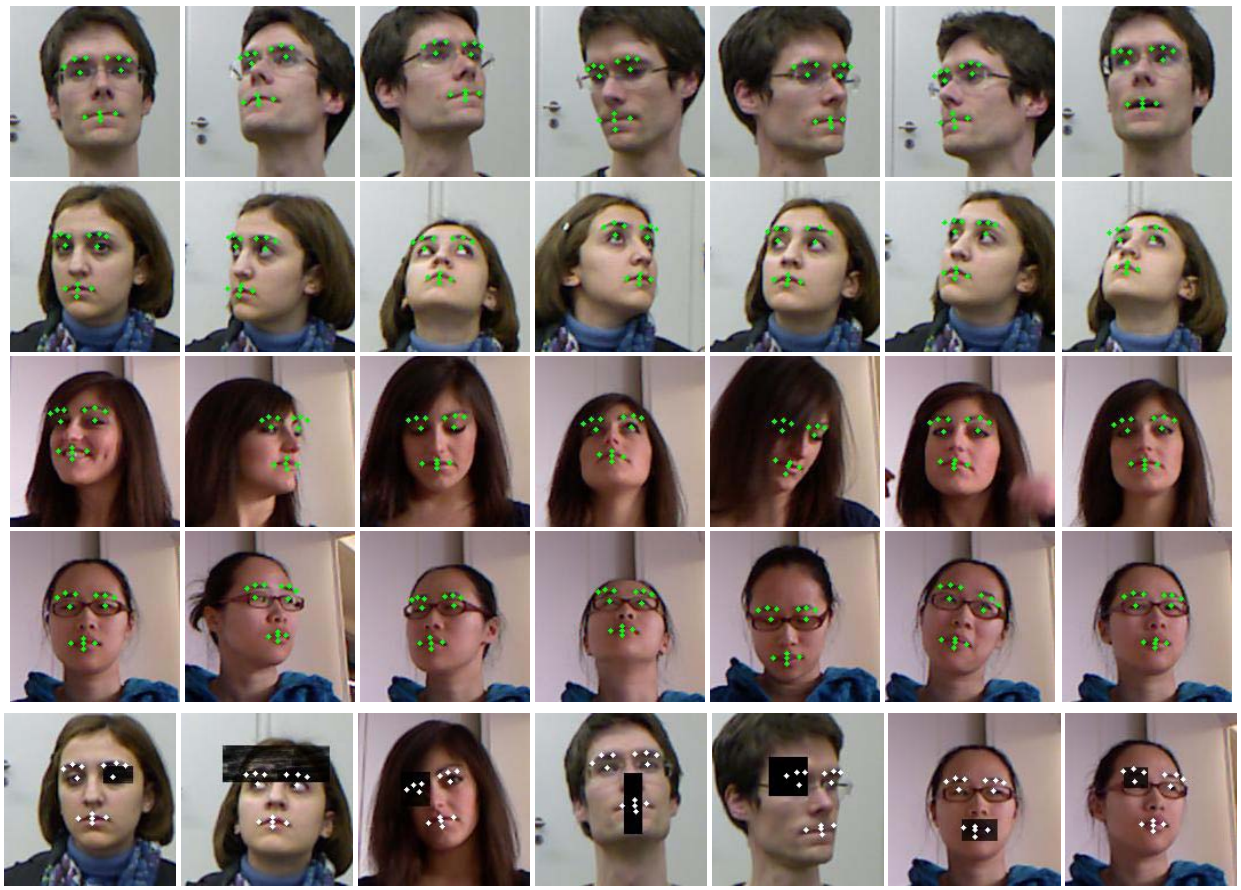


Fig. 15. Typical facial landmark localization results in the BIWI Kinect head pose database [27]. The last row contains facial landmark localization results with random partial occlusions.

TABLE I
RESULTS OF FIVEFOLD CROSS VALIDATION ON THE B3-D(AC)^2 DATABASE

fiducial	success rate (5/10mm) **[31]**	mean±std error **[31]**	success rate (5/10mm)	mean±std error
innEyeL	86.0/98.1	3.4 ± 2.1	83.6/99.0	3.0 ± 2.0
outEyeL	65.7/99.0	4.4 ± 1.9	95.0/99.9	2.1 ± 1.4
innEyeR	77.9/99.4	3.9 ± 1.9	91.0/99.7	2.6 ± 1.7
outEyeR	54.9/98.3	4.8 ± 2.3	85.6/99.5	3.0 ± 1.9
noseL	83.8/99.9	3.3 ± 1.7	55.6/76.4	3.6 ± 2.6
noseR	90.1/100.0	2.9 ± 1.7	62.0/75.1	2.9 ± 2.4
mouthL	54.5/90.7	5.3 ± 3.1	78.3/93.0	3.1 ± 1.8
upLip	82.1/99.8	3.5 ± 1.7	90.7/99.1	2.7 ± 1.6
mouthR	50.3/92.8	5.3 ± 2.9	94.4/99.0	2.3 ± 1.5
lowLip	61.7/82.1	6.2 ± 5.8	66.2/74.9	2.7 ± 1.7

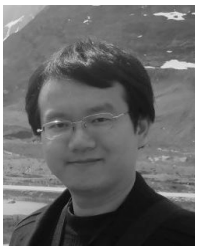
VI. CONCLUSION

In this paper, we have proposed a robust and real-time multiview facial landmark localization in RGB-D images. The localization problem is formulated as a regression problem and we present a hierarchical approach to deal with the high dimensional face landmark localization. Different from many existing methods, the proposed approach estimates both the head pose and the facial landmark locations sequentially with a unified regression framework. The experiments show that the combination of the RGB images and the depth maps can improve the head pose estimation accuracy. In addition, the proposed hierarchical pose regression can locate facial landmarks in the cases of large rotations and partial occlusions, outperforming the state-of-the-art techniques.

REFERENCES

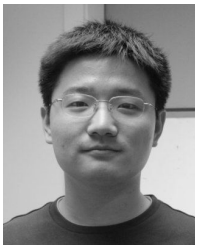
[1] M. Dantone, J. Gall, G. Fanelli, and L. Van Gool, "Real-time facial feature detection using conditional regression forests," in *Proc. IEEE Conf. CVPR*, Jun. 2012, pp. 2578–2585.
[2] W. Zhang, Q. Wang, and X. Tang, "Real time feature based 3-D deformable face tracking," in *Proc. 10th ECCV*, 2008, pp. 720–732.
[3] L. Liang, R. Xiao, F. Wen, and J. Sun, "Face alignment via component-based discriminative search," in *Proc. ECCV*, 2008, pp. 72–85.
[4] T. C. Patrick Sauer and C. Taylor, "Accurate regression procedures for active appearance models," in *Proc. BMVC*, 2011, pp. 30.1–30.11.
[5] Y. Huang, Q. Liu, and D. Metaxas, "A component based deformable model for generalized face alignment," in *Proc. IEEE 11th ICCV*, Oct. 2007, pp. 1–8.
[6] A. Blake and M. Isard, *Active Shape Models*. New York, NY, USA: Springer-Verlag, 1998.
[7] D. Cristinacce and T. Cootes, "Boosted regression active shape models," in *Proc. BMVC*, vol. 2. 2007, pp. 880–889.

[8] M. Valstar, B. Martinez, X. Binefa, and M. Pantic, "Facial point detection using boosted regression and graph models," in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 2729–2736.
[9] J. M. Saragih, S. Lucey, and J. F. Cohn, "Face alignment through subspace constrained mean-shifts," in *Proc. IEEE ICCV*, Sep. 2009, pp. 1034–1041.
[10] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Proc. IEEE Conf. CVPR*, Jun. 2012, pp. 2879–2886.
[11] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 545–552.
[12] D. Vukadinovic and M. Pantic, "Fully automatic facial feature point detection using Gabor feature based boosted classifiers," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, vol. 2. Oct. 2005, pp. 1692–1698.
[13] T. Cootes, G. Edwards, and C. Taylor, "Active appearance models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 681–685, Jun. 2001.
[14] X. Liu, "Generic face alignment using boosted appearance model," in *Proc. IEEE Conf. CVPR*, Jun. 2007, pp. 1–8.
[15] J. Saragih and R. Goecke, "A nonlinear discriminative approach to AAM fitting," in *Proc. IEEE 11th ICCV*, Oct. 2007, pp. 1–8.
[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[17] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *Proc. 11th ECCV*, 2010, pp. 778–792.
[18] V. Lepetit, P. Lagger, and P. Fua, "Randomized trees for real-time keypoint recognition," in *Proc. IEEE Conf. CVPR*, vol. 2. Jun. 2005, pp. 775–781.
[19] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. IEEE ICCV*, Nov. 2011, pp. 2564–2571.
[20] S. Leutenegger, M. Chli, and R. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *Proc. IEEE ICCV*, Nov. 2011, pp. 2548–2555.
[21] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast retina keypoint," in *Proc. IEEE Conf. CVPR*, Jun. 2012, pp. 510–517.
[22] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in *Proc. IEEE Conf. CVPR*, Jun. 2013, pp. 532–539.
[23] X. Cao, Y. Wei, F. Wen, and J. Sun, "Face alignment by explicit shape regression," in *Proc. IEEE Conf. CVPR*, Jun. 2012, pp. 2887–2894.
[24] T. Cootes, M. Ionita, C. Lindner, and P. Sauer, "Robust and accurate shape model fitting using random forest regression voting," in *Proc. ECCV*, Oct. 2012, pp. 278–291.
[25] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *Proc. IEEE Conf. CVPR*, Jun. 2013, pp. 3476–3483.
[26] J. Shotton *et al.*, "Real-time human pose recognition in parts from single depth images," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 1297–1304.
[27] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Gool, "Random forests for real time 3D face analysis," *Int. J. Comput. Vis.*, vol. 101, no. 3, pp. 437–458, 2013.
[28] M. Breitenstein, D. Kuettel, T. Weise, L. Van Gool, and H. Pfister, "Real-time face pose estimation from single range images," in *Proc. IEEE Conf. CVPR*, Jun. 2008, pp. 1–8.
[29] T. Weise, S. Bouaziz, H. Li, and M. Pauly, "Realtime performance-based facial animation," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 77:1–77:10, 2011.
[30] Q. Cai, D. Gallup, C. Zhang, and Z. Zhang, "3D deformable face tracking with a commodity depth camera," in *Proc. 11th ECCV*, 2010, pp. 229–242.
[31] G. Fanelli, M. Dantone, and L. V. Gool, "Real time 3D face alignment with random forests-based active appearance models," in *Proc. 10th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit.*, Apr. 2013, pp. 1–8.
[32] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001.
[33] A. Criminisi, J. Shotton, and E. Konukoglu, "Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning," *Found. Trends Comput. Graph. Vis.*, vol. 7, pp. 81–227, Feb. 2012.
[34] J. Gall and V. Lempitsky, "Class-specific hough forests for object detection," in *Proc. IEEE Conf. CVPR*, Jun. 2009, pp. 1022–1029.
[35] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, 2000.
[36] P. Dollar, P. Welinder, and P. Perona, "Cascaded pose regression," in *Proc. IEEE Conf. CVPR*, Jun. 2010, pp. 1078–1085.
[37] T. Huynh, R. Min, and J.-L. Dugelay, "An efficient LBP-based descriptor for facial depth images applied to gender recognition using RGB-D face data," in *Proc. ACCV Workshops*, New York, NY, USA, 2013, pp. 133–145.
[38] M. Kostinger, P. Wohlhart, P. Roth, and H. Bischof, "Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization," in *Proc. IEEE ICCV Workshops*, Nov. 2011, pp. 2144–2151.
[39] T. Baltrusaitis, P. Robinson, and L. Morency, "3D constrained local model for rigid and non-rigid facial tracking," in *Proc. IEEE Conf. CVPR*, Jun. 2012, pp. 2610–2617.



Zhanpeng Zhang (S'12) received the B.E. and M.E. degrees from Sun Yat-sen University, Guangzhou, China, in 2010 and 2013, respectively. He is currently working toward the Ph.D. degree with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong.

His research interests include computer vision and machine learning, in particular, face tracking and analysis.

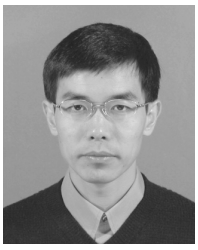


Wei Zhang (M'11) received the B.S., M.E., and Ph.D. degrees in computer engineering from Nankai University, Tianjin, China; Tsinghua University, Beijing, China; and The Chinese University of Hong Kong, Hong Kong, in 2002, 2005, and 2010, respectively.

He was with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China, as an Assistant Researcher, in 2011. He is a Researcher with the Media Technology Laboratory, Huawei Technologies Company Limited, Shenzhen. His research interests include computer vision, pattern recognition, and video processing.

Jianzhuang Liu (SM'02) received the Ph.D. degree in computer vision from The Chinese University of Hong Kong, Hong Kong, in 1997.

He was a Research Fellow with Nanyang Technological University, Singapore, from 1998 to 2000. From 2000 to 2012 he was a Post-Doctoral Fellow, then an Assistant Professor, and then an Adjunct Associate Professor with The Chinese University of Hong Kong. He was with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, as a Professor, in 2011. He is currently a Chief Scientist with Huawei Technologies Company Limited, Shenzhen. He has authored more than 100 papers, most of which are in prestigious journals and conferences in computer science. His research interests include computer vision, image processing, machine learning, multimedia, and graphics.



Xiaoou Tang (F'09) received the B.S. degree from University of Science and Technology of China, Hefei, China; the M.S. degree from the University of Rochester, Rochester, NY, USA; and the Ph.D. degree from Massachusetts Institute of Technology, Cambridge, MA, USA, in 1990, 1991, and 1996, respectively.

He is a Professor with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong. He was the Group Manager of the Visual Computing Group with the Microsoft Research Asia, Beijing, China, from 2005 to 2008. His research interests include computer vision, pattern recognition, and video processing.

Dr. Tang was a Program Chair of the 2009 IEEE International Conference on Computer Vision and an Associate Editor of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and *International Journal of Computer Vision*. He received the Best Paper Award at the 2009 IEEE Conference on Computer Vision and Pattern Recognition.