Multimed Tools Appl
https://doi.org/10.1007/s11042-018-5721-2

# Computer-assisted yoga training system

**Hua-Tsung Chen**[1] · **Yu-Zhen He**[2] · **Chun-Chieh Hsu**[2]

Received: 5 October 2016 / Revised: 1 May 2017 / Accepted: 24 January 2018


**Abstract** Self-training is essential in sports exercise. However, without the instruction of a coach, a practitioner may progress to a limited extent. Improper postures may even cause serious harm to muscles and ligaments of the body. Hence, the development of computer-assisted self-training systems for sports exercise is a recently emerging research topic. In this paper, we propose a yoga self-training system, which aims at instructing the practitioner to perform yoga poses correctly, assisting in rectifying poor postures, and preventing injury. Integrating computer vision techniques, the proposed system analyzes the practitioner's posture from both front and side views by extracting the body contour, skeleton, dominant axes, and feature points. Then, based on the domain knowledge of yoga training, visualized instructions for posture rectification are presented so that the practitioner can easily understand how to adjust his/her posture. Experiments on twelve yoga poses performed by different practitioners validate the feasibility of the proposed system in yoga training.

**Keywords** Sports training · Multimedia system · Computer vision · Image processing · Yoga · Posture analysis · Body skeleton

## 1 Introduction

### 1.1 Motivation

For most sports players/practitioners, it is essential to spend time exercising on their own, in addition to the regular training courses given by a coach or instructor. However, if not under the instruction of a coach/instructor, a sports player/practitioner may make progress only to a limited extent, and even may get injured during self-training due to improper postures or training ways.

✉ Hua-Tsung Chen
huatchen@fcu.edu.tw

[1] Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan

[2] Department of Computer Science, National Chiao Tung University, Hsinchu 30010, Taiwan

Published online: 03 February 2018

Thus, a great number of sports players/practitioners expect the development of computer-assisted training systems to assist them in improving their performance and protecting them from injury. Accordingly, the topic of computer-assisted sport/exercise training system is attracting more and more research attention. Numerous automatic or semi-automatic training systems have been developed for many sports or exercises, including golf [8, 12, 17, 18, 22], billiards (or snooker) [11, 14, 28, 29], yoga [15, 19, 25, 26, 31], rugby [9, 10, 21], table tennis [1, 27], and so on.

Since the ancient Indian art, yoga, not only promotes physical health but also helps to purge the body, mind, and soul, it gains growing popularity nowadays. When practicing yoga, the practitioner has to align body positions in a special way. If the practitioner does not perform a yoga pose correctly, improper postures may cause serious harm to muscles and ligaments of the body. Thus, we are motivated to develop a computer vision-based yoga self-training system for assisting practitioners in exercising by themselves. Postural instructions and feedback can be provided automatically to help practitioners to adjust their poses.

### 1.2 Related works

Due to the low cost of hardware devices and the rapid advance of computing power, more and more sports players/practitioners crave to improve their performance with the assistance of computer technology. Lots of research efforts have been devoted into this field, of which two hot topics are (i) tactic analysis [2–5, 16, 33, 34] (to cite a few) and (ii) computer-assisted sports training [1, 8–12, 14, 15, 17–19, 21, 22, 25–29, 31].

To adapt the operational policy timely during a game, it is essential to obtain the tactic patterns, player actions, and statistical data in a short time. Thus, sports professionals thirst for automatic/semi-automatic systems for the tasks of tactics analysis, game annotation, match recording, and statistics collection, which are time-consuming and labor-intensive manual efforts in the past years. Chen et al. [2–4] propose various ball trajectory-based applications, such as pitching evaluation in baseball, set type recognition in volleyball, and shooting location estimation in basketball. To recognize tactical patterns in soccer video, Zhu et al. [34] analyze the temporal-spatial interaction among the ball and players to construct a tactic representation, *aggregate trajectory*, based on multiple trajectories. There are also research works focusing on tactic analysis in basketball video [5, 16], which perform camera calibration to obtain 3D-to-2D transformation, map player trajectories to the real-world court model, and detect the *wide-open* event or recognize offensive tactic patterns.

For the majority of sports players/practitioners, it is essential to spend time exercising on their own, in addition to the regular training courses given by a coach or instructor. However, players/practitioners may get injured during self-training due to improper postures or training ways. Thus, computer-assisted sports training systems, which can act as tutors to give instructions, are highly demanded. Using a multi-camera based high speed motion capture system, Kelly et al. [17] present visualisation and analysis tools to identify and eliminate faults in a golfer's swing mechanics. The movements of a golfer's swing are aligned and compared with higher-skilled experts so as to give instructions on how the golfer should adjust the swing mechanics. Chen et al. [8] propose a vision-based golf training system, which detects postural faults by using three automatically generated feature lines to evaluate whether the golfer's body is properly aligned. There are some other golf training systems utilizing wearable devices to measure the wrist rotation [12], placing inertial measurement units (IMU) on a golf club to measure its acceleration and angular velocity [18], or using an optical motion capture system to obtain the swing movements [22].

There are existing works in computer-assisted yoga training. Rector et al. [26] develop an exercise game, Eyes-Free Yoga, which teaches six yoga poses for the people who are blind or low vision. Based on skeletal tracking with the Microsoft Kinect, auditory feedback can be provided, enabling people to practice yoga independently. Patil et al. [25] propose the "yoga tutor" project, which uses SURF (Speeded Up Robust Features) to detect and visualize the postural difference between a practitioner and an expert. However, only the contour information captured from one viewing-direction seems insufficient to describe and compare the postures appropriately. Luo et al. [19] propose a yoga training system based on motion replication technique (MoRep). The InterfaceSuit, comprising Inertial Measurement Units (IMUs) and tactors, is able to precisely capture the body motions, but may influence the practitioner's exercise. Wu et al. [31] develop a yoga expert system, which instructs training techniques based on images and text. However, the practitioner has no idea whether he/she is performing a yoga pose correctly since no posture analysis is conducted. Hsieh et al. [15] develop a distance yoga learning system based on computer vision techniques. By matching the distance transformation between the user silhouette and a standard yoga posture, the system can give a score for user posture evaluation.

There is growing interesting in utilizing *virtual environments* (VE) for training in ball sports [1, 9, 10, 21, 27] (to cite a few). A VE is built with a large collection of technologies (computer vision, stereoscopy, motion capture, tracking, photogrammetry, haptics, etc.) and hardware devices (data projector, head-mounted display, game pad, motion sensor, camera, etc.), enabling people to interact with computer-rendered vivid scenes using their skills and senses. For a more thorough discussion of VE-based training in ball sports, please refer to the comprehensive survey by Helen et al. [20].

In summary, some existing systems on sports training can be easily used by general users, such as [15, 25, 31]. However, they do not actually analyze the user's posture, and accordingly, no instructions for posture adjustment can be given, either. As for the sensor-based and VE-based systems, they can well capture the user's motion, analyze his/her posture, and provide real time response in a vivid sports scene. However, for the general users who just want to practice at home, it is not feasible to build up a studio-like environment with complex hardware devices. Therefore, there is still the need to develop sports training systems which can analyze user's posture and then give instructions merely by means of software operating on existing hardware, such as PC, notebook, webcam, etc.

### 1.3 Contribution

With the foregoing motivation and limitations of the existing works, we develop a computer vision-based yoga training system, termed *Y-system*, to analyze the postures of a practitioner and assist in rectifying incorrect postures. Enhanced from a preliminary version of this work in [6], the proposed system extracts more visual features and is capable of analyzing up to twelve yoga poses, including: (1) *Tree*, (2) *Full Boat*, (3) *Downward-Facing Dog*, (4) *Extended Hand-to-Big-Toe*, (5) *Chair*, (6) *Warrior I*, (7) *Warrior II*, (8) *Warrior III*, (9) *Cobra*, (10) *Plank*, (11) *Side Plank*, and (12) *Lord of the Dance*, as shown in Fig. 1.[1] The schematic diagram of the proposed Y-system is illustrated in Fig. 2. Yoga comprises various poses, and each has its respective training emphasis. Thus, the Y-system first recognizes what pose the practitioner is currently performing by our previous work [7], as shown in Fig. 2a. (In this illustration, the

[1] For more details about each yoga pose, please refer to [32].

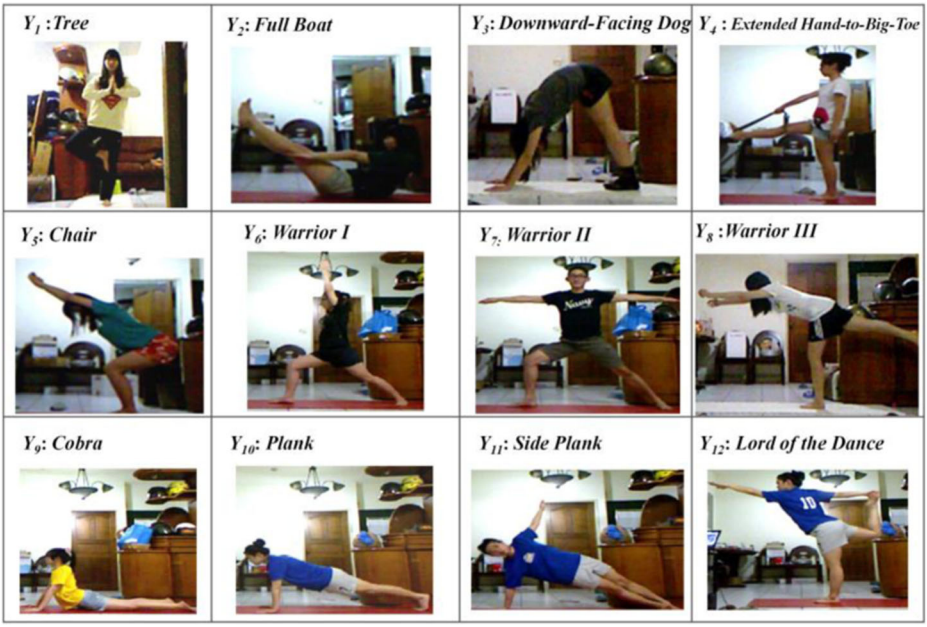



**Fig. 1** Twelve yoga poses in the proposed yoga training system

practitioner is performing the *Plank* pose.) Topological skeletons are generated from the practitioner's body maps of front and side views, as shown in Fig. 2b. Then, the Y-system extracts postural features including dominant axes, skeleton-based feature points, and contour-based feature points, as shown in Fig. 2c. Finally, the Y-system involves yoga training knowledge and presents visualized instructions for posture rectification, as shown in Fig. 2d, enabling the practitioner to easily perceive how to adjust his/her posture and preventing injury caused by an improper posture.

Compared with existing works, the main contributions of this paper are summarized as follows.

- We develop an accessible and practical yoga training system for the people who would like to practice yoga by him/herself. Acting as a yoga tutor, the Y-system can analyze the practitioner's posture, detect improper alignment of body positions, and give instructions.
- We design explicit feature extraction and posture judgment rules directed against the training emphases of up to twelve yoga poses. To the best of our knowledge, the proposed Y-system can analyze the most yoga poses among the existing vision-based yoga training systems.
- The Y-system is capable of analyzing the practitioner's posture from front and side views, merely by means of software operating on existing hardware, so the Y-system can be easily used by general users, in no need of a studio-like environment nor complex hardware devices.

The rest of this paper is organized as follows. Section 2 elaborates the detailed processing steps of postural feature extraction, including topological skeleton generation, dominant axis estimation, and feature point detection. Then, Section 3 presents explicit posture judgment

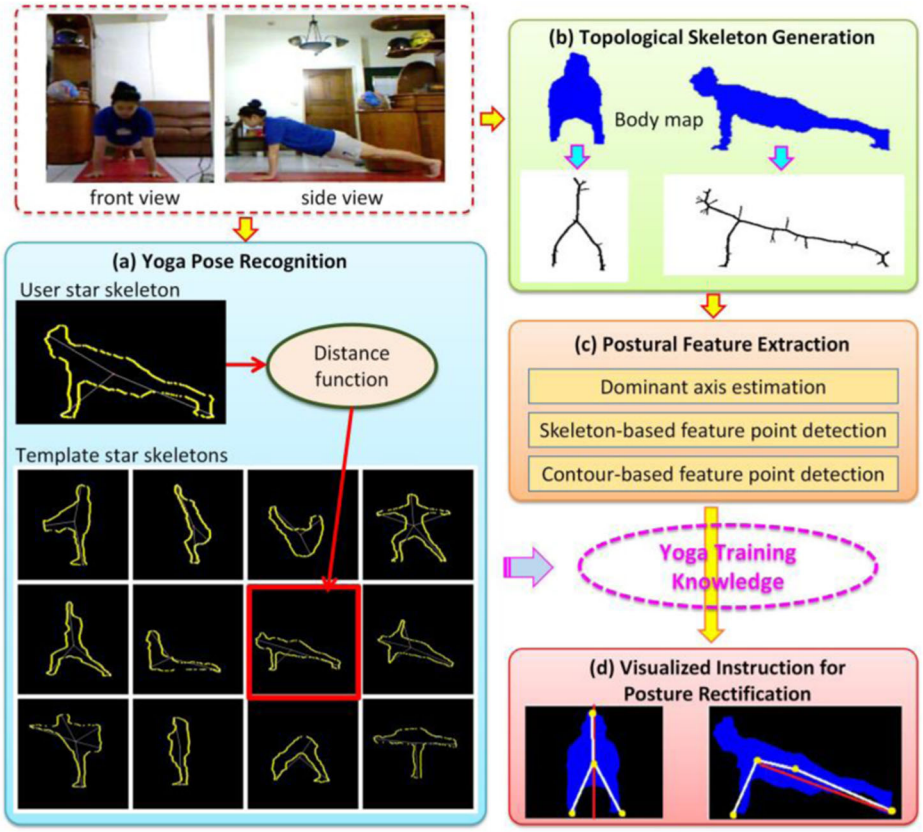



**Fig. 2** Schematic diagram of the proposed yoga training system

rules for the yoga poses. Experimental results and discussions are given in Section 4. Finally, Section 5 concludes this paper.

## 2 Postural feature extraction

### 2.1 Topological skeleton generation

The topological skeleton is a thin version of a shape that is equidistant to its boundaries. The processing steps of skeleton generation are illustrated in Fig. 3. The original RGB frame and practitioner body map are presented in Fig. 3a and b, respectively. A distance map (Fig. 3c) is produced by applying distance transform to the body map. (Distance transform is an operator normally only applied to binary images, resulting in a gray-level image that shows the distance to the closest boundary from each point.) Then the skeleton map is generated in the way that four line masks, as given in Fig. 4, are run individually through the distance map. Let $R_1$, $R_2$, $R_3$, and $R_4$ denote the responses of the masks, as shown in Fig. 5. For a certain point in the distance map, let $R_{max} = \max(|R_1|, |R_2|, |R_3|, |R_4|)$, and if $R_{max}$ is greater than a threshold, the pixel value at the corresponding point in the skeleton map to be produced is set to 1; otherwise, it is set to 0.

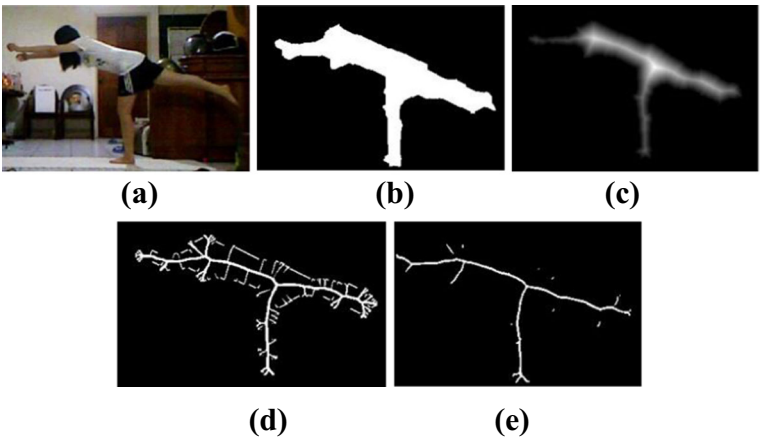



**Fig. 3** Illustration of topological skeleton generation. **a** Original RGB frame. **b** Practitioner body map. **c** Distance map. **d** Topological skeleton. **e** Refined skeleton map

One can see that the skeleton map presented in Fig. 3d can describe the posture appropriately. Finally, we refine the rough skeleton map by extracting the parts of the main connected components and eliminating the thin branches close to the contour, as shown in Fig. 3e.

## 2.2 Dominant axis estimation

To present the general distribution of the body and limbs, we estimate the dominant axes of the practitioner's body by applying Hough transform to the topological skeleton map. To discard the ones out of interest, we define the constraints on the range of the angle $\theta$ on the Hough accumulator plane ($r$, $\theta$), where $r$ represents the distance between the extracted line and the origin, and $\theta$ is the angle of the vector from the origin to the closest point on the line. The lines with the maximal values in the ranges of $\theta \in[-\pi/8, \pi/8]$ and $\theta \in[\pi/4, 3\pi/4]$ are respectively extracted as the dominant vertical and horizontal axes, termed $V_X$ and $H_X$, as shown in Fig. 6.

## 2.3 Skeleton and contour-based feature point detection

In addition to the dominant axes, feature points are also indispensable to describe the practitioner's posture in detail. Thus, we first detect the corners of the topological skeleton as feature points. As shown in Fig. 7, Harris corner detection [13] is applied to build a *corner response map*, on which the local maxima are extracted as our *skeleton-based feature points* (abbr. S-points).

| | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| -1 | -1 | -1 | -1 | -1 | 0 | -1 | -1 | 0 | 2 | -1 | 0 | 2 | 0 | -1 | 2 | 0 | -1 | -1 | 0 |
| 0 | 0 | 0 | 0 | 0 | -1 | -1 | 0 | 2 | 0 | -1 | 0 | 2 | 0 | -1 | 0 | 2 | 0 | -1 | -1 |
| 2 | 2 | 2 | 2 | 2 | -1 | 0 | 2 | 0 | -1 | -1 | 0 | 2 | 0 | -1 | -1 | 0 | 2 | 0 | -1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | -1 | -1 | -1 | 0 | 2 | 0 | -1 | -1 | -1 | 0 | 2 | 0 |
| -1 | -1 | -1 | -1 | -1 | 2 | 0 | -1 | -1 | 0 | -1 | 0 | 2 | 0 | -1 | 0 | -1 | -1 | 0 | 2 |
| 0° | | | | | 45° | | | | | 90° | | | | | 135° | | | | |

**Fig. 4** Line masks for topological skeleton generation

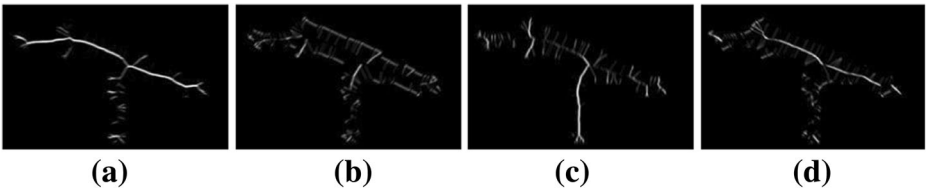



**Fig. 5** Responses of the distance map in Fig. 3b convolved with the line masks in Fig. 4a $R_1$. b $R_2$. c $R_3$. d $R_4$

On the other hand, we also extract the extreme points on the body contour as feature points. As illustrated in Fig. 8, the top and bottom points on the contour, termed $T$ and $B$, and the farthest points (from the body centroid $C$) on the left and right half contours, termed $L$ and $R$, are extracted as our *contour-based feature points* (abbr. C-points). In the upcoming section, different feature points are chosen for analyzing different yoga poses.

## 3 Visualized instructions for posture rectification

Our proposed Y-system is capable of assisting the practitioner in practicing twelve yoga poses, including: (1) *Tree*, (2) *Full Boat*, (3) *Downward-Facing Dog*, (4) *Extended Hand-to-Big-Toe*, (5) *Chair*, (6) *Warrior I*, (7) *Warrior II*, (8) *Warrior III*, (9) *Cobra*, (10) *Plank*, (11) *Side Plank*, and (12) *Lord of the Dance*, as shown in Fig. 1. Since the training emphases of different yoga poses are widely divergent (please refer to [32]), it may not be feasible to use a single model or algorithm for posture description and visualized instruction. In the following, we design explicit posture description models for the 12 yoga poses based on their respective training emphases.

(1) ***Tree*** is a basic pose in yoga, as shown in Fig. 9a and b, which emphasizes that the body should be upright and maintain balance. As shown in Fig. 9c and d, $T$ and $B$ are the C-points, and $O$ is the centroid of the practitioner body. Then, by connecting $O$ to $T$ and $B$, we obtain two axes $\overline{OT}$ and $\overline{OB}$ to reveal whether the practitioner is tilting in the front/side view. The tilt angle relative to a system-generated vertical reference line $V_{ref}$ (in red) is also displayed. Let $\theta_T$ ($\theta_B$) be the angle between $\overline{OT}$ ($\overline{OB}$) and $V_{ref}$. If both $\theta_T$ and $\theta_B$ are respectively less than thresholds $\tau_T$ and $\tau_B$, then the practitioner's posture is judged *good*; otherwise, it is judged *improper*, and the system will give an audio alert to inform the practitioner of the improper posture. More formally, this model can be formulated as

$$Pose_{Tree} = \begin{cases} good, if\ \theta_T < \tau_T\ and\ \theta_B < \tau_B; \\ improper, otherwise. \end{cases} \quad (1)$$


**Fig. 6** Dominant axis estimation

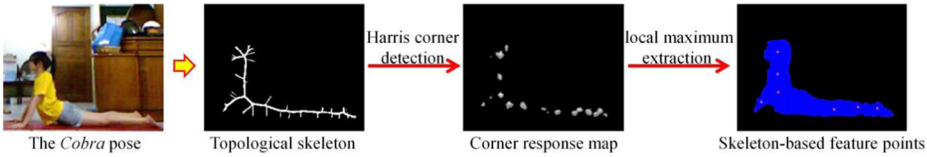



**Fig. 7** Processing flow of skeleton-based feature point detection

The threshold values can be user-specified since professional/experienced practitioners may set a lower threshold to require themselves strictly, while beginners/novices may prefer a higher threshold for higher error tolerance so that the system will not keep alerting. Table 1 lists the threshold values suggested by an experienced yoga expert.

Note that the same posture analysis is conducted for the front views of several other poses, including *Full Boat*, *Extended Hand-to-Big-Toe*, *Chair*, *Warrior II*, *Warrior III*, *Side Plank*, and *Lord of the Dance*. Thus, we will skip the same explanation in the following.

(2) ***Full Boat*** is a great yoga pose for strengthening the abdominal organ. As presented in Fig. 10a and b, the practitioner sits on the floor and lifts feet off the floor. Then, the thighs are angled about 60° relative to the floor. Figure 10c gives the visualized instruction for the front view, which is similar to the *Tree* pose. For the side view, as presented in Fig. 10d, we first extract the topmost S-points to the left and right of the body centroid $O$, termed $P_{LT}$ and $P_{RT}$, respectively. Then, with $O$ and the C-point $B$ obtained, another feature point $P_{CB}$ is located at the position with the same $x$-coordinate as $O$ and the same $y$-coordinate as $B$. By connecting $P_{CB}$ to $P_{LT}$ and $P_{RT}$, the practitioner can perceive the angles of the torso and thighs from the floor. In this example, one can see from Fig. 10d that, as suggested by the system-generated reference line (in red), the practitioner should lift her feet higher. Let $\theta_{LT}$ be the angle between $\overline{P_{CB}P_{LT}}$ and a horizontal reference line $H_{ref}$. The model of *Full Boat* can be formulated as

$$Pose_{Full\ Boat} = \begin{cases} good, if\ \left|\theta_{LT} - 60^{\circ}\right| < \tau_{LT}; \\ improper, otherwise. \end{cases} \tag{2}$$

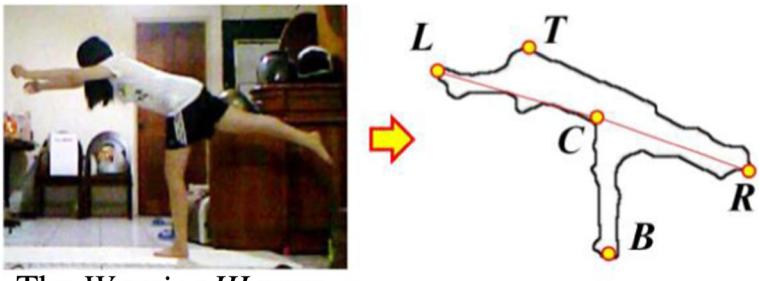



**Fig. 8** Illustration of contour-based feature points: $T$, $B$, $L$, and $R$

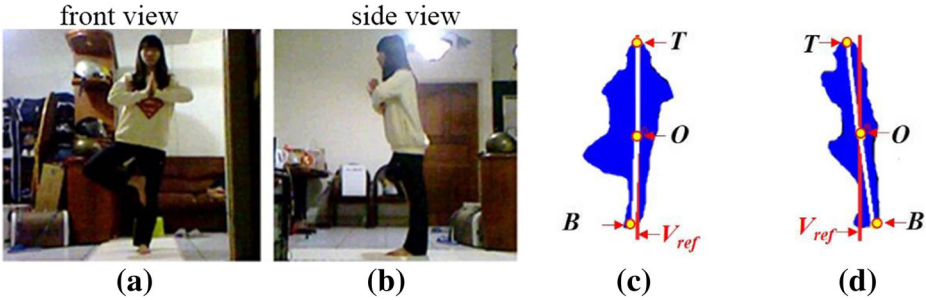



**Fig. 9** The *Tree* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

(3) ***Downward-Facing Dog*** is an essential pose in the majority of yoga classes that the hands are put on the floor, and the back should be lengthened along its entire length so that the arms and back form one line, as demonstrated in Fig. 11a and b. Besides, legs should be stretched and straightened. In both views, the C-points $L$, $R$, and $T$ are used. For the front view in Fig. 11c, the body centroid $O$ is connected to $T$, $L$, and $R$. The axis $\overline{OT}$ discloses the left or right tilt of the body, while $\overline{OL}$ and $\overline{OR}$ show whether the body weight is put equally on both sides. The same front view posture analysis is conducted for the later *Cobra* and *Plank* poses. As for the side view in Fig. 11d, the axes $\overline{TL}$ and $\overline{TR}$ indicate whether the back and arms (in red) form one straight line and whether the legs (in green) are straightened.

(4) ***Extended Hand-to-Big-Toe*** is a pose that helps stretch the hamstrings so the practitioner performing this pose has to stand straight, raise one leg as high as possible, and maintain balance, as shown in Fig. 12a and b. Figure 12c gives the visualized instruction for the front view, which is similar to the *Tree* pose. For the side view, another feature point $X$ is located at the intersection of $V_X$ and $H_X$, as illustrated in Fig. 13. By connecting $X$ to the C-points $T$, $B$, and $L$, as show in Fig. 12d, the axes $\overline{XT}$ and $\overline{XB}$ show whether the body is tilting and $\overline{XL}$ indicates the degree of the raising leg. Let $\theta_T$ ($\theta_B$) be the angle between $\overline{XT}$ ($\overline{XB}$) and a vertical reference line $V_{ref}$, and $\theta_{XL}$ be the

**Table 1** Suggested threshold values

| Threshold | Value (degree) |
|---|---|
| $\tau_T$ | 3 |
| $\tau_B$ | 3 |
| $\tau_{LT}$ | 5 |
| $\tau_{XL}$ | 5 |
| $\tau_{p4}$ | 10 |
| $\tau_{LB}$ | 5 |
| $\tau_L$ | 3 |
| $\tau_R$ | 3 |
| $\tau_{PT}$ | 5 |
| $\tau_{PR}$ | 3 |
| $\tau_{hR}$ | 5 |
| $\tau_{TR}$ | 5 |

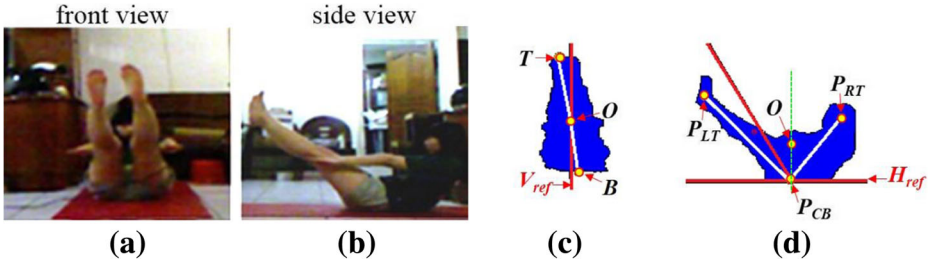



**Fig. 10** The *Full Boat* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

angle between $\overline{XL}$ and a horizontal reference line $H_{ref}$. The model of this pose can be formulated as

$$Pose_{\textbf{Extended Hand}\text{–}\textbf{to}\text{–}\textbf{Big}\text{–}\textbf{Toe}} = \begin{cases} good, if\ \theta_T < \tau_T\ and\ \theta_B < \tau_B\ and\ \theta_{XL} < \tau_{XL}; \\ improper, otherwise. \end{cases} \quad (3)$$

(5) ***Chair***, as shown in Fig. 14a and b, is a pose that the practitioner sits back with arms raised overhead. The practitioner should keep the natural curve of his/her lower back, draw the lower belly, and maintain balance while sending the tailbone towards the ground. Because the pose is just like sitting on a chair, the thighs should be relatively parallel to the floor. Fig. 14c gives the visualized instruction for the front view, which is similar to the *Tree* pose. For the side view, the obtained S-points are sorted according to their $y$-coordinates, termed $q_1, q_2 \ldots q_n$. For noise removal, a point $q_i$ will be discarded if the angle between $\overline{q_{i-1}q_i}$ and $\overline{q_iq_{i+1}}$ is larger than a threshold. From the remaining feature points $p_1, p_2 \ldots p_m$, as shown in Fig. 15, the line segment $\overline{p_ip_{i+1}}$ with the most different orientation from $\overline{TB}$ is selected, i.e., $\overline{p_4p_5}$ in this example. As shown in Fig. 14d, by connecting $T$, $p_4$, $p_5$, and $B$, the practitioner can clearly perceive his/her body structure. Besides, the line segment $\overline{p_4p_5}$ and the system-generated reference line (in red) reveal the angle between the thighs and a horizontal reference line $H_{ref}$. Let $\theta_{p4}$ be the angle between $\overline{p_4p_5}$ and $H_{ref}$. The model of *Chair* can be formulated as

$$Pose_{Chair} = \begin{cases} good, if\ \theta_{p4} < \tau_{p4}; \\ improper, otherwise. \end{cases} \quad (4)$$

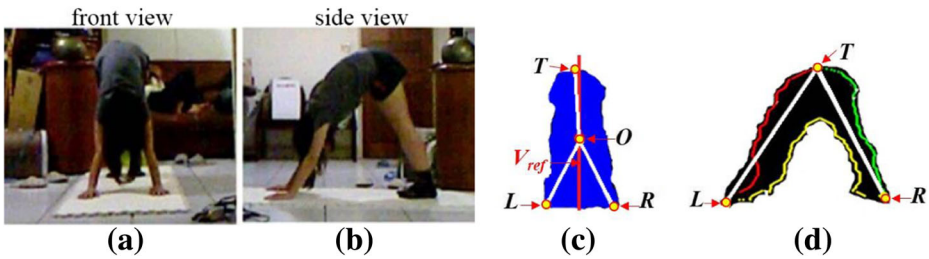



**Fig. 11** The *Downward-Facing Dog* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

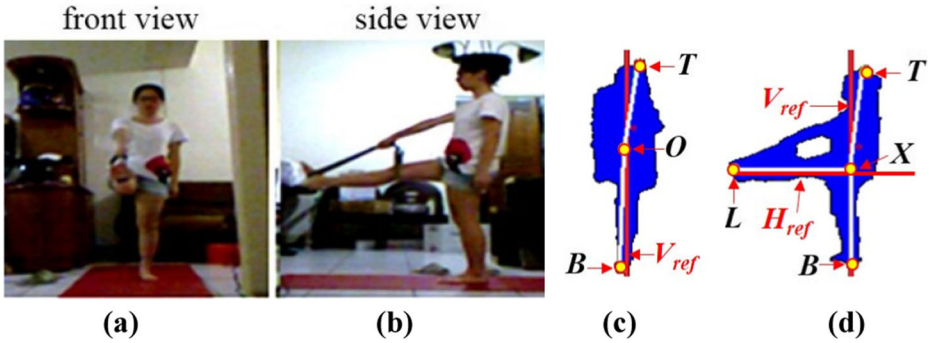



**Fig. 12** The *Extended Hand-to-Big-Toe* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

(6) ***Warrior I*** is a beautiful pose that inspires grace and strength, as presented in Fig. 16a and b. First, the practitioner steps his/her feet apart and raises arms perpendicular to the floor. Then, the front knee is bent over the ankle so the front shin/thigh is perpendicular/parallel to the floor. The front view analysis is similar to that of the *Tree* pose, except that a new feature point *T'* located at the midpoint of two highest C-points is used instead of *T*, as shown in Fig. 16c. For the side view, we first select the topmost S-point below the body centroid *O*, termed $P_{OT}$, as shown in Fig. 17. Then, the bottommost S-point $P_{LB}$ ($P_{RB}$) to the left (right) of $P_{OT}$ is located as the foot position. The front knee position is located at the S-point $P_k$ which is farthest to $\overline{P_{LB}P_{OT}}$ to the lower left of *O*. As shown in Fig. 16d, after connecting $P_{OT}$ to *T*, $P_K$, $P_{RB}$, and connecting $P_k$ to $P_{LB}$, the practitioner's posture can be well visualized. The axis $\overline{TP_{OT}}$ verifies whether the upper body is stretched upright, while $\overline{P_{OT}P_k}$ and $\overline{P_kP_{LB}}$ assist the practitioner to know the angle of the thigh and calf. Let $\theta_T$ be the angle between

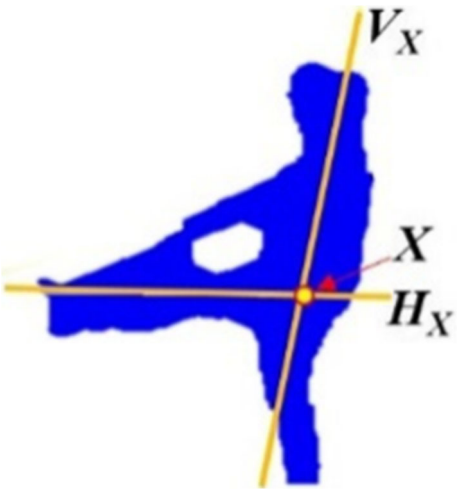



**Fig. 13** The feature point *X* is located at the intersection of dominant axes $V_X$ and $H_X$

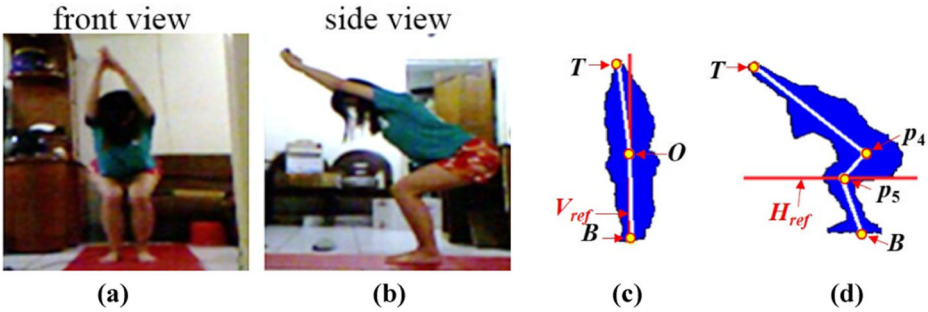



**Fig. 14** The *Chair p*ose. **a**, **b** Original frames. **c**, **d** Visualized instruction

$\overline{TP_{OT}}$ and a vertical reference line $V_{ref}$, and $\theta_{LB}$ be the angle between $\overline{P_{OT}P_k}$ and $\overline{P_kP_{LB}}$. The model of *Warrior I* can be formulated as

$$Pose_{Warrior\ I} = \begin{cases} good, if\ \ \theta_T < \tau_T\ and\ \left|\theta_{LB} - 90^{\circ}\right| < \tau_{LB}; \\ improper, otherwise. \end{cases} \quad (5)$$

(7) ***Warrior II*** is a pose similar to *Warrior I*, as shown in Fig. 18a and b. The lower part of the body is the same as *Warrior I*, but the arms are raised to the side and parallel to the floor. Figure 18c gives the visualized instruction for the front view, which is similar to the *Tree* pose. As for the side view, the feature points of the lower body (the half part below the body centroid *O*), i.e., $P_{OT}$, $P_{LB}$, $P_{RB}$, and $P_k$ are located in the same way as *Warrior I*. For the upper body, the C-points *T*, *L*, and *R* are extracted. Moreover, the S-point closest to the midpoint of $\overline{TP_{OT}}$ is also extracted, termed $P_X$, as shown in Fig. 18d. By connecting $P_X$ to *L* and *R*, the axes can verify whether the arms are parallel to the floor. Let $\theta_T$ be the angle between $\overline{TP_{OT}}$ and a vertical reference line $V_{ref}$, and $\theta_{LB}$ be the angle between $\overline{P_{OT}P_k}$ and $\overline{P_kP_{LB}}$. Let $\theta_L$

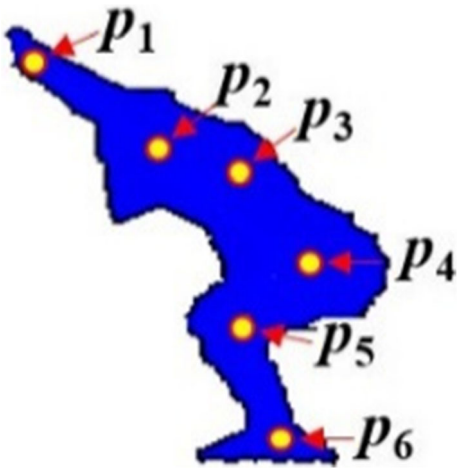



**Fig. 15** Skeleton-based feature points

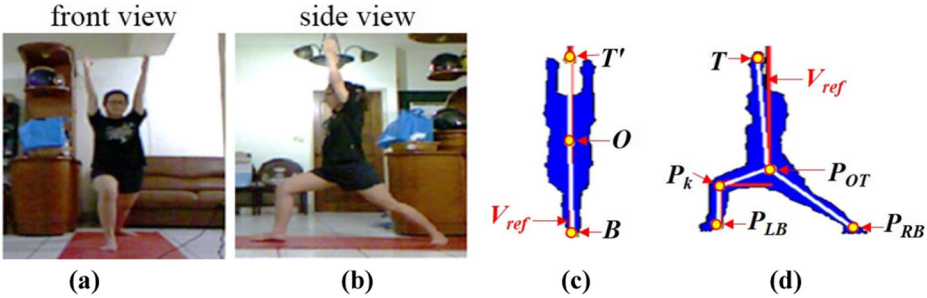



**Fig. 16** The *Warrior I* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

($\theta_R$) be the angle between $\overline{P_X L}$ ($\overline{P_X R}$) and a horizontal reference line $H_{ref}$. The model of *Warrior II* can be formulated as

$$Pose_{Warrior\ II} = \begin{cases} good, if\ \ \theta_T < \tau_T\ and\ \ \left|\theta_{LB} - 90^{\circ}\right| < \tau_{LB}\ and\ \theta_L < \tau_L\ and\ \theta_R < \tau_R; \\ improper, otherwise. \end{cases} \tag{6}$$

(8) ***Warrior III***, as shown in Fig. 19a and b, is kind of difficult that the arms, torso, and one raised leg should be positioned relatively parallel to the floor, with balance maintained by the other leg. Figure 19c gives the visualized instruction for the front view, which is similar to the *Tree* pose. For the side view in Fig. 19d, the C-points $L$, $R$, and $B$ are used, and another feature point $X$ is located at the intersection of $V_X$ and $H_X$, as used in the *Extended Hand-to-Big-Toe* pose. By connecting $X$ to $L$, $R$, and $B$, the axes $\overline{XL}$ and $\overline{XR}$ can be used to measure whether the arms, torso, and raised leg are parallel to the floor, and $\overline{XB}$ shows whether the lower leg is tilting. In the example of Fig. 19d, the practitioner is suggested to raise the leg higher, and the arms and torso can be a little lower. Let $\theta_L$ ($\theta_R$) be the angle between $\overline{XL}$ ($\overline{XR}$) and a

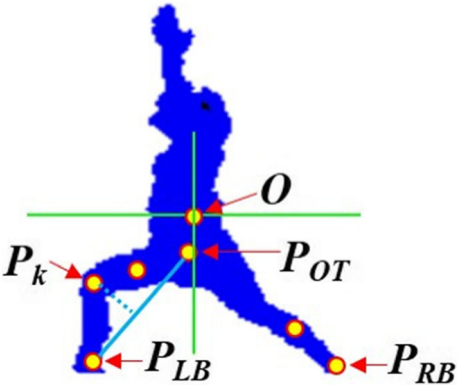



**Fig. 17** Skeleton-based feature point selection

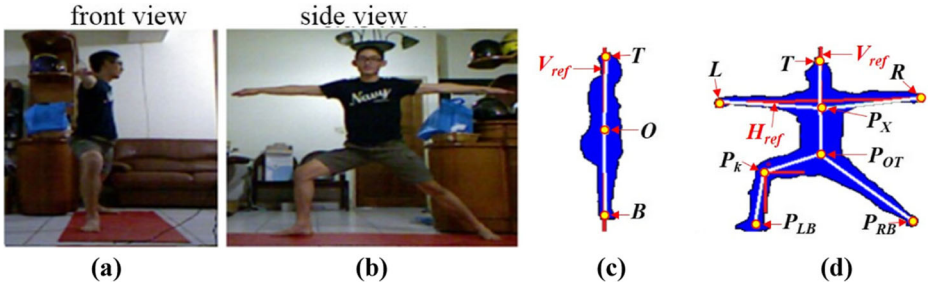



**Fig. 18** The *Warrior II* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

horizontal reference line $H_{ref}$, and $\theta_B$ be the angle between $\overline{XB}$ and a vertical reference line $V_{ref}$. The model of *Warrior III* can be formulated as

$$Pose_{Warrior\ III} = \begin{cases} good, if\ \theta_L < \tau_L\ \ and\ \theta_R < \tau_R\ and\ \theta_B < \tau_B; \\ improper, otherwise. \end{cases} \tag{7}$$

(9) ***Cobra*** is a great exercise for people with lower back aches. As shown in Fig. 20a and b, the practitioner lies face downwards on the floor with the palms flat, placed beneath the shoulders. Then, he/she pushes the upper body off the floor and straightens the arms as much as is comfortable while keeping the hips, legs, and feet planted on the floor. Note that the practitioner should not overdo the back bend; otherwise, he/she may get injured. Figure 20c gives the visualized instruction for the front view, which is similar to the *Downward-Facing Dog* pose. For the side view, we aim to obtain the curve of the main part of the body. In addition to the topmost and rightmost S-points, $P_T$ and $P_R$ respectively, we extract the two S-points closest to the body centroid $O$, termed $P_a$ and $P_b$, as shown in Fig. 20d. By connecting $P_T$, $P_a$, $P_b$, and $P_R$ in order, the practitioner can perceive whether he/she is overdoing the back bend from the visualized instruction. Let $\theta_{PT}$ be the angle between $\overline{P_T P_a}$ and a vertical reference line $V_{ref}$, and $\theta_{PR}$ be the angle between $\overline{P_b P_R}$ and a horizontal reference line $H_{ref}$. The model of *Cobra* can be formulated as

$$Pose_{Cobra} = \begin{cases} good, if\ \theta_{PT} < \tau_{PT}\ \ and\ \theta_{PR} < \tau_{PR}; \\ improper, otherwise. \end{cases} \tag{8}$$

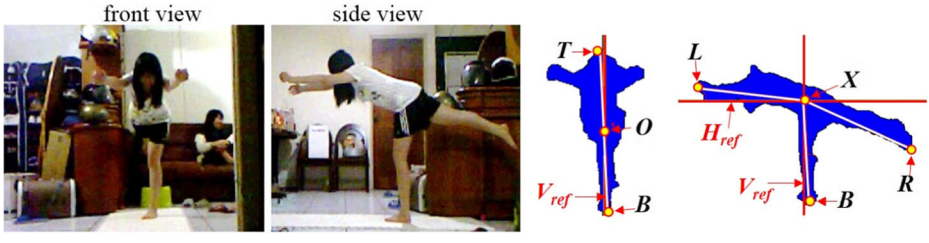



**Fig. 19** The *Warrior III* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

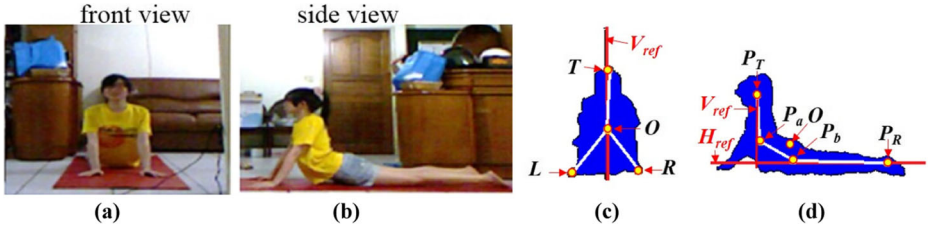



**Fig. 20** The *Cobra* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

(10) ***Plank*** is an arm balancing yoga pose that tones the abdominal muscles while strengthening the arms and spine. As shown in Fig. 21a and b, the practitioner puts the hands on the floor and brings the body into a straight line, from the shoulder to the heels. Figure 21c gives the visualized instruction for the front view, which is similar to the *Downward-Facing Dog* pose. For the side view, the C-point $R$ is extracted as the foot point, and the point $P_{LB}$ as used in *Warrior I* is also extracted, as shown in Fig. 22. Then, we locate the topmost point $U$ on the sub-contour between $P_{LB}$ and $R$, i.e., the green curve in Fig. 22. Let the $x$-coordinate of $U$ be $x_U$. Among the S-points whose $x$-coordinates are within the range of $[x_U - \tau, x_U + \tau]$, the topmost one is selected to represent the shoulder position, termed $P_s$. According to the adult body proportion [30], the ratio of the distance from the shoulder to the hip to that from the hip to the feet is about 1:2. Based on this proportion, we extract the S-point closest to the ideal hip location, termed $P_h$. As shown in Fig. 21d, the two axes formed by connecting $P_h$ to $P_s$ and $R$ can be used to infer whether the practitioner body is presented in a straight line. Let $\theta_{hR}$ be the angle between $\overline{P_sP_h}$ and $\overline{P_hR}$. The model of *Plank* can be formulated as

$$Pose_{Plank} = \begin{cases} good, if \left|\theta_{hR} - 180^{\circ}\right| < \tau_{hR}; \\ improper, otherwise. \end{cases} \tag{9}$$

(11) ***Side Plank*** challenges the stability and improves core strength by working the muscles along the side of the body. As shown in Fig. 23a and b, the practitioner starts the pose by lying on the side with legs straight and feet stacked. Then, the practitioner straightens the bottom arm, raises the hips until the body forms a straight line from the shoulder to the ankles, and extends the other hand toward the ceiling. Figure 23c gives the visualized instruction for the front view, which is similar to the *Tree* pose. For the side view, the feature point $X$ is extracted in the same way as *Extended Hand-to-Big-*

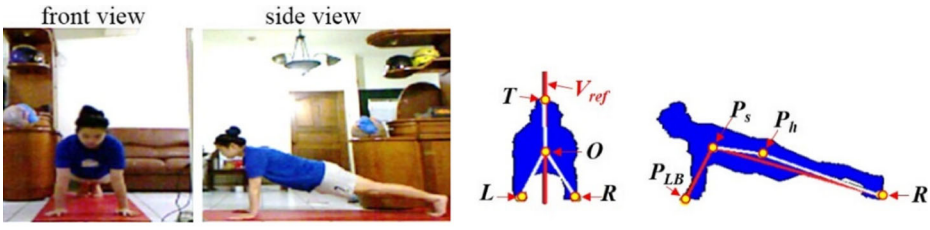



**Fig. 21** The *Plank* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

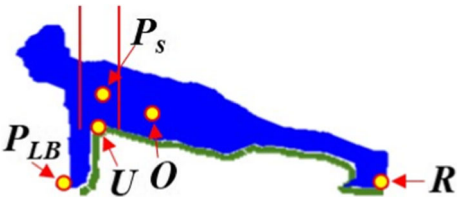



**Fig. 22** Illustration of feature point extraction for the *Plank* pose

*Toe*, and $P_{LB}$ and $P_h$ are extracted in the same way as *Plank*, as shown in Fig. 23d. Besides, the C-points $T$ and $R$ are also used. By connecting $X$ to $T$ and $P_{LB}$, whether the arms form a straight line is visualized. By connecting $P_h$ to $X$ and $R$, the practitioner can know whether his/her body is presented in a straight line. Let $\theta_{hR}$ be the angle between $\overline{XP_h}$ and $\overline{P_hR}$. The model of *Side Plank* can be formulated as

$$Pose_{Side\ Plank} = \begin{cases} good, if\ \left|\theta_{hR} - 180^{\circ}\right| < \tau_{hR}; \\ improper, otherwise. \end{cases} \tag{10}$$

(12) ***Lord of the Dance*** requires the pose to be done gracefully, almost like a dance, as shown in Fig. 24a and b. The first is to reach back with the left (right) hand and grasp the outside of the left (right) foot or ankle. Then, the practitioner begins to lift the left (right) foot up, away from the floor, and back, away from his/her torso with stretching the right (left) arm forward, in front of the torso. Figure 24c gives the visualized instruction for the front view, which is similar to the *Tree* pose. For the side view, the practitioner has to keep balance and try hard to raise the back foot as high as possible, so we extract the rightmost S-points above and below the body centroid $O$, termed $P_{TR}$ and $P_{BR}$, respectively, to locate the raised leg. In addition, the S-point closest to $P_T$ (c.f. the *Cobra* pose) is located, termed $P_n$, as shown in Fig. 24d. By connecting $P_n$ to $L$ and $P_{TR}$, the axes $\overline{P_nL}$ and $\overline{P_nP_{TR}}$ can reveal if the foot (leg) is raised high enough. Let $\theta_L$ ($\theta_{TR}$) be the angle between $\overline{P_nL}$ ($\overline{P_nP_{TR}}$) and a horizontal reference line $H_{ref}$. The model of this pose can be formulated as

$$Pose_{\textbf{Lord of the Dance}} = \begin{cases} good, if \theta_L < \tau_L \ \ and\ \theta_{TR} < \tau_{TR}; \\ improper, otherwise. \end{cases} \tag{11}$$

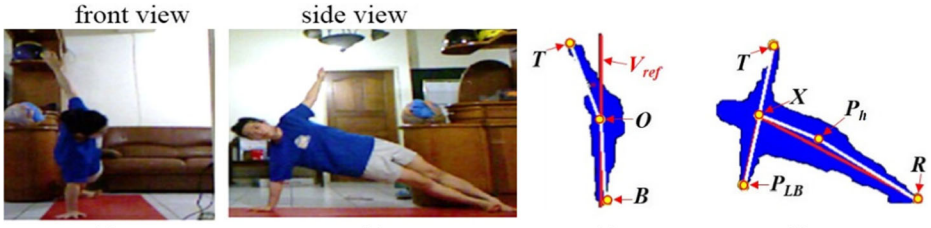



**Fig. 23** The *Side Plank* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

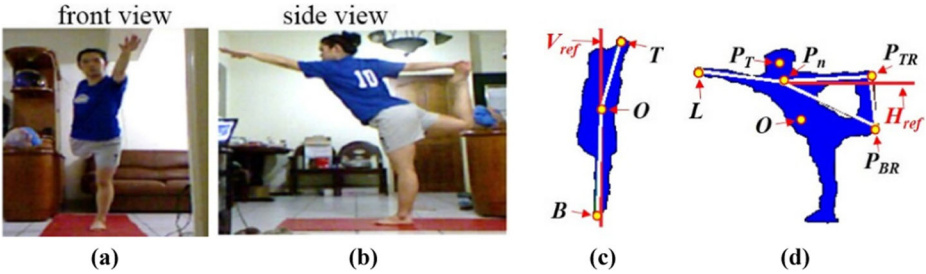



**Fig. 24** The *Lord of the Dance* pose. **a**, **b** Original frames. **c**, **d** Visualized instruction

## 4 Experimental results

The proposed yoga training system is implemented in C++ with OpenCV (Open Source Computer Vision) 2.3.1 libraries [23], and runs on an Acer notebook (Intel Core i5 CPU M430 @2.27GHz, 4GB RAM, Windows 7 64-bit OS). Twelve typical yoga poses (abbr. $Y_1$ to $Y_{12}$), as given in Fig. 1, are selected in our system. Experiments are conducted in such a way that five practitioners perform each of the twelve yoga poses five times. Then a yoga expert is asked to judge whether the visualized instruction generated by the proposed system is appropriate frame by frame through a simple user-friendly interface.

The performance evaluation of visualized instruction for posture rectification is presented in Table 2, wherein the first columns indicate the twelve yoga poses to be analyzed. In the top row, "#*G*" and "#*N*G" respectively indicate the numbers of *good* and *no-good* visualized instructions, and the accuracy is computed by

$$Accuracy = \frac{\#G}{\#G + \#NG}. \tag{12}$$

The experimental results are also summarized in Fig. 25, showing that for most of the yoga poses, the feature points can be located correctly and the visualized

**Table 2** Results of visualized instruction generation

| Pose | Frame# | Front view | | | Side view | | |
|---|---|---|---|---|---|---|---|
| | | *#G* | *#NG* | Accuracy | *#G* | *#NG* | Accuracy |
| $Y_1$ | 3711 | 3706 | 5 | 99.87% | 3581 | 130 | 96.50% |
| $Y_2$ | 1637 | 1547 | 90 | 94.50% | 1584 | 53 | 96.76% |
| $Y_3$ | 2943 | 2873 | 70 | 97.62% | 2918 | 25 | 99.15% |
| $Y_4$ | 2014 | 1961 | 53 | 97.37% | 1714 | 300 | 85.10% |
| $Y_5$ | 1924 | 1852 | 72 | 96.26% | 1750 | 174 | 90.96% |
| $Y_6$ | 2635 | 2589 | 46 | 98.25% | 2502 | 133 | 94.95% |
| $Y_7$ | 2840 | 2826 | 14 | 99.51% | 2638 | 202 | 92.89% |
| $Y_8$ | 3327 | 2536 | 791 | 76.22% | 2974 | 353 | 89.39% |
| $Y_9$ | 2426 | 2249 | 177 | 92.70% | 2294 | 132 | 94.56% |
| $Y_{10}$ | 2237 | 2169 | 68 | 96.96% | 2096 | 141 | 93.70% |
| $Y_{11}$ | 1533 | 1509 | 24 | 98.43% | 1407 | 126 | 91.78% |
| $Y_{12}$ | 2033 | 2023 | 10 | 99.51% | 1888 | 145 | 92.87% |

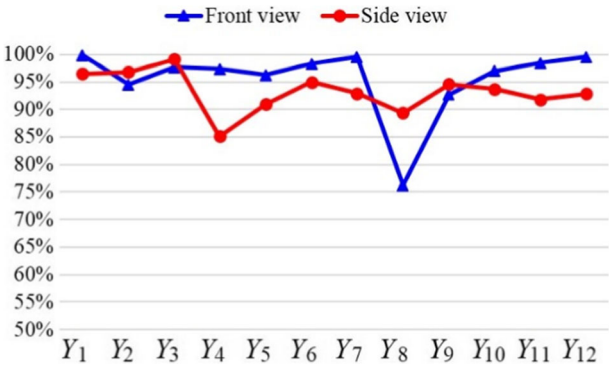



**Fig. 25** Performance of our proposed system

instructions can be generated appropriately. However, the system may get confused in extracting feature points for some poses, such as $Y_4$ (*Extended Hand-to-Big-Toe*) and $Y_8$ (*Warrior III*). Especially, the accuracy of $Y_8$ front view is relatively lower than the others. This is because when performing $Y_8$, a practitioner tends to put his/her head too low so that his/her hand or shoulder may be mis-detected as the *C*-point *T*, as shown in Fig. 26, resulting in incorrect visualized instructions. It would be one of our future works to improve or even redesign the methods of feature point detection and assistant axis generation for some poses. Based on observation, we discover that there are also several errors in visualized instructions caused by incorrect segmentation of the body map. As shown in Fig. 27, the visualized instruction cannot represent the raised leg satisfactorily due to incorrect segmentation of the foot.

For performance comparison, we also try generating body skeletons for yoga poses with Microsoft Kinect and OpenNI library [24]. With a RGB camera and a depth sensor, Kinect is known to be able to access the depth information of a human body and estimate/track the articulate pose. However, as demonstrated in Fig. 28, the results are not as good as expected. Based on observation, we find that the Kinect can generate good body skeletons in the situation that the body and limbs are distinguishable. However, when performing yoga, some body parts may be occluded, leading to incorrect body skeletons. For example, the skeletons in Fig. 28b, c, and e can describe the yoga poses neither from the front view nor from side view. This is why we design explicit rules for feature extraction and posture judgment, instead of directly using the Kinect skeleton.

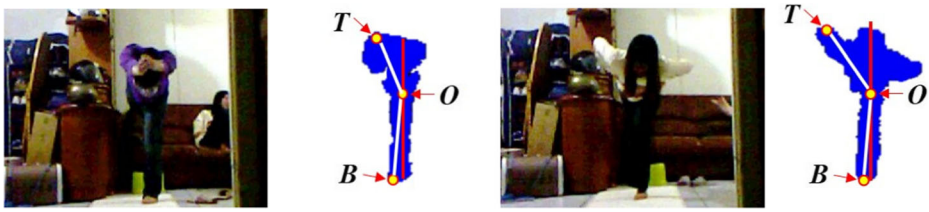



**Fig. 26** Two error cases of $Y_8$ (*Warrior III*) that the C-point *T* is located incorrectly

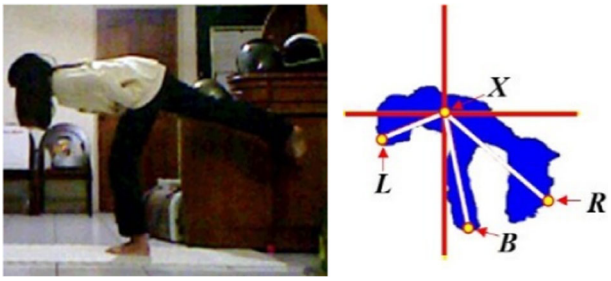
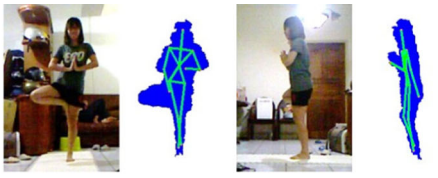
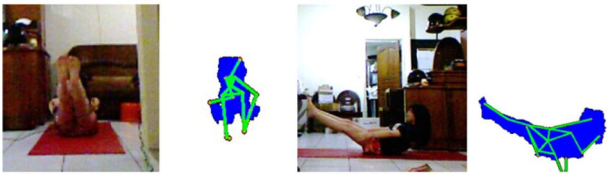


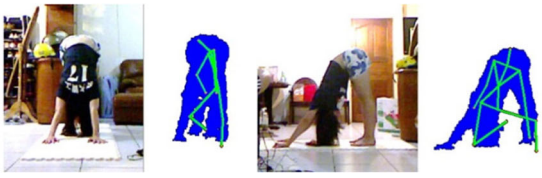

**Fig. 27** Improper visualized instruction due to incorrect segmentation of the foot



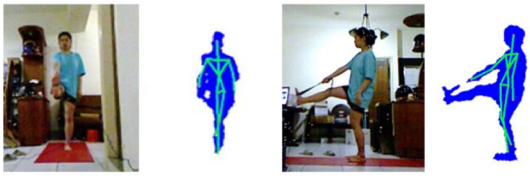
**(a)** The *Tree* pose.



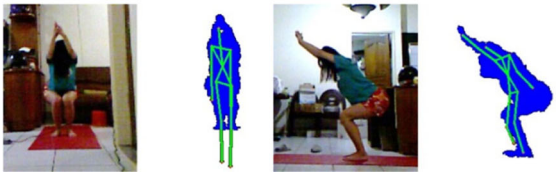
**(b)** The *Full Boat* pose.



**(c)** The *Downward-Facing Dog* pose.



**(d)** The *Extended Hand-to-Big-Toe* pose.



**(e)** The *Chair* pose.

**Fig. 28** Body skeletons generated with Kinect for yoga poses

**(f)** The *Warrior I* pose.

**(g)** The *Warrior II* pose.

**(h)** The *Warrior III* pose.

**(i)** The *Cobra* pose.

**(j)** The *Plank* pose.

**(k)** The *Side Plank* pose.

**(l)** The *Lord of the Dance* pose.

**Fig. 28** (continued)

## 5 Conclusion

Computer-assisted self-training in sports exercise is an ever-growing trend. In this paper, we develop a computer-assisted yoga training system, aiming at instructing the practitioner to

perform yoga poses correctly and preventing injury caused by improper postures. Integrating computer vision techniques, the proposed system analyzes the practitioner's posture from both front and side views by extracting the contour, skeleton, dominant axes, and feature points of the human body. Then, based on the domain knowledge of yoga training, visualized instructions for posture rectification are presented so that the practitioner can easily understand how to adjust his/her posture.

Although the experiments show satisfactory results, there is still much room for us to improve. For example, the current Y-system may get confused in extracting feature points for some poses, such as $Y_4$ (*Extended Hand-to-Big-Toe*) and $Y_8$ (*Warrior III*). Thus, it will be one of our future works to improve or even redesign the methods of feature point detection and assistant axis generation for some poses, making the system more solid. Besides, we are working on enhancing the system by adding more modules of other yoga poses. Also, we attempt to enhance the system by adding voice feedback.

**Acknowledgments** This research is supported in part by MOST-106-2221-E-009-196, MOST-106-2221-E-035-102, MOST-105-2221-E-009-065, MOST-104-3115-E-009-001, ICTL-103-Q528, and ATU-103-W958.

## References

1. Brunnett G, Rusdorf S, Lorenz M (2006) V-Pong: an immersive table tennis simulation. IEEE Comput Graph Appl 26(4):10–13
2. Chen HT, Chen HS, Hsiao MH, Tsai WJ, Lee SY (2008) A trajectory-based ball tracking framework with enrichment for broadcast baseball videos. J Inf Sci Eng 24(1):143–157
3. Chen HT, Tien MC, Chen YW, Tsai WJ, Lee SY (2009) Physics-based ball tracking and 3D trajectory reconstruction with applications to shooting location estimation in basketball video. J Vis Commun Image Represent 20(3):204–216
4. Chen HT, Tsai WJ, Lee SY, Yu JY (2009) Ball tracking and 3D trajectory approximation with applications to tactics analysis from single-camera volleyball sequences. Multimedia Tools Appl 6(3):641–667
5. Chen HT, Chou CL, Fu TS, Lee SY, Lin BS (2012) Recognizing tactic patterns in broadcast basketball video using player trajectory. J Vis Commun Image Represent 23(6):932–947
6. Chen HT, He YZ, Chou CL, Lee SY, Lin BS, Yu JY (2013) Computer-assisted self-training system for sports exercise using kinects. In: Proc IEEE ICME 2013:1–4
7. Chen HT, He YZ, Hsu CC, Chou CL, Lee SY, Lin BS (2014) Yoga posture recognition for self-training. In: Proc MMM 2014:496–505
8. Chen HT, Huang TW, Chou CL, Tsai HC, Lee SY (2015) Improving golf swing skills using intelligent glasses. In: Proc VCIP 2005:1–4
9. Chong AK, Croft H (2009) A photogrammetric application in virtual sport training. Photogramm Rec 24(125):51–65
10. Chong AK, Milburn P, Newsham West R, ter Voert M, Croft H (2008) Recent practical applications of close-range photogrammetry for complex motion study. Int Arch Photogramm Rem Sens Spatial Inform Sci 37(Part B5):921–926
11. Chou CW, Tien MC, Wu JL (2009) Billiards wizard: a tutoring system for broadcasting nine-ball billiards videos. In: Proc IEEE ICASSP 2009:1921–1924
12. Ghasemzadeh H, Loseu V, Jafari R (2009) Wearable coach for sport training: a quantitative model to evaluate wrist-rotation in golf. J Ambient Intell Smart Environ 1(2):173–184
13. Harris C, Stephens M (1988) A combined corner and edge detector. In: Proc Alvey vision conference 15: 147–151
14. Höferlin M, Grundy E, Borgo R, Weiskopf D, Chen M, Griffiths IW, Griffiths W (2010) Video visualization for snooker skill training. Comput Graphics Forum 29(3):1053–1062
15. Hsieh CC, Wu BS, Lee CC (2011) A distance computer vision assisted yoga learning system. J Comput 6(11):2382–2388
16. Hu MC, Chang MH, Wu JL, Chi L (2011) Robust camera calibration and player tracking in broadcast basketball video. IEEE Trans Multimedia 13(2):266–279

17. Kelly P, Healy A, Moran K, O'Connor NE (2010) A virtual coaching environment for improving golf swing. In: Proc ACM workshop surreal media and virtual cloning 2010:51–56
18. King K, Yoon SW, Perkins NC, Najafi K (2008) Wireless MEMS inertial sensor system for golf swing dynamics. Sensors Actuators A Phys 141(2):619–630
19. Luo Z, Yang W, Ding ZQ, Liu L, Chen IM, Yeo SH, Ling KV, Duh HBL (2011) Left arm up! Interactive yoga training in virtual environment. In: Proc. IEEE VR 2011:261–262
20. Miles HC, Pop SR, Watt SJ, Lawrence GP, John NW (2012) A review of virtual environments for training in ball sports. Comput Graph 36(6):714–726
21. Miles HC, Pop SR, Watt SJ, Lawrence GP, John NW, Perrot V, Mallet P, Mestre DR (2013) Investigation of a virtual environment for rugby skills training. In: Proc IEEE int conf cyberworlds 2013:56–63
22. Noiumkar S, Tirakoat S (2013) Use of optical motion capture in sports science: a case study of golf swing. In: Proc int conf informatics and creative multimedia 2013:310–313
23. OpenCV. [Online]. Available: http://opencv.org/
24. OpenNI. [Online]. Available: http://www.openni.ru/
25. Patil S, Pawar A, Peshave A, Ansari AN, Navada A (2011) Yoga tutor visualization and analysis using SURF algorithm. In: Proc. IEEE ICSGRC 2011:43–46
26. Rector K, Bennett CL, Kientz JA (2013) Eyes-free yoga: an exergame using depth cameras for blind & low vision exercise. In: Proc 15th int ACM SIGACCESS conf on computers and accessibility 1–12
27. Rusdorf S, Brunnett G, Lorenz M, Winkler T (2007) Real-time interaction with a humanoid avatar in an immersive table tennis simulation. IEEE Trans Visual Comput Graphics 13(1):15–25
28. Shih CH, Koong CS, Hsiung PA (2012) Billiard combat modeling and simulation based on optimal cue placement control and strategic planning. J Intell Robot Syst 67(1):25–41
29. Sousa L, Alves R, Rodrigues JMF (2016) Augmented reality system to assist inexperienced pool players. Comput Vis Media 2(2):183–193
30. Wikipedia. [Online]. Available: http://en.wikipedia.org/wiki/Body_proportions
31. Wu W, Yin W, Guo F (2010) Learning and self-instruction expert system for Yoga. In Proc 2nd int workshop on intelligent systems and applications 2010:1–4
32. Yoga Journal. [Online]. Available: http://www.yogajournal.com/
33. Zhu G, Xu C, Huang Q (2009) Sports video analysis: from semantics to tactics. In: Divakaran A (ed) Multimedia content analysis. Signals and Communication Technology. Springer, Boston, pp 295–338
34. Zhu G, Xu C, Huang Q, Rui Y, Jiang S, Gao W, Yao H (2009) Event tactic analysis based on broadcast sports video. IEEE Trans Multimedia 11(1):49–67

**Hua-Tsung Chen** received the B.S., M.S., and Ph.D. degrees in Computer Science and Information Engineering from National Chiao Tung University, Hsinchu, Taiwan in 2001, 2003, and 2009, respectively. He is currently an Associate Professor with the Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan. His research interests include computer vision, video signal processing, content-based video indexing and retrieval, multimedia information system, and music signal processing.

**Yu-Zhen He** received the B.S. degree in Applied Mathematics from National Chiao Tung University, Hsinchu, Taiwan in 2011, and the M.S. degree in Computer Science from National Chiao Tung University in 2013. Her research interests include computer vision, video signal processing, and human posture analysis.

**Chun-Chieh Hsu** received the B.S. and M.S. degrees in computer science from National Chiao Tung University, Hsinchu, Taiwan, in 2009 and 2011, respectively, and is currently working toward the Ph.D. degree in computer science at National Chiao Tung University. His research interests include computer vision, video signal processing, sports video analysis, and video surveillance system.