Telecommun Syst (2011) 47:215–225
DOI 10.1007/s11235-010-9313-3

Monocular camera-based face liveness detection by combining eyeblink and scene context

Gang Pan · Lin Sun · Zhaohui Wu · Yueming Wang

Published online: 4 August 2010

Abstract This paper presents a face liveness detection system against spoofing with photographs, videos, and 3D models of a valid user in a face recognition system. Anti-spoofing clues inside and outside a face are both exploited in our system. The inside-face clues of spontaneous eyeblinks are employed for anti-spoofing of photographs and 3D models. The outside-face clues of scene context are used for anti-spoofing of video replays. The system does not need user collaborations, i.e. it runs in a non-intrusive manner. In our system, the eyeblink detection is formulated as an inference problem of an undirected conditional graphical framework which models contextual dependencies in blink image sequences. The scene context clue is found by comparing the difference of regions of interest between the reference scene image and the input one, which is based on the similarity computed by local binary pattern descriptors on a series of fiducial points extracted in scale space. Extensive experiments are carried out to show the effectiveness of our system.

G. Pan · L. Sun · Z. Wu
Department of Computer Science, Zhejiang University,
Hangzhou, China

G. Pan
e-mail: gpan@zju.edu.cn

L. Sun
e-mail: sunlin@zju.edu.cn

Z. Wu
e-mail: wzh@zju.edu.cn

Y. Wang (✉)
Qiushi Academy for Advanced Studies, Zhejiang University,
Hangzhou, China
e-mail: ymwang@zju.edu.cn

Keywords Face recognition · Liveness detection · Face spoofing · Video-replay attack

1 Introduction

Biometrics is an emerging technology that enables recognizing humans based upon one or more intrinsic physiological or behavioral characteristics, including faces, fingerprints, irises, voices, etc. [1]. However, spoofing attack (or copy attack) is still a fatal threat for biometric authentication systems [2] because our faces are visible, our voices are recordable, and fingerprints are left everywhere we go. Even though many high performance biometric technologies for identification and verification have been developed, these measurements rarely indicate liveness. Liveness detection, whose goal is to distinguish between spoofing attack and live person, is becoming an active topic in the field of biometrics, such as fingerprint, iris, and face recognition [2–5].

In face recognition community, numerous recognition approaches have been proposed, but the effort on anti-spoofing is still very limited [6]. Attackers in face recognition usually use photos, videos, or 3D models of a valid user to fake face recognition systems: (1) The facial photograph is the most common way to spoof face recognition systems, as usually it is relatively easy to obtain a valid user's photos from public. The main characteristic of the photograph is that it is a planar object without varying facial expressions and three dimensional information. (2) Videos of valid users are also not difficult to get nowadays thanks to high quality pinhole cameras. Spoofing videos have more physiological clues than photos, such as eyeblinks, facial expressions, and head movements. The difficulty of detecting spoofing video is that it is a re-imaging of the original live face. High quality spoofing videos are almost the same as live faces in a non-intrusive scenario. (3) Tangible 3D models of a valid user, such as wax, are much more difficult to obtain than videos and photos. The trait of wax is that it has distinct 3D structure which videos and photos do not have. 3D model can imitate rigid head motions by rotation. However, facial expressions, such as blinks and lip movements, are very difficult to be imitated because it is hard for 3D model to act non-rigid motions well.

The goal of this paper is to develop a real-time liveness detection system to resist common spoofing attackers in a non-intrusive manner for face recognition. Most of the current face recognition systems are based on intensity images and equipped with a camera. Our anti-spoofing method shares the camera with the existing face recognition systems and does not need any additional devices, which makes it more preferable.

The contributions of this paper are three-fold as follows.

1. The *scene context clue* is employed for the liveness detection. The scene context clue is outside face while all the clues in literature are inside face. In our system, the camera is assumed to be stationary. Considering spoofing clues occurring outside a face, such as the screen border of a display and the body of an attacker, the region around the detected face of an input is extracted to match the reference scene.
2. We propose the combination of the clues of eyeblinks and scene context for liveness detection. The advantages of the combination are non-intrusion, no extra hardware requirements, and real-time efficiency. The scene context clue is to detect video imposters that the eyeblink clue can not detect, while the eyeblink clue is to detect trimmed photograph imposters that the scene context clue can not detect.
3. An effective scene context matching approach for the liveness detection is presented. To cope with common changes of light conditions and slight shifts of the camera, in our approach, a series of fiducial points are extracted in image scale space as the intermediate representation of a scene, which illustrate the notable objects in the scene and outperform non-fiducial-point approaches.

This paper is structured as follows. Section 2 reviews the previous work and discusses liveness clues. Then our framework of the fusion of eyeblinks and scene context for the liveness detection is presented in Sect. 3. Detection of eyeblinks and scene context clue is described detailedly in Sects. 4 and 5 respectively. Section 6 shows the experimental results. Section 7 concludes this paper.

2 Related work

There are two categories of liveness approaches proposed in the literature: non-intrusive methods and intrusive ones. Intrusive methods need users' response to system actions, such as uttering words [7, 8], rotating head in a certain direction [9], while non-intrusive methods does not need. Non-intrusive approaches exploit spontaneous physiological activities or physical phenomena on a face, such as 3D geometry properties [10, 11], eyeblinks [12], spontaneous non-rigid deformation [16], Fourier spectra [13], and thermogram. For non-intrusive approaches, users usually do not know which liveness clues are used in face recognition systems.

Non-intrusive methods against photo spoofing often utilize the physical phenomenon that photograph is two dimensional planar structure while live face is a three dimensional object. Choudhary et al. [10] employs the structure from motion yielding the depth information of the face to differentiate live person heads and still photos. Kollreider et al. [11] exploits the motion characteristics of a 3D face by optical flow. It assumes that a 3D face generated a special 2D motion which is higher at central face parts (e.g. nose) than the outer face regions (e.g. ears). However, video imposters also have these 2D motion characteristics, since a video imposter before a camera is re-imaging of the original live face. Another disadvantages of using depth information are that depth estimation is sensitive to light variations.

Physiological activities also can be used as clues of non-intrusive liveness detection. Pan et al. [12] proposes an eyeblink-based face liveness detection approach in non-intrusive manner. The eyeblink is a physiological activity which is an essential physiological function of eyes. The eyeblink is detected by an undirected conditional graphical model. However, the liveness detection based on physiological activities in a non-intrusive environment fails to prevent video imposters.

The phenomenon of quality degradation when re-imaging is another clue for non-intrusive liveness detection. Li et al. [13] uses Fourier spectra to classify live faces or the faked images, based on the assumption that the high frequency components of the photo is less than those of live face images. However, it is sensitive to the lighting condition and vulnerable to photograph of high quality. With an infrared camera, the vein map or thermogram of a face could also be applied to liveness detection [14].

Intrusive approaches usually ask users to respond to some actions specified by the system. The method presented by Frischholz et al. [9] requires the user to rotate the head into a certain direction which is randomly chosen by the system. For this method, attackers can select prepared videos including all head poses that will be tested or rotate 3D model to response random challenges. Uttering a specific digit sequence prompted randomly is adopted by Kollreider et al. [8]. Digits $0\ldots 9$ are recognized one by one in a sequence from lip motions. Ten lip motion classes of uttering digits $0\ldots 9$ are trained by 10-class SVM using optical

Table 1 Comparison of anti-spoofing clues

Clues	Additional hardware	User collaboration	Irresistible imposter
Facial expression	No	Middle	Video
3D properties	No	Middle	Video & 3D model
Mouth movement	No	Middle	Video
Fourier spectra	No	Low	–
Facial thermogram	Yes	Low	–
Facial vein map	Yes	Low	–
Interactive response	No	High	Video & 3D model
Eye blinking	No	Low	Video

flow features. Some multi-modal approaches [7, 15] also use the interactive manner of speaking. They use three features of faces, voices, and lip movements to obtain higher recognition accuracy and security than single modality. The lip movement is used to resist photo and voice record attacks simultaneously.

Table 1 summaries these anti-spoofing clues for comparison in terms of additional hardware, user collaborations, and irresistible imposters. In the literature, there is yet no good non-intrusive anti-spoofing approach against video replay imposters with a generic camera.

3 Fusion of eyeblink and scene context for liveness detection

Here we present an approach combining two liveness clues of eyeblinks and scene context against spoofing in face recognition systems. We categorize liveness clues into two kinds: the inside-face clue and the outside-face clue. Most of previous work focuses on inside-face clues, such as head movements, facial expressions, Fourier spectra, and eyeblinks. However, inside-face clues are difficult to detect video replay spoofing. We believe that the outside-face clue of scene context is an important hint for anti-spoofing of video replays, since the characteristics of scene context depend on the intrinsic and extrinsic parameters of the camera and the local scene where the face recognition system is deployed. Outside-face clues can serve as an effective supplement to inside-face clues.

We assume that the camera used by a face recognition system is fixed during the system is working. Then, it is obvious that the scene image without any persons, called *reference scene*, is same in background part as the one when a live human stands in front of the camera. However, the scenes where imposter videos were taken are usually different from the reference scene. Therefore, it is feasible to detect spoofing by modeling the difference between the input and reference scenes. We call this kind of clues *scene context*. The scene context clue needs neither user collaborations nor extra hardware. It is helpful to reject video replay imposters as well as photo imposters.

The eyeblink is a spontaneous physiological activity of rapid closing and opening of eyelids, which is an essential function of eyes that helps to spread tears across and remove irritants from the surface of the cornea and conjunctiva. Although blink speed can vary with elements such as fatigue, emotional stress, behavior category, amount of sleep, eye injury, medication, and disease, researchers report that [17, 18], the spontaneous resting blink rate of a human being is nearly from 15 to 30 eyeblinks per minute. That is, a person blinks approximately once every 2 to 4 seconds and the blink lasts about 250 milliseconds on average. Thus, it is easy for a generic camera to capture two or more frames for each blink when the face looks into the camera. It is feasible to adopt eyeblinks as a clue inside face.

The flowchart of the liveness detection system combining eyeblinks and scene context is shown in Fig. 1. The red rectangles in the figure indicate the regions where the blink and scene context clues occur respectively. If both clues succeed, the system will consider the input face as live face (No. 2 output in Fig. 1). If either of two clues fails, imposter will be considered. The No. 1 output in Fig. 1 is mainly caused by photo and 3D model imposters and No. 3 output in Fig. 1 is mainly caused by video and photo imposters.

4 Eyeblink detection

We have modeled eyeblink behaviors [12] in a Conditional Random Field framework [19], incorporated with a discriminative measure of eye states. Here we give a brief introduction, and the details can be referred in [12].

An eyeblink activity can be represented by an image sequence $\mathbb{S}$ consisting of T images, where $\mathbb{S} = \{I_i, i = 1, \ldots, T\}$. The typical eye states are *opening* and *closing*. In addition, there is an ambiguous state when blinking from open state to close or from close state to open. We define a three-state set for eyes, $\mathcal{Q} = \{\alpha : open, \gamma : close, \beta :$

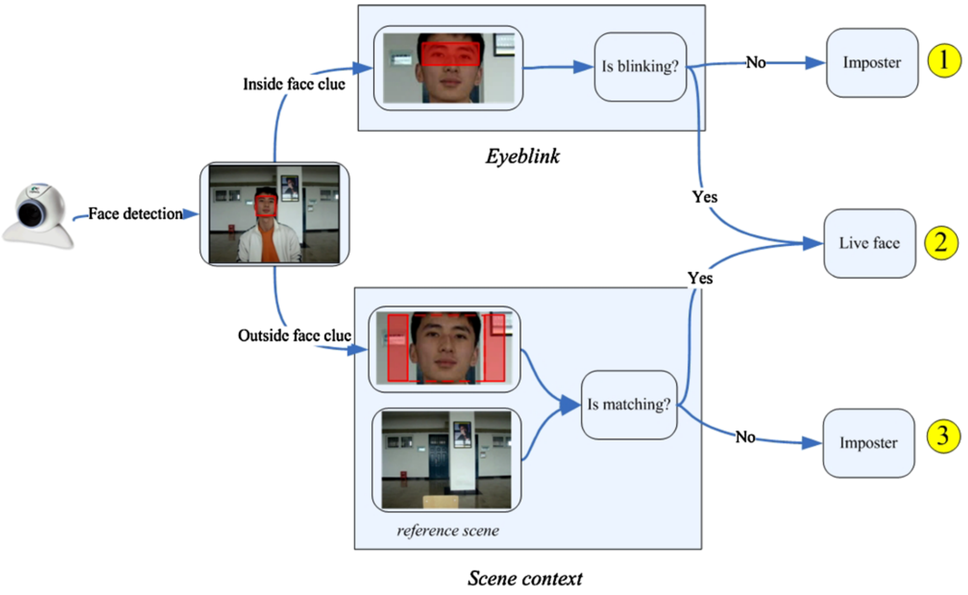


Fig. 1 Illustration of liveness detection system using a combination of eyeblinks and scene context. *Yellow circle labels* (*1*, *2*, *3*) indicate three result branches according to live faces and imposters of photographs, videos or 3D models

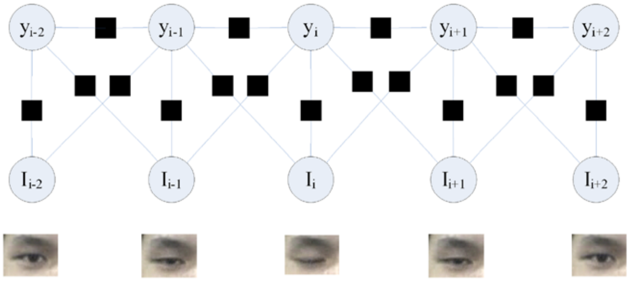


Fig. 2 Illustration of graphical structures for eyeblink. Graphical model of a linear-chain CRF, where the *circles* are variable nodes and the *black square* boxes are factor nodes, in this example the state is conditioned on contexts of 3 neighboring observations, that is, $W = 1$

ambiguous}. Thus, a typical blink activity can be described as a state change pattern of $\alpha \to \beta \to \gamma \to \beta \to \alpha$.

Suppose that $\mathbb{S}$ is a random variable over observation sequences to be labeled and Y is a random variable over the corresponding label sequences to be predicted. All of components y_i of Y are assumed to range over a finite label set $\mathcal{Q}$.

We yield a linear chain structure, shown in Fig. 2. In this graphical model, a parameter of observation window size W is introduced to describe the conditional relationship between the current state and $(2W + 1)$ temporal observations around the current one. Using the Hammersley and Clifford theorem [20], the joint distribution over the label sequence Y given the observation $\mathbb{S}$ can be written as the following form:

$$p_\theta(Y|\mathbb{S}) = \frac{1}{Z_\theta(\mathbb{S})} \exp\left(\sum_{t=1}^{T} \Psi_\theta(y_t, y_{t-1}, \mathbb{S})\right), \tag{1}$$

where $Z_\theta(\mathbb{S})$ is a normalized factor summing over all state sequences and an exponentially large number of terms,

$$Z_\theta(\mathbb{S}) = \sum_Y \exp\left(\sum_{t=1}^{T} \Psi_\theta(y_t, y_{t-1}, \mathbb{S})\right). \tag{2}$$

The potential function $\Psi_\theta(y_t, y_{t-1}, \mathbb{S})$ is the sum of CRF features at time t:

$$\Psi_\theta(y_t, y_{t-1}, \mathbb{S}) = \sum_i \lambda_i f_i(y_t, y_{t-1}, \mathbb{S}) + \sum_j \mu_j g_j(y_t, \mathbb{S}), \tag{3}$$

with parameters $\theta = \{\lambda_1, \ldots, \lambda_A; \mu_1, \ldots, \mu_B\}$, to be estimated from training data [12].

The f_i and g_j are *within-label* and *between-observation-label* feature functions, respectively. λ_i and μ_j are the feature weights associated with f_i and g_j. The *within-label* feature functions f_j are as:

$$f_i(y_t, y_{t-1}, \mathbb{S}) = \mathbf{1}_{\{y_t = l\}} \mathbf{1}_{\{y_{t-1} = l'\}}, \tag{4}$$

where $l, l' \in \mathcal{Q}$, and $\mathbf{1}_{\{x=x'\}}$ denotes an indictor function of x which takes the value 1 when $x = x'$ and 0 otherwise. Given a temporal window size W around the current observation, the *between-observation-label* feature functions g_j are as:

$$g_j(y_t, \mathbb{S}) = \mathbf{1}_{\{y_t = l\}} \mathcal{U}(I_{t-w}), \tag{5}$$

where $l \in \mathcal{Q}$, $w \in [-W, W]$, and $\mathcal{U}(I)$ is *eye closity*, described later. Here, feature functions f_i and g_j are based on conjunctions of simple rules.

Motivated by the idea of the adaptive boosting algorithm [22, 23], we define a real-value discriminative feature for the

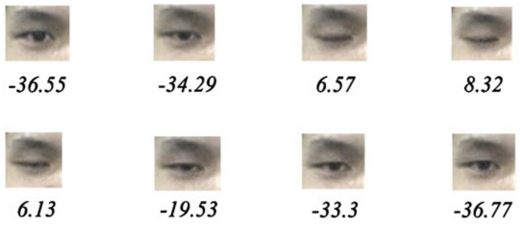


Fig. 3 Illustration of the closity for a blinking activity sequence. The closity value of each frame is below the corresponding frame. Bigger the value, higher the degree of closeness

eye image, called *eye closity*, $\mathcal{U}(I)$, measuring the degree of eye's closeness. It is constructed by a linear ensemble of a series of weak binary classifiers and computed by an iterative procedure.

$$\mathcal{U}_M(I) = \sum_{i=1}^{M}\left(\log\frac{1}{\beta_i}\right)h_i(I) - \frac{1}{2}\sum_{i=1}^{M}\log\frac{1}{\beta_i}, \tag{6}$$

where,

$$\beta_i = \epsilon_i/(1-\epsilon_i) \tag{7}$$

and $\{h_i(I) : R^{\mathrm{Dim}(I)} \rightarrow \{0,1\}, i = 1,\ldots,M\}$ is a set of binary weak classifiers. Each classifier h_i is for classifying the input I as the open eye: $\{0\}$, or the close eye: $\{1\}$. Given a set of labeled training data, the efficient selection of h_i and the calculation of ϵ_i can be performed by an iterative procedure similar to the adaptive boosting algorithm in [23].

A blinking activity sequence is shown in Fig. 3, where the value is closity of the corresponding image, computed after training nearly by 1,000 samples of open eyes and 1,000 samples of close eyes. The bigger the value of *closity*, the higher the degree of eye closeness.

The inference tasks to label an unknown eye image sequence $\mathbb{S}$ is satisfied by $Y^* = \mathrm{argmax}_Y\ \ p(Y|\mathbb{S})$, which can be performed efficiently and exactly using dynamic programming methods for HMM [21]. The eyeblink pattern can be written in regular expression,

$$\alpha\beta?\gamma + \beta?\alpha, \tag{8}$$

where '?' indicates there is zero or one of the preceding element and '+' indicates there are one or more of the preceding element.

5 Scene context analysis

5.1 Extraction of scene context clue

For the occurrence of the scene context clue, we define the left and right parts beside the detected face as scene region of interest for its rich anti-spoofing clues. The choice of region of interest is based on the following observations: (1) The up and bottom parts near a face are hair and neck but not the scene; (2) The region far from a face is not considered as scene context, since the spoofing photo/video scene does not appear in the region. The scene region of interest R in this paper, where spoofing clue may appear, consists of two rectangles, shown as the red rectangles in Fig. 4. The height of the two rectangles is the same as the height of the detected face and the width is $1/4$ width of the face d. To locate the region R, the width of face will be enlarged $d/4$ towards left and right respectively in order to avoid that hair and ears are included in the region of interest.

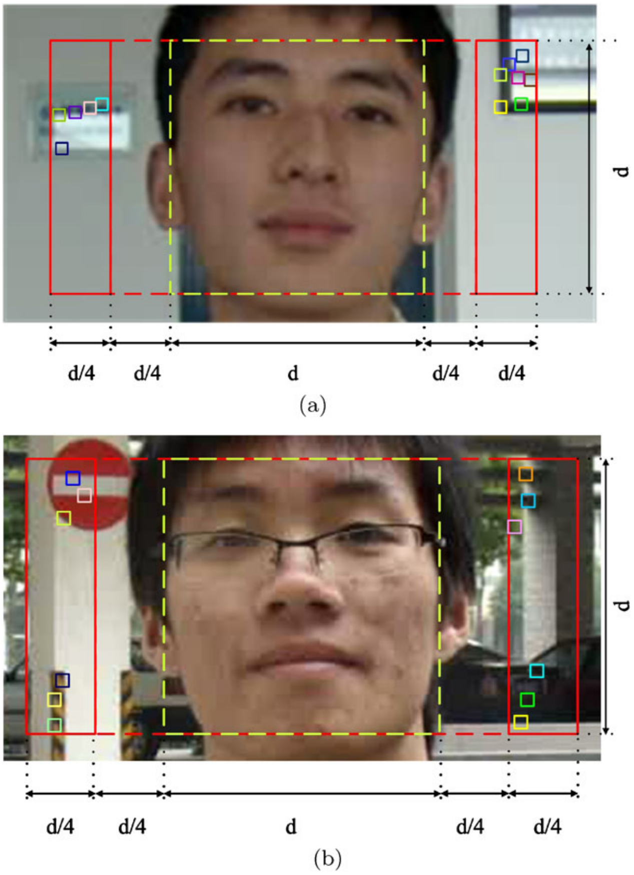


Fig. 4 Examples of scene region of interest and fiducial points extraction. Yellow dashed line rectangles are face regions and red solid line rectangles are scene regions of interest. Fiducial points are labeled by colorful squares

Analysis of scene context has to cope with effect of the noises, such as changes of illumination, for robust scene comparison; otherwise, non-spoofing inputs may be falsely recognized as imposters. This paper extracts a set of key points in the scene for comparison. These points should represent the distinctiveness of the scene as much as possible. We call this kind of key points *fiducial points*. The fiducial point-based representation can also reduce computational cost in the subsequent post-processing steps.

```
Input: an input image $I_i$ and the reference scene image $I_R$
Output: the fiducial point set $P_i$
if a face is detected in $I_i$ then
    Get the scene region of interest $R$ from $I_i$ around the detected face;
    Initialize traversal flag $trav\_flag_{m,n} \leftarrow false$ and fiducial point flag $fiducial\_flag_{m,n} \leftarrow false$, for all
    $m = 0 \ldots \Lambda_M - 1$ and $n = 0 \ldots \Lambda_N - 1$;
    Initialize start coordinate $(x_{m,n} \leftarrow m \times W_\Lambda, y_{m,n} \leftarrow n \times H_\Lambda)$ of each grid $\Lambda_{m,n}$, for all $m = 0 \ldots \Lambda_M - 1$ and
    $n = 0 \ldots \Lambda_N - 1$;
    while $(number(P_i) < N)$ and $(\exists u, v(trav\_flag_{u,v}$ is false$))$ do
        for each grid $\Lambda_{m,n}$ in $R$ do
            if $trav\_flag_{m,n}$ is false then
                while $(x_{m,n}, y_{m,n})$ is in grid $\Lambda_{m,n}$ do
                    if $is\_local\_extrema(x_{m,n}, y_{m,n}, I_i)$ or $is\_local\_extrema(x_{m,n}, y_{m,n}, I_R)$ then
                        Add the point $(x_{m,n}, y_{m,n})$ to the fiducial point set $P_i$;
                        $fiducial\_flag_{m,n} \leftarrow true$;
                        break;
                    $x_{m,n} \leftarrow x_{m,n} + 1$;
                    $y_{m,n} \leftarrow y_{m,n} + 1$;
                if $(x_{m,n}, y_{m,n})$ is out grid $\Lambda_{m,n}$ then
                    $trav\_flag_{m,n} \leftarrow true$;
            if $number(P_i) = N$ then
                exit;
    while $number(P_i) < N$ do
        for each grid $\Lambda_{m,n}$ in $R$ do
            if $fiducial\_flag_{m,n}$ is false then
                Add the point, which has the maximum gradient value in grid $\Lambda_{m,n}$, to the fiducial point set $P_i$;
                if $number(P_i) = N$ then
                    exit;
```

Algorithm 1 Algorithm of fiducial point extraction

To efficiently find the stable fiducial points in the region R, we use local extrema detection in the difference of two nearby scales of the image [24]. Scale-space extrema are scale-invariant, highly distinctive, and robust to illumination change. They have been successfully applied in image matching, object recognition, and image retrieval. The difference-of-Gaussian function, $D_I(x, y, \sigma)$ [24], is employed for scale space, which is computed by the difference of two nearby scales separated by a constant multiplicative factor k:

$$D_I(x, y, \sigma) = \big(G(x, y, k\sigma) - G(x, y, \sigma)\big) * I(x, y), \tag{9}$$

where $*$ is the convolution operation in x and y, and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x+y)/2\sigma^2}. \tag{10}$$

The number of scale-space extrema extracted are different from frame to frame. To get robust result and reduce computing time, we fix the number of fiducial points as a constant N. To avoid that fiducial points concentrate on an objects in the scene region of interest R, we divide the region R into a rectangle grid matrix with Λ_N rows and Λ_M columns to extract fiducial points distributed equably. The width W_Λ and height H_Λ of each rectangle grid are $\frac{d}{2\Lambda_M}$ and $\frac{d}{\Lambda_N}$ respectively. Scale-space extrema are searched in grids by turns. At most one fiducial point is selected for a grid in one turn. If all grids are traveled and the number of fiducial points is less than N, the maximum gradient point in a grid which does not have scale-space extreme will be selected into fiducial point set. To ensure that N fiducial points are extracted, Λ_N and Λ_M are set to satisfy $\Lambda_N \times \Lambda_M \geq N$. The algorithm of fiducial point extraction is detailed in Algorithm 1. Here we merge the

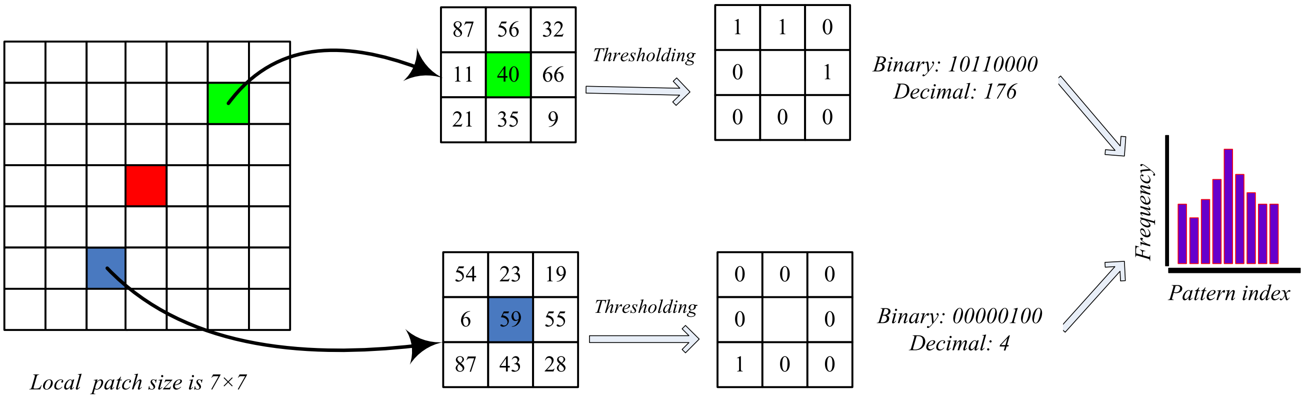


Fig. 5 Illustration of the LBP descriptor for a fiducial point. Local patch size is 7 × 7 pixels for example. The fiducial point in the center of local patch is indicated by red point. The green and blue points are two examples in the patch for $LBP_{8,1}$ calculation

local extrema of the input image and those of the reference scene image in the region R for candidates of fiducial points. The function $is\_local\_extrema(x, y, I)$ determines whether the point (x, y) is the extremum among 26 neighbors in $3 \times 3 \times 3$ cube of (x, y) at current scale σ and its adjacent scales [24]. Samples of extracted fiducial points are shown in Fig. 4, labeled by colorful squares, where $N = 12$.

5.2 Scene context clue matching

If the fiducial point set P_i is ready, we need to describe the local texture characteristics at each fiducial point in both an input image I_i and a reference scene image I_R for scene matching. For this purpose, we combine the local feature descriptor Local Binary Pattern (LBP) [25] into a proposed liveness measure for its powerful texture descriptive capability [26, 27] and computational simplicity. The LBP descriptor characterizes local texture of points with binary patterns. Figure 5 illustrates that how LBP serves as the descriptor of a fiducial point. The local neighborhood of a pixel is thresholded with its intensity value, and then a binary number vector (i.e. binary pattern) is generated for the pixel. A binary vector for pixel (x, y) as $B(x, y) = \langle b_{P-1}, \ldots, b_1, b_0\rangle$. We use the same notation $LBP_{P,R}$ as reference [25], where R is the radius of the circle to be sampled and P is the number of sampling points. It is common to illustrate $LBP_{P,R}$ in decimal form via binomial weighting:

$$LBP_{P,R}(x, y) = \sum_{i=1}^{P-1} b_i 2^i. \tag{11}$$

For a local patch centered at a fiducial point (x, y), each pixel in the local patch has a binary pattern. The frequency histogram $H_{LBP}(I, x, y)$ of these local binary patterns over the local patch depicts the texture distribution around the fiducial point (x, y).

Given a face image sequence $\mathbb{S} = \{I_i\}_{i=1}^{T}$, the reference scene image I_R, and the extracted fiducial point sets $\{P_i\}_{i=1}^{T}$, we define the scene matching score Φ for the image sequence $\mathbb{S}$ as follows,

$$\Phi(\mathbb{S}, I_R) = \frac{1}{T}\sum_{i=1}^{T}\left(\frac{1}{N}\sum_{(x,y)\in P_i} D\big(H_{LBP}(I_i, x, y), H_{LBP}(I_R, x, y)\big)\right), \tag{12}$$

where $H_{LBP}(I, x, y)$ is the distribution histogram of LBP for the local image patch at the fiducial point (x, y) in the image I. $D(\bullet)$ is the LBP histogram distance. This paper chooses χ^2 distance for $D(\bullet)$. N is the number of fiducial points for each frame. Therefore, whether the scene of image sequence $\mathbb{S}$ matches the reference scene can be determined by comparing the scene matching score with a predefined threshold η.

$$isMatch(\mathbb{S}, I_{\text{ref}}) = \begin{cases} \text{yes}, & \Phi(\mathbb{S}, I_R) \geq \eta, \\ \text{no}, & \Phi(\mathbb{S}, I_R) < \eta. \end{cases} \tag{13}$$

6 Experiments

To evaluate the performance of our approach against the photograph spoofing and video spoofing, two databases are built: *photo-imposter video database* and *live face video database*. The first one is mainly for evaluation of capability of anti-spoofing of photographs, the second one is used to simulate video imposters of input.

6.1 Database

The databases to evaluate our approach are collected by Logitech Pro5000, a generic webcamera. Each video clip is captured with 30 fps and the size is 320 × 240 pixels.

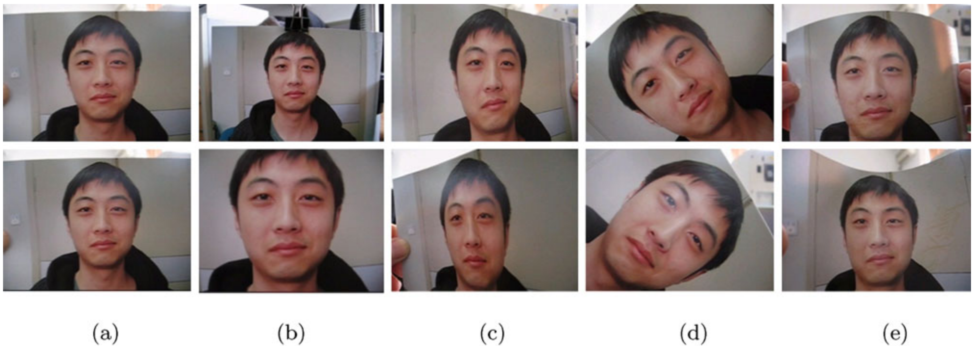


Fig. 6 Samples from the photo-imposter video database. Each column with two images illustrates one sort of photo attacks. (**a**) Keep photo still. (**b**) Move vertically, horizontally, backward and forward. (**c**) Rotate in depth. (**d**) Rotate in plane. (**e**) Bend inward and outward

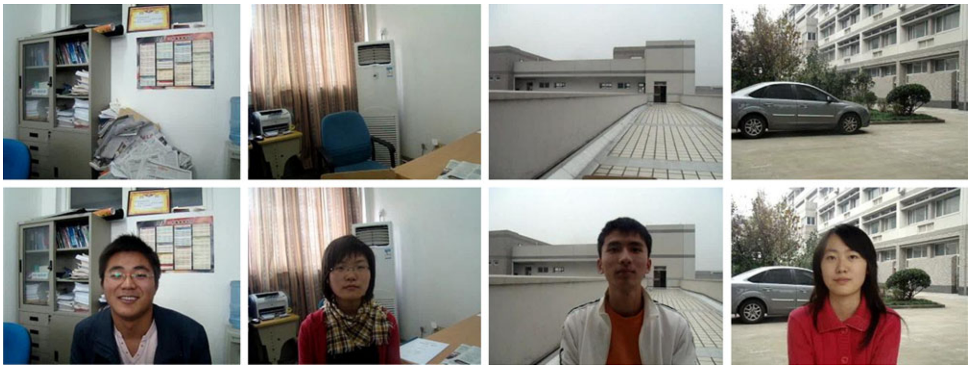


Fig. 7 Samples from the live face video database. The top row is four scene reference images, two are indoor scenes and the other two are outdoor scenes. The bottom row is live faces in video

The *photo-imposter video database* consisting of 100 video clips from 20 persons is used to test capabilities against photo imposters. A high-quality photo in front viewpoint is taken for each person, then five categories of photo-attacks are simulated before the camera: (1) keep the photo still; (2) move the photo vertically, horizontally, back and front; (3) rotate the photo in depth along the vertical axis; (4) rotate the photo in plane; (5) bend the photo inward and outward along the central line. For each attack, one video clip is captured with length of about 10 to 15 seconds. Some samples are shown in Fig. 6.

The *live face video database* contains 196 clips for 14 individuals. There are 2 indoor and 5 outdoor scenes. Each scene has a scene reference image. Each individual appears before the camera twice and stays there about 5 seconds for liveness verification. Some samples are shown in Fig. 7.

6.2 Performance of eyeblink detection for photo-spoofing

To compute *eye closity*, we need to train a series of efficient weak classifiers. A total of 1,016 labeled images of close eyes (positive samples) and 1,200 images of open eyes (negative samples) are used in the training stage. We do not differentiate left and right eyes. All the eye samples are scaled to a base resolution of 24×24 pixels. Eventually 50 weak classifiers are selected for computing the *eye closity* (6).

In both the testing stage and the training stage, the center of left and right eyes is automatically localized for each frame by the eye localization system using cascaded Adaboost classifiers. Each pattern of eye state variation $\alpha\beta?\gamma+\beta?\alpha$ is accounted as one blink for this eye.

Five categories of photo attacks are simulated using the video clips in the photo-imposter video database. The clip of input is recognized as an imposter if no blink of single eye is detected in the clip. Table 2 shows the results with various temporal window size in the eyeblink detection. The number in the table shows how many clips failed during the attack test. It can be seen that performance with different window size is very close. There are only 1–2 photo-imposter clips failed to detect out of 100 clips.

6.3 Performance of scene context for video-spoofing

Videos in the *live face video database* are used to attack each other among different reference scenes for evaluating the anti-video spoofing ability of the scene context clue. The false accept rate (FAR) is computed by video imposter attacks between different scenes (total $7 \times 6 \times 28 = 1176$ matches) and the false reject rate (FRR) is evaluated by matching among clips of the same scene (total $7 \times 28 = 196$ matches). The equal error rate (EER, when FAR=FRR) is employed for evaluation.

We use $LBP_{8,1}$ in the experiment. The size of local patch for LBP pattern histogram is optimized to 17×17. The number of objects in scene region of interest is roughly 3–5. Considering nearly 3 fiducial points for each object, the number of fiducial points N is set to 12 in our experiment. The rows

Table 2 Results of photograph attacks using *photo-imposter video database*, which includes 20 subjects, five categories of photo attacks for each. The number shown in the table means the failed clip number

Category of attacks	$W = 0$	$W = 1$	$W = 2$	$W = 3$	$W = 4$
Keep photo still	0	0	1	0	0
Move vert., hor., back and front	0	0	1	0	0
Rotate in depth	0	0	0	0	0
Rotate in plane	1	1	0	0	0
Bend inward and outward	0	0	0	1	0
Total	1	1	2	1	0

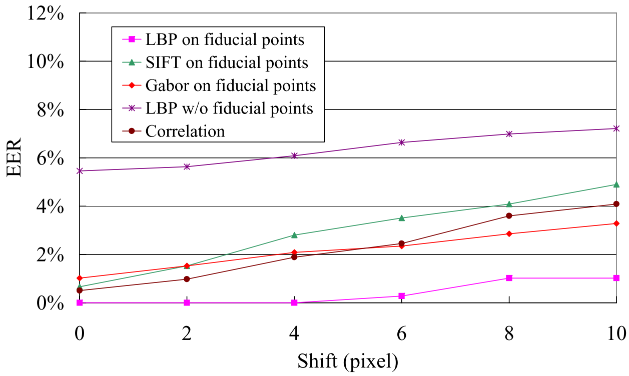


Fig. 8 Performance of LBP descriptor on fiducial points and comparison with other methods

of the grid matrix is set to $\Lambda_N = 6$ and the columns of the grid matrix is set to $\Lambda_M = 2$. Thus, 6×2 grids ensure 12 fiducial points in the region R.

To make a comparison with other local descriptors, we implement two frequently used descriptors: Gabor feature [28] and SIFT feature [24]. Gabor descriptor has 40 complex Gabor wavelet coefficients, which are yielded by 8 orientations and 5 frequencies parameters for a fiducial point. Phase similarity is used as the distance between two Gabor descriptors [29]. For SIFT descriptor, we use 4×4 histograms, each histogram has 8 orientation bins, computed from a 16×16 sample array around one fiducial point. This leads to a SIFT descriptor vector with 128 elements ($4 \times 4 \times 8$) [24].

In addition, two non-fiducial-point approaches are implemented for comparison. The first one is Pearson product-moment correlation, which is a well-known correlation coefficient in statistics. The second one is LBP descriptor over the entire region of interest, i.e., LBP pattern of each pixel in the region is counted into the histogram.

Furthermore, considering that slight movements of the stationary camera may occur when the face recognition is working, we simulate the slight camera movements by slight shifts of video clips in pixels vertically and horizontally, in order to test the capability against slight shifts.

Figure 8 demonstrates the experimental results. It can see that LBP with fiducial points significantly outperforms other methods. When slight shift happens, our approach achieves very low FAR and FRR simultaneously. Compared with LBP without fiducial points, extraction of fiducial points greatly improves the performance.

Table 3 Blink detection rates for a clip with varying window size ($W = \{0, 1, 2, 3, 4\}$)

Window size	$W = 0$	$W = 1$	$W = 2$	$W = 3$	$W = 4$
Detection rate	100%	100%	100%	100%	100%

6.4 Combination of eyeblink and scene context for liveness detection

The *live face video database* is used to evaluate capabilities of blink and scene context clues to detect a live face. The parameters for the blinking detection are same as Sect. 6.2 and those for scene context analysis are same as Sect. 6.3.

Using the eyeblink clue only, the clip is considered as a live face if any blinks are detected in the clip. The liveness detection result using different temporal window size is shown in Table 3. The result shows that all the live faces could be detected.

Using the scene context clue only, the detection rate of a live face is evaluated by matching among clips of the same scene (total $7 \times 28 = 196$ matches). The distribution of matching score for live faces and spoofing videos is shown in Fig. 9. If the threshold η in (13) is set to 0.65, only one live face is rejected falsely by the scene context clue.

If we combine the clues of eyeblinks and scene context as shown in Fig. 1, based on the results above, the liveness detection rate after combination is 99.5% (195 out of 196).

The system is implemented on the hardware platform of Intel Core2 Duo 2.8 GHz, 2 GB RAM. The average computational time for each frame is listed in Table 4. Benefiting from dual-core, two liveness clues can be computed parallelly. The whole system can achieve real-time processing speed of nearly 20 fps, which satisfies many practical applications.

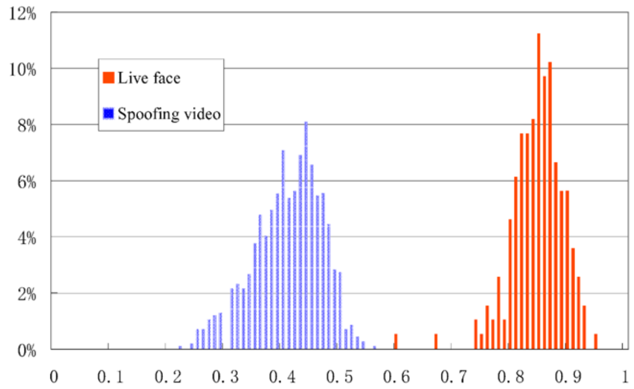


Fig. 9 Distribution of matching scores Φ for live faces and spoofing videos

Table 4 Average computational time of each stage per frame

		Time (ms)
Face detection & eye localization		22
Blink clue	Eye closity	10
	Conditional model inference	12
Scene context clue	Fiducial point extraction	15
	Matching	3
Total parallel processing time		49

7 Discussions

For the scene context clue, although the attack using the system background is theoretically possible, however, it requires strong consistencies in scale, viewpoint, and illumination. It may be practically impossible to match the region of interest in a spoofing video to the one in the reference scene by placing a video clip in front of the camera, even the scene of the spoofing video is the same as the reference scene. The coordinates of objects in the reference scene are constant from the viewpoint of the stationary camera, while the coordinates of objects in the spoofing video scene are variational for video scaling and moving in front of the camera. Thus, the appearance texture in region of interest from the video and the reference scene are hardly identical from the viewpoint of the camera. However, if the appearances of the regions of interest both in the video and reference scenes are very simple, such as filling with white wall, it would be hard for the scene context clue to distinguish.

8 Conclusions

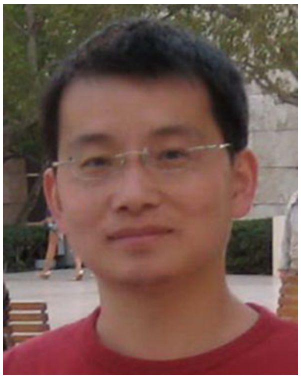
This paper presents a real-time face liveness detection system using a monocular camera in non-intrusive manner. It assumes the camera is stationary. We employ combination of the clues of eyeblinks and scene context for liveness detection. The scene context clue is to detect video imposters that the eyeblink clue can not detect and the eyeblink clue is to detect trimmed photograph imposters that the scene context clue can not detect. The experimental results show that the system has reasonable ability to resist common spoofing methods and it is robust to camera's tiny shifts in practical applications.

Acknowledgements The authors would like to thank the comments and suggestions from the anonymous reviewers. The students' participation in the data collection is greatly appreciated. This work was partly supported by NSF of China (Nos. 60525202, 60533040), and National 863 High-Tech Programme (2008AA01Z149).

References

1. Jain, A., Bolle, R., & Pankanti, S. (1999). *Biometrics: personal identification in networked society*. Berlin: Springer.
2. Schuckers, S. (2002). *Spoofing and anti-spoofing measures, information security technical report* (Vol. 7, pp. 56–62). Amsterdam: Elsevier.
3. Bigun, J., Fronthaler, H., & Kollreider, K. (2004). Assuring liveness in biometric identity authentication by real-time face tracking. In: *IEEE international conference on computational intelligence for homeland security and personal safety* (*CIHSPS'04*) (pp. 104–111), 21–22 July 2004.
4. Parthasaradhi, S., Derakhshani, R., Hornak, L., & Schuckers, S. (2005). Time-series detection of perspiration as a liveness test in fingerprint devices. *IEEE Trans. Syst. Man Cybern.*, *35*(3), 335–343.
5. Antonelli, A., Cappelli, R., Maio, D., & Maltoni, D. (2006). Fake finger detection by skin distortion analysis. *IEEE Trans. Inf. Forensics Secur.*, *1*(3), 360–373.
6. Zhao, W., Chellappa, R., Phillips, J., & Rosenfeld, A. (2003). Face recognition: a literature survey. *ACM Comput. Surv.*, *35*, 399–458.
7. Frischholz, R. W., & Dieckmann, U. (2000). BioID: a multimodal biometric identification system. *IEEE Comput.*, *33*(2), 64–68.
8. Kollreider, K., Fronthaler, H., Faraj, M. I., & Bigun, J. (2007). Real-time face detection and motion analysis with application in liveness assessment. *IEEE Trans. Inf. Forensics Secur.*, *2*(3), 548–558.
9. Frischholz, R. W., & Werner, A. (2003). Avoiding replay-attacks in a face recognition system using head-pose estimation. In *IEEE international workshop on analysis and modeling of faces and gestures* (*AMFG'03*) (pp. 234–235).
10. Choudhury, T., Clarkson, B., Jebara, T., & Pentland, A. (1999). Multimodal person recognition using unconstrained audio and video. In *Proc. 2nd int. conf. audio-video based person authentication* (*AVBPA'99*) (pp. 176–181), Washington, DC, 1999.
11. Kollreider, K., Fronthaler, H., & Bigun, J. (2009). Non-intrusive liveness detection by face images. *Image Vis. Comput.*, *27*(3), 233–244.
12. Pan, G., Sun, L., Wu, Z. H., & Lao, S. H. (2007). Eyeblink-based Anti-Spoofing in Face Recognition from a Generic Webcamera. In *The 11th IEEE international conference on computer vision* (*ICCV'07*), Rio de Janeiro, Brazil, 14–20 October 2007.
13. Li, J. W., Wang, Y. H., Tan, T. N., & Jain, A. K. (2004). Live face detection based on the analysis of Fourier spectra, biometric technology for human identification. *SPIE*, *5404*, 296–303.
14. Socolinsky, D. A., Selinger, A., & Neuheisel, J. D. (2003). Face recognition with visible and thermal infrared imagery. *Comput. Vis. Image Underst.*, *91*(1–2), 72–114.
15. Chetty, G., & Wagner, M. (2006). Multi-level liveness verification for face-voice biometric authentication. In *Biometrics symposium 2006*, Baltimore, Maryland, 19–21 Sep. 2006.

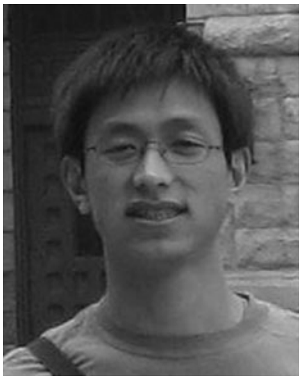
16. Kollreider, K., Fronthaler, H., & Bigun, J. (2008). Verifying liveness by multiple experts in face biometrics, In *IEEE computer society conference on computer vision and pattern recognition workshop on biometrics* (pp. 1–6).
17. Karson, C. (1983). Spontaneous eye-blink rates and dopaminergic systems. *Brain*, *106*, 643–653.
18. Tsubota, K. (1998). Tear dynamics and dry eye. *Prog. Retin. Eye Res.*, *17*(4), 565–596.
19. Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields: probabilistic models for segmenting and labeling sequence data. In *Proc. 18th int. conf. machine learning* (pp. 282–289).
20. Li, S. Z. (2001). *Markov random field modeling in image analysis*. Berlin: Springer.
21. Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, *77*(2), 257–286.
22. Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, *55*(1), 119–139.
23. Viola, P., & Jones, M. J. (2004). Robust real-time face detection. *Int. J. Comput. Vis.*, *57*(2), 137–154.
24. Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.*, *60*(2), 91–110.
25. Ojala, T., Pietikäinen, M., & Mäenpää, T. (2002). Multiresolution gray scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, *24*(7), 971–987.
26. Heikkilä, M., & Pietikäinen, M. (2006). A texture-based method for modeling the background and detecting moving object. *IEEE Trans. Pattern Anal. Mach. Intell.*, *28*(4), 657–662.
27. Ahonen, T., Hadid, A., & Pietikäinen, M. (2006). Face description with local binary patterns: application to face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, *28*(12), 2037–2041.
28. Jones, J., & Palmer, L. (2006). An evaluation of the two dimensional Gabor filter model of simple receptive fields in cat striate cortex. *J. Neurophys.*, *58*(6), 1233–1258.
29. Wiskott, L., Fellous, J. M., Kruger, N., & Malsburg, C. (1997). Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, *19*(7), 775–779.



Gang Pan Gang Pan received the B.S. and Ph.D. degrees in computer science from Zhejiang University, Hangzhou, China, in 1998 and 2004, respectively. Since 2004, he has been with Zhejiang University, where he has been an associate professor of computer science since 2006. His research interests include pervasive computing, computer vision, and pattern recognition. He has served as a PC member for numerous international conferences, such as ICCV, CVPR, UIC, and as a reviewer for more than ten journals, including TPAMI, TIP, TVCG, TSMC-B, and PUC Journal.

Lin Sun Lin Sun received the B.S. degree in communication engineering in 2001 and M.S. degree in Computer science in 2004 from East China University of Science and Technology, China. He is currently a Ph.D. candidate at the College of Computer Science and Technology, Zhejiang University, China. His research interests include biometrics, pattern recognition, and computer vision.

Zhaohui Wu Zhaohui Wu received the Ph.D. degree in computer science from Zhejiang University, Hangzhou, China, in 1993. From 1991 to 1993, he was with the German Research Center for Artificial Intelligence (DFKI) as a joint Ph.D. student in the area of knowledge representation and expert system. Currently he is a Professor of computer science with Zhejiang University and the Director of the Institute of Computer System and Architecture. He has authored four books and more than 100 refereed papers. His major interests include intelligent systems, semantic grid, and ubiquitous embedded systems. He is on the editorial boards of several journals and has served as PC member for various international conferences.



Yueming Wang Yueming Wang received the Ph.D. degree from Zhejiang University, P.R. China, in 2007. From 2007 to 2010, he was a postdoctoral fellow in the Department of Information Engineering, the Chinese University of Hong Kong. He is now a faculty member in the Qiushi Academy for Advanced Studies, Zhejiang University, P.R. China. He serves as the corresponding author. His research interests include 3D face processing and recognition, object detection, and statistical pattern recognition.