Combining Body Sensors and Visual Sensors for Motion Training

Doo Young Kwon
Computer Graphics Laboratory
ETH Zurich, Switzerland
dkwon@inf.ethz.ch

Markus Gross
Computer Graphics Laboratory
ETH Zurich, Switzerland
grossm@inf.ethz.ch

ABSTRACT

We present a new framework to build motion training systems using machine learning techniques. The goal of our approach is the design of a training method based on the combination of body and visual sensors. We introduce the concept of a Motion Chunk to analyze human motion and construct a motion data model in real-time. The system provides motion detection and evaluation and visual feedback generation. We discuss the results of user tests regarding the system efficiency in martial art training. With our system, trainers can generate motion training videos and practice complex motions precisely evaluated by a computer.

Categories and Subject Descriptors

H.5.1 [**Information Interfaces and Presentations**]: Multimedia Information Systems; I.3.6 [**Computer Graphics**]: Methodology and Techniques—*Interaction Techniques*

Keywords

Motion Training System, Body Sensor, Visual Sensor, Motion Chunk, Motion Analysis, Visual Feedback

1. INTRODUCTION

Watching and following motions performed by a trainer has been considered a fundamental principle for motion training. Beyond the conventional books and video, there have been plenty of interactive CD-ROMs with multimedia contents. Moreover, to provide a bidirectional motion based interaction, Virtual Reality (VR) systems have also been employed to check how a trainee follows an avatar [6, 14]. Along this line, motion training environments have been investigated focusing on the interactivity based on human motion and feedback type.

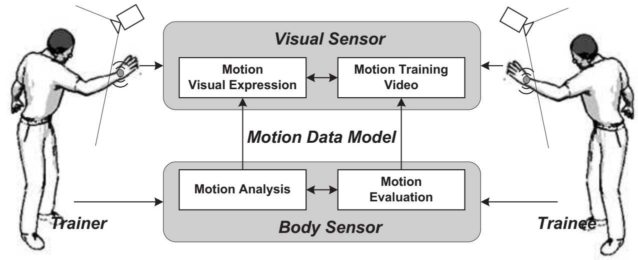

Figure 1: Concept diagram. Body sensors and visual sensors are combined to develop a motion training system for both a trainer and a trainee.

However, most of the current systems use only visual sensors to reconstruct the user's postures and check how the trainee imitates the trainer's motion. These approaches are limited in sensing and evaluating detailed movements which can be critical in the practical motion training. Furthermore, the previous systems have been developed only for trainees and not for trainers. Generally, traditional motion training is performed between trainers and trainees. A motion training system should be a medium where trainees practice following expert motions and trainers also perform motions to create instructive material which the trainees refer to. Thus providing functionality for both trainees and trainers is required. In this paper, we describe our approach to build a motion training system for both trainers and trainees. Our goal is twofold. Firstly, we aim to combine body sensor and visual sensor data to provide an unique motion training method which improves the traditional motion training. Secondly, we provide intelligent functionalities of a motion training system that are important for trainers and trainees, including motion evaluation and motion training video generation, as shown in figure 1.

We utilize wireless sensor technologies for a body sensor and a camera as a visual sensor remaining the users' movements un-tethered. The body sensor precisely measures the tilt detection, movement, and vibration of the body parts. On the other hand, like a mirror in conventional training places, the visual sensor provides the images of the users in real-time. Combining these heterogeneous sensor types, we improve the required tasks for motion training. For example, the accelerometer on a trainer's wrist provides precise tilt angles of the hand and amount of speed changes which are not visible to the naked eyes. Users can observe the sensor data of the performed motions and correct their motions by comparing them with another user's data such as a trainer.

In a real-time motion training system, one challenge is to continuously capture and process human motions. We introduce the *motion chunk* as a flexible segment unit to store a piece of motion information. Motion chunk allows us to make a motion model out of unstructured human motions. With this model, we can apply motion detection and evaluation for motion training. We developed various functionalities for both trainers and trainees. Trainers and trainees can analyze their motion performances by watching a hybrid representation of visual and body sensor data and by generating a motion training video automatically. Thus, they do not have to manually record and edit the video for editing image frames.

We performed user tests in martial arts training to validate our training system (figure 2). Since we have developed our framework as general as possible, the approach can be applied to a wide variety of motion trainings which require real-time motion recognition and visual motion data processing.

2. BACKGROUND AND RELATED WORK

The following research is relevant to our work. It includes specifically sensors, motion analysis, and prior art regarding motion training. Due to large body of literature in these fields we will only give an overview of the most relevant publications.

2.1 Sensors

The advances in sensor technologies including wireless sensors and cameras has increased interest in motion analysis [8]. The sensors can be categorized into two groups: environment sensors and body sensors. Environmental sensors such as ultrasound trackers or visual sensors like cameras have been investigated to capture motion not burdening a user with heavy sensing devices [1]. Specifically, visual sensors have been utilized to reconstruct a 3D user body which can be useful to analyze human motions [9]. A typical example of using body sensors is a wearable computer system. In contrast to environment sensors, body sensors can provide straightforward motion information. These days, further advances in miniaturization make it possible to develop sensors in the convenient form factor of watches, bracelets, adhesive patches, or belt which could be placed on various parts of the user's body. Body-centric sensors do not hamper the users' movement any more. Thus combining body sensors and visual sensors has become a demanding task for developing interactive applications. In this paper, we use both a body sensor and a visual sensor to support motion training.

2.2 Motion Analysis

The speech analysis has been developed using the *Hidden Markov Model* (HMM) to recognize words and sentences and to verify speakers [10]. However, in motion analysis, it is still problematic to process real-time motion signals and apply machine learning techniques to motion data. Thus body sensor have been mainly used to analyze pre-recorded general human activity using machine learning techniques such as running, walking, etc [3]. This paper introduces a novel concept called *motion chunk* to structure and analyze sequential human motions. We analyze motion data for motion training using a Hidden Markov Model. We employ wireless sensors to analyze complex user-defined motions in real-time. Wireless sensors have also been studied for visualizing human motion using audiovisual media components for entertainment [11]. While these approaches combine visual and body sensors, we apply sophisticated machine learning techniques to analyze complex motions. Using the HMM, Chambers used a body sensor for the purpose of annotating video frames [5]. Starner and Pentland recognized hand gestures out of the vocabulary of the American Sign Language using a camera [12]. However, we detect and evaluate human motions for motion training by combining visual and body sensor data.

2.3 Systems for Motion Training

A number of applications have been proposed for motion training systems. Davis developed a vision-based motion training system, called Virtual PAT (Personal Aerobics - Trainer) using IR light sources providing manually pre-recorded instructive videos and audio feedback [7]. Becker described a system for teaching Tai Chi gestures based on head and hand tracking by using a pair of stereo cameras [4]. Yang developed the "Just Follow Me" system using an optical motion capture system [14]. From this, Baek proposed evaluation methods by retargeting trainees' motion to the pre-generated avatar [2]. Chua developed a wireless VR system for Tai Chi training using a light-weight HMD display and optical motion capture device [6]. The trainees' motions are evaluated based on skeleton matching to measure how they mimic avatar motions. Takahata presented a martial art training method using sound generators and accelerometers without providing motion recognition and visual feedback [13].

To the best of our knowledge, our system is the first approach which combines visual and body sensor data to develop a motion training system. While most of the previous training systems have been developed for trainees, ours supports both trainers and trainees. Using the machine learning techniques, we structuralize and label human motions in real time and automatically generate an instructive motion training video. Thus, we achieved full automation while supporting motion training functionalities. We evaluated our training methods in a scenario of teaching martial arts.

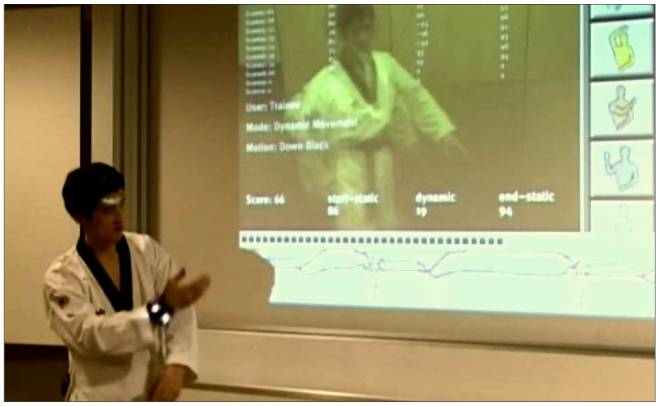

Figure 2: Motion training system in action.

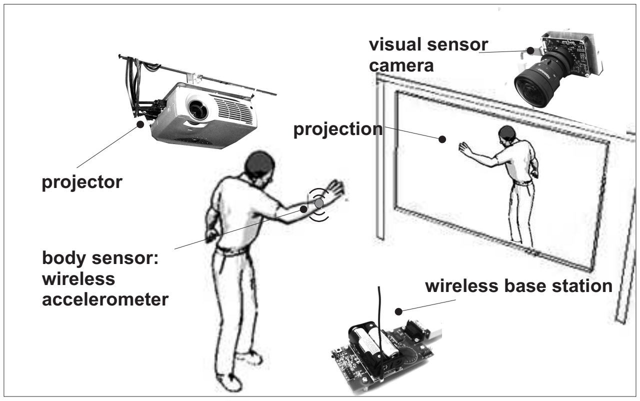

Figure 3: Motion training system setup using a body sensor, a visual sensor, and display devices.

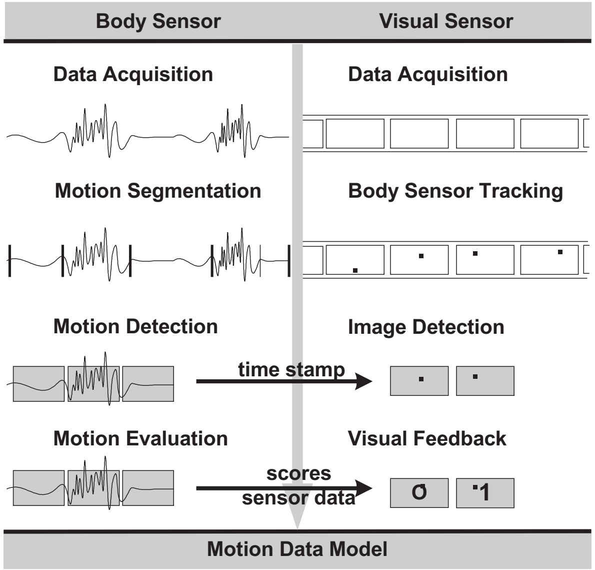

Figure 4: The procedure to evaluate human motions and to generate visual feedback combining visual and body sensor data.

3. SYSTEM ARCHITECTURE

The system consists of a visual sensor (camera), a body sensor (wireless sensor network) and a display device (a projector or a monitor) (figure 3). The system is operated with a series of software components that constructs a motion data model by combining body and visual sensor data. Figure 4 describes the sequential dataflow between the components. During data acquisition, we collect signals from both the body sensor and the visual sensor. Wireless accelerometers transfer signals to the sensor base station which is connected to the main PC. We read the data with a sample rate of 10 Hz and transmit each packet 10 readings in size, so that the update frequency is overall 100 Hz. Synchronously, a Point Grey Research Dragonfly captures 15 images per second and transmits the data to the host PC.

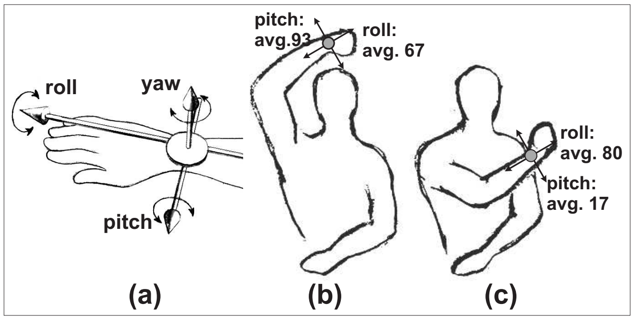

Figure 5: Euler coordinate system of our body sensor (a) roll, pitch and yaw axes of the sensor on the wrist, (b) an example posture 1 with roll average 67 and pitch average 93 (c) an example posture 2 with roll average 80 and pitch average 17 in range 1-100

The host PC accomplishes several steps to analyze the human motion in real time. First, signal segmentation (Sect. 4.2) is performed to divide signals based on the structure of a motion chunk (Sect. 4.1). To recognize reference motions from the segmented motion chunks, motion detection (Sect. 4.3) is performed based on HMMs. Afterwards, the input motion is evaluated and assigned a score by comparing with reference data (Sec. 4.4). For processing visual sensor data, first the acquired image data is processed in real-time to track the body position. Visual and body sensor data are synchronized by time-stamp. We generate visual feedback in the images incorporating the body sensor data (Sect. 5.2). The next chapter describes each component in detail.

3.1 Body Sensor

For the accelerometer we use an Euler coordinate system. The orientation is represented by three different angular values: yaw, pitch and roll. These values are commonly used to describe the movement of a ship or a plane. To measure these values, the accelerometer is the most suitable because it enables the detection of tilt, movement, and vibration. The small size of such sensors is also appropriate for the human body. For example, we can attach the sensor to the wrist like a watch (figure 5). Then, the roll axis of the sensor is parallel to the forearm, and the pitch axis is horizontal and perpendicular to the roll axis. Yaw values point out the upright position of the hand. Unlike the other two angles, yaw values are changed depending on the absolute orientation of the attached body part. In our setting, we utilize only 2-axes of the accelerometer providing pitch and roll, as shown in figure 5c. Using pitch and roll, we can measure the posture of the body part it is attached to, i.e. the forearm. We can infer other, adjacent body parts as well. For example, since the sensor is located on the forearm near the hand, it is also indicative for the orientation of the hand. That is, we can estimate whether the palm is facing back or front, or facing up and down. We divide the body sensor information into two categories: *postures* and *gestures*. Postures are static expressions described orientation, whereas gestures are dynamic movements essentially defined by velocity and by the changes thereof. For instance, if the forearm rests in a certain position, it provides constant values for roll and pitch, i.e. a posture. Figure 5 illustrates different static accelerations of two postures which have different average roll and pitch angles. On the other hand, when the forearm moves

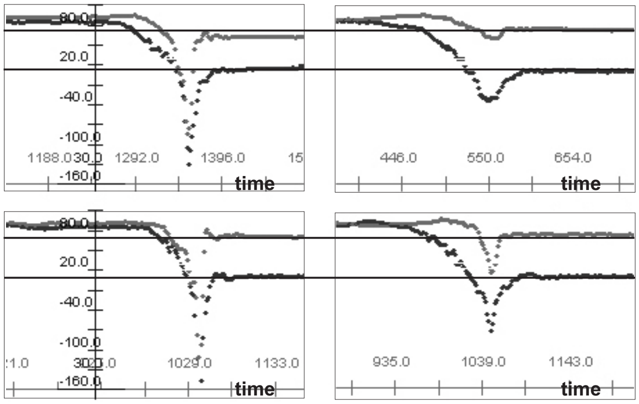

Figure 6: 4 example roll and pitch signals of a gesture ranging from a start posture (1) to and end posture (2). Notice that each signal starts and ends with approximately same average roll and pitch values.

in space, it provides a precise rate of change of velocity over time, i.e. a gesture. Figure 6 shows a collection of signals generated between two postures. As can be seen in Figure 6, when a motion is performed between two distinct postures, an acceleration signal starts at the end of posture 1 and finishes with the beginning of posture 2. As will be explained subsequently, it is this acceleration signal that we utilize as a basis for our motion analysis.

4. MOTION RECOGNITION

We developed a method for motion analysis which supports motion training functionalities in real-time. We introduce the notion of a motion chunk to explain how to decompose and analyze human motion. Then, we process the sensor data following the techniques outlined in figure 4.

4.1 Motion Chunk

We introduce the motion chunk to process and decompose unstructured human motion. Similar to the human voice, human motion is sequential in time. We assume that human motion can be represented with a sequential combination of chunks similar to speech analysis. We induce several types of motion structure from the recognition point of view: single motion recognition, recognition of a sequence of motions, and overall motion understanding. We extend the analogy as follows: a single motion such as punching, blocking, kicking, and striking can be considered as a "word level" motion since they do not involve sequences of other motions. Using "word level" motion recognition, we can detect "sentence level" motions by evaluating transition probabilities between "word level" activities. For example, sparring could be a sentence level motion involving sequences of punching and blocking.

The basic idea at the motion chunk is to decompose complex, sequential human motions into atomic units to simplify analysis. These units, called motion chunks, are similar in spirit to phonemes in speech recognition. Following our earlier definitions of postures and gestures we create two types of motion chunks: *static chunks* and *dynamic chunks*.

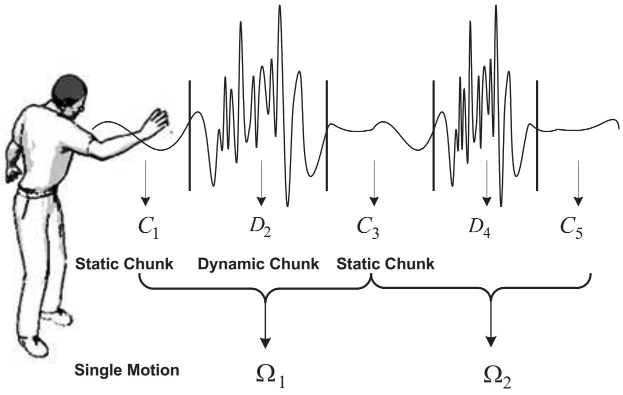

Figure 7: Structure of a motion chunk. A recognized single motion consists of two static chunks and one dynamic chunk.

We now define the motion chunk of one single motion as a combination of three chunks: *start-static chunk+dynamic chunk+end-static chunk*. This is intuitively clear as it combines the start posture, the gesture, and the end posture. Figure 7 illustrates that a single step motion consists of two static chunks denoted by C and one dynamic chunk denoted by D. Likewise, two step motions are combined sharing one in-between static chunk and so forth. To recognize an input single motion, each of the K single motions known to the system are assigned a motion type Ω_K. The recognition rule r_K maps the observation sequence on the basis of the start-static chunk C_{i-1}, the end-static chunk C_{i+1}, and the dynamic chunk D_i to a motion index κ of motion type Ω_K, i.e.

$$r_K : [C_{i-1}, D_i, C_{i+1}] \mapsto \kappa \qquad (1)$$

Based on the motion chunk structure, we detect human motions using machine learning techniques (Section 4.3), and evaluate them for improved motion training (Section 4.4).

4.2 Signal Segmentation

Signal segmentation divides the acquired signal into a sequence of motion chunks. In speech analysis, it is required to segment start and end of the human voice. Similarly, we also need to detect when a motion starts and ends within the motion signals. Our goal is to segment the acceleration signals for the static chunk and the dynamic chunk. We developed a simple segmentation method by measuring a standard deviation of the raw signal. First, we compute a standard deviation over 10 points of the raw signal. In a subsequent step, we calculate a second standard deviation over the 10 previously computed standard deviations. If the second standard deviation value is above a threshold, we assume that a motion starts. This strategy provides segmentation results which are robust against regularly-vibrated motions. In addition, we check the length of the segmented signal and eliminate too long and too short signals.

4.3 Motion Detection

Motion detection in our system distinguishes reference motions from the user's arbitrary motions. Whereas in the speech recognition, the major challenging problem is to extract human voice from the environmental noise, our application requires to detect a motion by eliminating the arbitrary motions from a long sequence of human motions. We employ HMMs to accomplish this.

4.3.1 Hidden Markov Models

HMMs have been applied extensively in speech recognition to determine written words from speech. A HMM is based on the assumption that the process can be described as a first-order Markov process and represented as a set of distinct states [10]. The change from one state to another is a stochastic process. The general idea of an HMM is that a sequence of hidden "states" can be inferred from the observed data. For example in speech recognition, the hidden states may represent words or phonemes and the observations represent the acoustic signals. In our motion detection, the motion chunk is represented by a sequence of hidden states, and motion signals are processed to generate each observation of the states respectively. A HMM of a set of states (S) is characterized by initial state distribution (π), transition probabilities (A) and item output probabilities (B). Thus, an HMM $lambda$ (figure 8) can be characterized by a set of parameters regarding two states S_1 and S_2:

$$\lambda = (\pi, A, B) \qquad (2)$$

where A is the transition matrix $A = \{a_{1,1}, a_{1,2}, a_{2,1}, a_{2,2}\}$, $\pi = \{\pi_1, \pi_2\}$ are the prior probabilities, and $B = \{b_1(O_n), b_2(O_n)\}$ are the observation probability distributions for each state given the observation $O_n = \{S_1, S_2\}$. Considering our purpose to detect a motion chunk, there are two main processes to use HMMs. Firstly, we create a HMM for each motion and adjust the model parameter using observation sequences maximizing the probability of the observation sequence denoted by $P(O_n|\lambda)$ given the HMM model. Second, we apply the new observation sequence to compute $P(O_n|\lambda)$ given the HMMs trained in the previous step. Comparing these probability values, we detect input motions.

4.3.2 Motion Chunk based HMMs

We apply the concept of a motion chunk to represent the hidden states. The topology of each motion is represented as two distinct states which can be regarded as a two-state machine. We use the start-static chunk and the end-static chunk only, because the in-between dynamic chunk features a highly variant signal heavily depending on speed and power. As illustrated in figure 6, individual signals posses a large variability even when the same motions are performed by the same user. However, the signals of the static chunks are stable enough to be used as an observation vector. Each HMM is created with the performance of two static postures (start and end). This usually takes below one minute, as explained in Section 6.3. Using the signals as the observation sequences, we train the HMM parameters of each motion. We employ an iterative procedure called Baum-Welch method [10] widely used to find a local maximum of the probability. Once the HMM model is trained, the system is able to detect the newly performed motion. For this, we employ the probability of the observations using a Viterbi algorithm [10]. If the probability is high enough given a HMM model of a motion type, we detect the input motion and generate a motion chunk with resampled data. The resampling is necessary to evaluate the quality of the motion. Then, the further processes, such as motion evaluation and motion training video generation, are executed. In our current training scenario, we usually detect only one motion type at a time from arbitrary motions. However, the system can also classify several motions by comparing their probabilities to each other (figure 9).

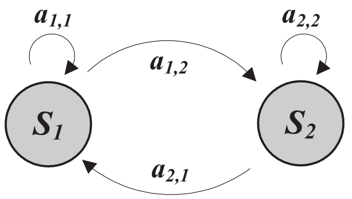

Figure 8: An example HMM topology of two states: S_1 (start-static chunk) and S_2 (end-static Chunk) , $a_{1,2}$and $a_{2,1}$ (transition matrix between $S_1$and S_2)

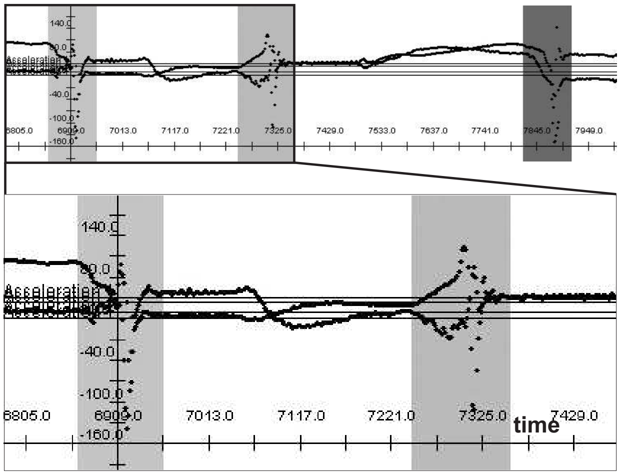

Figure 9: An example for motion detection for three different motions. Notice that each detected motion has relatively different start- and end-static chunks.

4.4 Motion Evaluation

We evaluate motions based on the quality of their motion chunks. While we use only static chunks for motion detection, we also take a dynamic chunk for evaluation. The evaluation of each motion chunk provides distinct scores for *start posture*, *gesture*, and *end posture*. The three scores are averaged for a score of one motion. The evaluation of two static chunks explains how the start and end postures have been performed respectively. The evaluation of the dynamic chunk tells how the gesture is performed with respect to power and speed. We use a Euclidean distance metric to measure the similarity of two motion chunks. During the motion detection process explained in Section 4.3, the motion chunk is generated and resampled to the resolution of the reference motion in the motion data base. This compensates for potential timing differences. We compare the actual motion chunk with 10 different reference motion chunks of a trainer stored in our database. We take the *minimum distance* as the final score. The scores are normalized to a

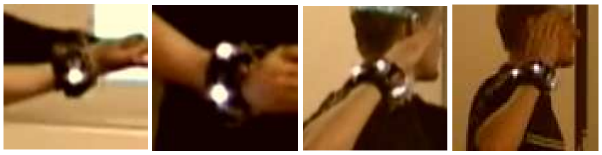

Figure 10: Observations for body sensor tracking with LED markers on the wrist body sensor.

maximum value of 100. During the tests, we found that this minimum distance is better suited than mean or median to measure quality. The computed scores are displayed on the motion training video panel in real-time.

5. MOTION TRAINING VIDEO

A motion training video is necessary for trainees and trainers as a reference to follow and analyze motions. However, producing such a video usually takes a lot of time. First, it requires simultaneous video recording during the trainer's performance. Also, the captured videos should be edited for the purpose of motion training such as selecting video frames and adding explanatory information. We provide a method for automatic generation of motion training videos. As soon as the input motion is detected, we save both the relevant video frames and the body sensor data. Then we generate a video displaying body sensor data along the tracked sensor positions, as illustrated in figure 11.

5.1 Body Sensor Tracking

We extract sensor positions from the captured images and use the positions to generate visual feedback. We made various experiments to find suitable tracking solution for our purpose. First, the color band tracking highly depends on the training environment condition such as lighting and color. We also tested IR light sources, but they omit color information which is required. We found that color LED markers are most suitable for our purpose. Their brightness provides relatively robust tracking results in indoor training environments. We developed a simple vision tracking algorithm to find the pixel positions within a certain color and brightness range. The number and position of LEDs are designed depending on the sensor position. In our tests, we attached four LEDs at the four sides of the wrist bend. This installation allows us to detect at least one point reliably even when the hand is rotated in different directions. Figure 10 illustrates four cases where one, two, three points are detected respectively. We use the center of the detected point as the position of the body sensor.

5.2 Visual Feedback

Visual feedback helps trainers and trainees to explain and improve their motion practice. During the user tests, we ascertain the fact that visual feedback for body sensor data is absolutely needed. Users wanted to see how the body sensor data is changing with the appearance of the posture. Especially for the trainees, visualizing motion path helps significantly to understand a dynamic gesture between two static postures. Thus we focus on visualizing body sensor data on the images along the motion path as illustrated in figure 11. We use the tracked sensor positions and design a simple template to display a moving circle along the path changing its size as a function of the magnitude of the acceleration. There are various design alternatives, of course, varying the shape and its transformation rules.

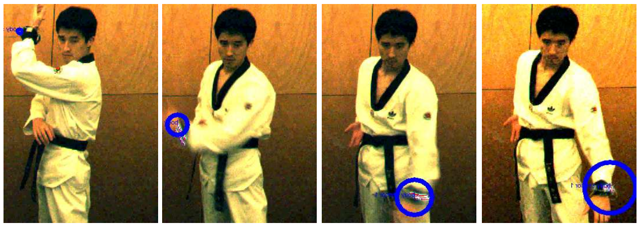

Figure 11: Example video frames for visual feedback with the mean power of acceleration signal.

6. USER EXPERIMENTS

We conducted a set of user experiments to quantify the costs and benefits of combining visual and body sensor data for motion training. We expect our motion training system to provide significant benefits over conventional motion training. In our motion training system, visual sensor data is used as a feedback to the user allowing him to coarsely adjust his motion to the reference motion. Conversely, the body sensor data is utilized for adjusting required body part of the user precisely. In our experiment we measure this benefit. In addition, we quantify how our system evaluates postures and gestures and how it detects human motions in real-time. To this end, we use a martial art training scenario. Martial art training is specifically suited for our experiment because it includes highly complex, precise motions which contain both postures and gestures.

6.1 Subjects

For this experiment, we used a trainer who is a master of Taekwondo and six additional subjects as trainees, three male and three female, all of them having no experience in martial art training.

6.2 Task

We designed the separate tasks for the trainer and for the trainees. The task of the trainer was to produce the reference motion data model for 10 sets of five motions each (punch, outside block, upper block, inside block, and down block). This model was used for the trainee experiment later on. Subsequently, the trainer was asked to perform 5 sets of 10 outside-blocks for testing the motion evaluation methods. The rest time between each set was two hours, and in each set he performed 10 times repeatedly without resting.

For the trainees, we designed two basic training conditions: posture training and gesture training. The task of posture training was to learn start and end postures of the five motions. The gesture training serves for practicing individual gestures between a start posture and an end posture. In posture training, the trainees were told to perform postures of four motions three times each while watching a reference image without resting. We measured how long it takes to learn to match their postures to the trainer's average roll and pitch values. Among the five motions, we selected the punch motion which is relatively easy for teaching novices to use the system. The end posture of the punch is also

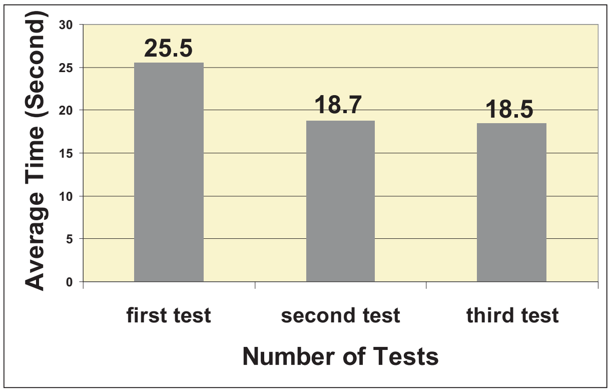

Figure 12: Experimental results of trainee subjects in posture training: average time (in seconds) to complete the task of posture training for four different motions. Note that the first and the last experiment use only a Boolean feedback indicating whether the current posture is correct or not. The second experiment displays a signal which trainees can follow.

employed as an initial posture for the trainee to start with. We designed a simple progress bar which provides Boolean feedback indicating whether the current posture is correct.

In gesture training, we evaluate how the trainees perform a single motion in rapid succession of the start posture, the gesture, and the end posture. Each trainee was told to do an outside block motion. Before this, they had to learn the start and end postures of the block motion. In our training scenario, we utilized trainer's reference data to train the HMM. Thus, trainees first have to learn the required two start and end postures so that their motion can be detected. This usually takes several minutes, as we found out during the user tests. Note that this process is most similar to the practical training and thus, it is not considered as an additional, unnecessary step to prepare the system. We evaluated each motion based on the trainers' data of the previous experiment. From the first posture training experiment, we found that the outside block is the most difficult one and is appropriate for testing the gesture training. Again, trainees were asked to perform 3 sets of 10 outside-blocks with approximately a two hours time interval between each set. All subjects were granted a minimum time to become familiar with the new training devices, the wireless accelerometer worn on the right wrist, and the video projection.

6.3 Results

We collected the trainees' performance data for each of the tasks. For the trainer's task, we found that the trainer completed nearly all tasks correctly. Thus, we could use the time to complete the task as an overall performance measure for potential trainers. It only took the trainer 10 minutes to create the 10 reference sets of the five motion data. From this, we obtained 50 motion training videos (10 for the five motions) containing both body and visual sensor data as well as visual feedback. The motion training video generation was performed very well with the help of automatic motion detection. Figure 13 shows the average scores after the trainer's gesture training. As mentioned in Section 4.4, each performed motion was scored against three parts: start posture, gesture, and end posture. The overall scores are provided by averaging the three scores. We can see how he improved after 5 sets of 10 outside blocks each. While the scores of the end postures are constant, the scores of the start postures and the gestures change slightly over time indicating the adaptation to the reference. After interviewing with the trainer, we realized that the end posture scores are slightly going down, because the trainer focused on the gesture while spending less effort on the postures. We could also infer that gesture training bears more potential for further improvement than static posture. Even though the trainer masters the postures, it is very difficult to keep the right

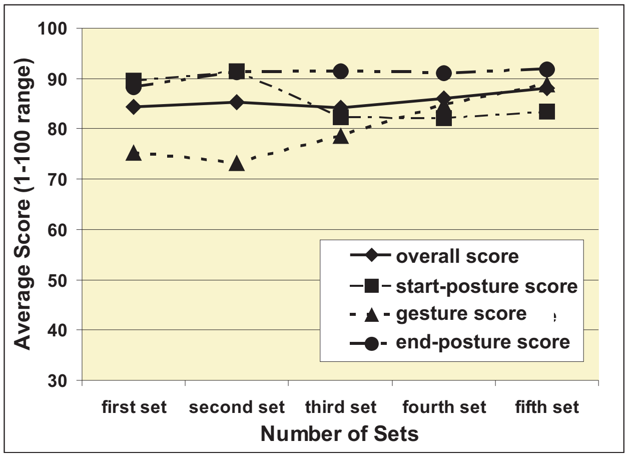

Figure 13: Experimental results of a trainer subject during gesture training: average scores of two postures and one gesture and their overall score after five sets of 10 outside blocks.

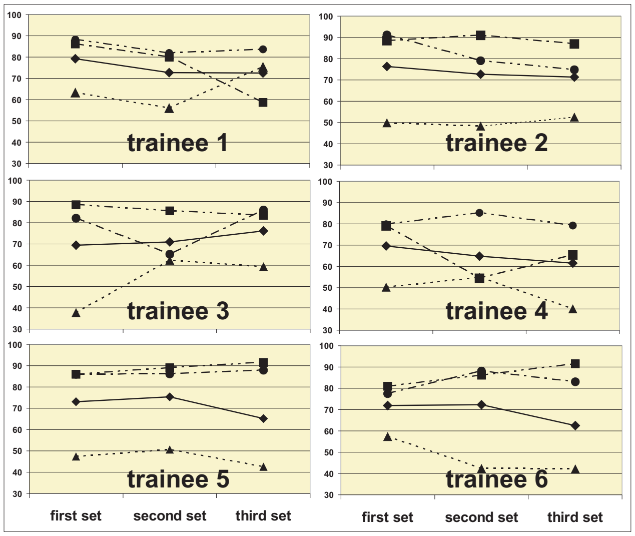

Figure 14: Experimental results of trainee subjects for gesture training: average scores of two postures and one gesture and their overall score after doing three sets of 10 outside blocks. The legend is the same as in figure 13

postures during the dynamic performance - a skill that distinguishes masters on the highest level. This shows that the resolution of our training system is sufficiently high for evaluations on highest levels and that it can be used to practice gestures with self-created reference data.

The result of the posture and gesture training clearly demonstrates that the system helps trainees learn complex martial art postures in a short time. As illustrated in figure 12, watching the body sensor signals helps the trainees to find the correct postures. After this experiment, we could also compare the individual postures and realized that some postures are relatively difficult to learn. We found that if the body sensor is further away from the trainee's body, it is more difficult to repeat a constant posture. The gesture training experiment of the six subjects yielded quite interesting result. Similar to the trainer's experiment, the static posture scores are higher on average than the gesture. Interestingly, we found that there were three different styles. First, trainee1 and trainee2 focused very much on improving gestures. As a result their start postures were getting worse over time. On the other hand, trainee4, trainee5 and trainee6 focused their attention to the improvement of their static postures rather than on the dynamic gestures. Finally, trainee3 improved both postures at the same time - which is clearly the desirable case. Even though trainees know the start and end postures, it was difficult for them to perform correctly in the dynamic setting. We also felt that the male subjects tend to use more power, whereas female subjects focus on technique. However, this finding did not influence the results significantly.

6.4 User Feedback

During the experiments, we collected interesting user reactions and received many comments. Some users felt that our training system can be useful for computer games related to sports and martial art sparring. They suggested that using real motions would make the interactions in playing computer games more appealing. Some participants were getting very involved in the training and all of them performed very seriously. As one participant commented: "The system helps me to focus my attention on precise my body movements". Most of people asked to use the system on a regular basis. Since we employ low cost technologies, the system can be easily tailored towards an individualized personal training system. We also let the members of Computer Graphics Laboratory, at ETH play with the system. In order to test long term training progress, we also repeated some experiments after a while. Although some users had lower initial scores compared to last time, they quickly caught up and made progress.

7. CONCLUSION AND FUTURE WORK

We presented an approach to build a motion training system combining body and visual sensor data. We described a novel motion decomposition procedure called motion chunk for real-time motion analysis. Based on the motion chunk, we detect and evaluate a specific motion using Hidden Markov Models. We also presented an automatic video editing method to generate a motion training video including visual feedback. During a series of user experiments, we demonstrated how our motion training system can be used for the practical training. We also tested the motion detection, evaluation and motion training video generation in real-time. The system helps both trainers and trainees to improve fine static postures and dynamic gestures. So far, our research has mainly focused on analyzing single motions. Future work will be devoted to the analysis of longer sequences of motions. From this, users can eventually practice combinations of multiple motions.

8. ACKNOWLEDGMENTS

We thank all participants in the user experiments and the members of Computer Graphics Laboratory, ETH Zurich. This work is carried out in the context of the blue-c-II project, funded by ETH grant No. 0-21020-04 as an internal poly-project.

9. REFERENCES

[1] J. K. Aggarwal and S. Park. Human motion: Modeling and recognition of actions and interactions. In *3D Data Processing, Visualization, and Transmission, 2nd International Symposium on (3DPVT'04)*, pages 640–647, Thessaloniki, Greece, September 06 - 09 2004.

[2] S. Baek, S. Lee, and G. Kim. Motion retargeting and evaluation for vr-based training of free motions. *The Visual Computer*, 19(4):222–242, July 2003.

[3] L. Bao and S. Intille. Activity recognition from user-annotated acceleration data. In *Proceedings of Pervasive 2004*, volume LNCS 3001, pages 1–17, 2004.

[4] D. Becker. Sensei: A real-time recognition, feedback, and training system for t'ai chi gestures. Master's thesis, Massachusetts Institute of Technology, 1997.

[5] G. S. Chambers, S. Venkatesh, G. West, and H. Bui. Hierarchical recognition of intentional human gestures for sports video annotation. In *Proceedings of ICPR02*, pages 1082–1085, Thessaloniki, Greece, September 2002.

[6] C. Chua, N. H. Daly, V. Schaaf, and H. P. Camill. Training for physical tasks in virtual environments: Tai chi. In *Proceedings of IEEE Virtual Reality 2003 Conference*, pages 87–94, Los Angeles, Califonia, March 2003. IEEE Computer Society.

[7] J. W. Davis and A. F. Bobick. Virtual pat: A virtual personal aerobics trainer. In *Proceedings of Workshop on Perceptual User Interfaces*, pages 13–18, November 1998.

[8] D. Estrin, D. Culler, K. Pister, and G. Sukhatme. Connecting the physical world with pervasive networks. *IEEE Pervasive Computing*, 1(1):59–65, 2002.

[9] M. Gross, S. Wuermlin, M. Naef, E. Lamboraz, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. V. Gool, S. Lang, K. Strehlke, A. V. Moere, and O. Staadt. blue-c: A spatially immersive display and 3d video portal for telepresence. In *Proceedings of ACM SIGGRAPH 2003*, pages 819–827, 2003.

[10] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286, Feburuary 1989.

[11] X. Sha, G. Iachello, S. Dow, Y. Serita, T. S. Julien, and J. Fistre. Continuous sensing of gesture for control of audio-visual media. In *Proceedings of ISWC 2003*, 2003.

[12] T. Starner and A. Pentland. *Visual Recognition of American Sign Language Using Hidden Markov Models.* M.I.T. Media Laboratory, Cambridge MA.

[13] M. Takahata, K. Shiraki, Y. Sakane, and Y. Takebayashi. Sound feedback for powerful karate training. In *Proceedings of International Conference on New Interfaces for Musical Expression (NIME04)*, 2004.

[14] U. Yang. Just follow me: An immersive vr-based motion training system. In *Proceedings of International Conference on Virtual Systems and Multimedia*, 1999.