ABSTRACT

In this paper, we propose a human-marionette interaction system based on a human action recognition approach for applications to interactive artistic puppetry and a mimicking-marionette game. We developed an intelligent marionette called “i-marionette” that is controlled by a sophisticated control device to achieve various human actions. Moreover, we utilized an action recognition approach to enable the i-marionette to learn and recognize complex dance movements. The idea of artistic puppetry is to present a conflict scenario between two different cultural worlds: the performer is active and represents the culture of modern technology based in the real world. In contrast, the i-marionette represents traditional culture and is passive and based in a virtual world. The active performer guides the passive i-marionette to form a space-time connection between the real world and the virtual world. The i-marionette mimics the performer’s action, while the performer also mimics the i-marionette’s action. The performance represents an artistic conception in which humans invent technology and the i-marionette is manipulated by human control. However, in this interactive circle, the human is implicitly affected by the i-marionette. In our mimicking-marionette game, a player mimics the i-marionette’s action. Subsequently, our human action recognition system measures the action similarity between the player and the i-marionette, and our system provides a similarity score.

Categories and Subject Descriptors

I.2.1 [Computing Methodologies]: Artificial Intelligence-Applications and Expert Systems; I.5.4 [Computing Methodologies]: Pattern Recognition-Applications; J.5 [Computer Applications]: Arts and Humanities

Keywords

Human action recognition, interactive art, interactive game.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MM’12, October 29–November 2, 2012, Nara, Japan.
Copyright 2012 ACM 978-1-4503-1089-5/12/10...$15.00.
performer (puppeteer) implies modern technology, and the i-marionette implies traditional culture. Through the interactions between the i-marionette and the puppeteer, puppetry represents an interactive circle: humans invent technology, the technology controls the marionette, and the marionette influences humans as well. Figure 1 shows an interaction between the performer (puppeteer) and our i-marionette. Although the outer appearance is that the i-marionette is manipulated by the performer, the performer is implicitly affected by the i-marionette.

In our mimicking-marionette game, the i-marionette performs a contemporary dance for the player to mimic by following its actions. Our action recognition system measures the action similarity between the player and the i-marionette. At the end of the game, the system demonstrates the measured similarity as a score. If the score is high, the actions of the player and the marionette are similar. If the score is low, the actions of the player and the marionette are not similar. The mimicking-marionette game is not only for entertainment purposes. The major motivation is to make the player experience being affected by the marionette. When the player obtains a high similarity score, the player appears to win this game. However, the player has been influenced by the marionette as well. That is to say, humans develop the mimicking-marionette game, a human plays the game, and the game influences the human. In the end, the human was manipulated by the game.

Figure 2. Chinese Liyuan Dance

To recognize and learn complex and sophisticated human actions, we present an approach called the Action Trait Code (ATC) [15] for human action classification. Our approach involves a Microsoft Kinect sensor and derived 3D body joints obtained with the Kinect SDK. ATC represents an action with a set of velocity types derived from the average velocity of each body part. An effective graph model based on ATC classification is employed for learning and recognizing human actions. To examine the recognition accuracy, we evaluate our approach on a self-collected database containing six dance videos. Each dance video is composed of several Chinese Liyuan basic dance steps (Figure 2). The experimental results show that the proposed approach not only achieves high recognition accuracy but also operates in real time.

The rest of this paper is organized as follows: Section 2 briefly introduces the related works. Section 3 specifies the proposed action recognition approach. The i-marionette and its control device are described in Section 4. Section 5 describes the interactive artistic puppetry and its artistic motivation. The mimicking-marionette game is presented in Section 6. Conclusions are drawn in Section 7.

Figure 3. Robot dance: (a) Honda Asimo is capable of learning and mimicking human actions; (b) HRP-2 reproduced the Japanese folk dance with a geisha. Although these robots can perform human actions, they are limited by a predefined program.

2. RELATED WORK

The marionette is a sophisticated mechanical system, often containing more than 40 degrees of freedom. Some researchers have attempted to create an automatic marionette for puppetry art. Murphey et al. [1][2] propose a motion description language (MDLp) for specifying and encoding autonomous puppetry plays. Sillam and Luciani [14] create puppetry art by building a physical puppet model and connecting the strings to handles connected to piano keys. The puppet is triggered by pressing a piano key.

In addition to studies attempting to create electronic marionettes, we are also interested in the study of human-robot interaction. Many researchers have studied human-robot interaction in the context of dance. The Honda humanoid robot called Aismo [12] has been developed to learn and mimic a person’s movement (Figure 3a). HRP-2 [9] is a humanoid dancing robot designed by Japanese researchers that is programmed to reproduce the Japanese folk dance with a geisha (Figure 3b). HRP-4C [10], fembot (female robot), moves like a human, mimics human facial movements, and executes dance steps. However, HRP-4C is also programmed by a choreographed program. In the above studies, even if the robots successfully performed the complex movements, such as dancing, they are limited by the predefined program.

Human action recognition and categorization have been attracting much attention over the past few decades due to their applicability to many areas, including human-computer interaction and game design. However, the recognition task involves many challenging problems. According to a comprehensive survey [5], many previous studies on action recognition have concentrated on using 2D videos [3][5] or still images [7]. However, those approaches are limited to expressing lateral motion only. Recently, 3D body joint locations have been widely used in human action recognition tasks [4][6][8] because they provide more explicit information for describing human movement. Sung et al. [6] propose a hierarchical maximum entropy Markov model (MEMM) using 3D skeleton data to perform human detection and recognition of unstructured human activity in an unstructured environment. Raptis et al. [8] propose a real-time dance gestures classification system, which is composed of an angular representation of the skeleton designed, a cascaded correlation-based classifier, and a

3. Human Action Recognition

To recognize complex human activity, such as contemporary dance, we propose a robust action recognition approach to address this problem. In this paper, we divide the human action understanding task into two processes. One process is a classical action recognition task in which we employ a probabilistic model for learning and recognizing human actions. The second process is an action categorization task in which we classify actions based on quantized human movement.

3.1 Action Recognition Overview

Our proposed action recognition system comprises three major parts, including preprocessing, action classification, and action recognition. A detailed flowchart is shown in Figure 4.

During preprocessing, our system extracts the 3D body joint sequence from a human action. For classifying human action, we employ an approach called Action Trait Code (ATC) [15] for human action classification in the action classification step. ATC represents an action with a set of velocity types derived by the average velocity of each body part.

The action recognition part introduces an effective graph model based on an ATC classification, which is then employed for learning and recognizing human actions.

3.2 Action Trait Code

An Action Trait Element represents the average velocity of a specific body part using the mean distance of corresponding joints of the 3D skeleton in a pose sequence (a human action).

We divide the human body into $N$ body parts. Let $\Psi = \{\varphi_1, \varphi_2, \ldots, \varphi_N\}$ be a velocity detector set, where $\varphi_i$ denotes the $i$th velocity detector. Given a body part $J^j$ with a set of corresponding $L$ joints, $\{j_1^j, \ldots, j_L^j\}$. The average velocity of $J^j$ can be obtained by

$$\varphi_i(J^j) = \frac{1}{T} \sum_{t=0}^{T-1} \sum_{k=1}^{L} \text{dist}(j_{k,t}^j, j_{k,t+1}^j) / T$$

where $T$ denotes the length of the input action sequence and $\text{dist}(\cdot)$ is a distance function that is formulated with the Euclidean distance.

A velocity discriminator is employed for quantifying each body part’s average velocity. If the number of action training data is $q$, then the velocity discriminator uses the k-means algorithm (Figure 5) to divide the $q$ average velocity values into $k$ clusters. In other words, the range of ATE is from 0 to $k$. Therefore, each average velocity of $J^j$ is tagged with a number as a symbol for the Action Trait Element.
3.3 ATC Encoding based on ATC

An ATC is like a specific body movement descriptor of an action. An ATC is constructed by a combination of ATEs. Each ATE is represented by values ranging from 0 to \( k \). Therefore, each action can be encoded to an individual ATC.

If we use a length \( N \) for the ATC (using \( N \) body parts) and each ATE is divided into \( k \) levels, then the ATC codebook represents \( k^N \) action types. Thus, the action retrieval process using ATC classification reduces the computational cost by \( k^N \) times. Furthermore, to prevent the over-fitting problem, we employ the k-means algorithm to cluster similar ATCs by calculating the L2 distance between two ATCs.

3.4 Action Classification based on ATC

In our system, we categorize actions by ATCs. In other words, an action database is grouped into several small action databases. Our action classification task comprises two steps. First, we precategorize actions by ATCs. In other words, because ATC is encoded using each body part's movement, various types of actions might be categorized in the same database. Therefore, we use a graphical model for dealing with the recognition process.

3.5 Graphical Modeling based on ATC

The Action Graph [3][4] is a useful graphic model approach to represent a dynamic human motion with a set of salient poses (Figure 6). The salient poses are shared among the various actions. The proposed modified action graph \( G_c \) from the database of action class \( c \) that encodes \( L \) actions with \( M \) salient postures \( \Omega_c = \{ \omega_{c,1}, \omega_{c,2}, \ldots, \omega_{c,M} \} \) can be represented as

\[
G_c = \{ \Omega_c, \Lambda_{c,1}, \ldots, \Lambda_{c,L} \} \tag{2}
\]

where each pose represents a node and \( \Lambda_{c,l} = \{ p(\omega_{c,j} | \omega_{c,i}, \Lambda_{c,l}) \}_{i,j=1}^{L} \) denotes the transitional probability matrix of the \( l \)-th action \( \Lambda_{c,l} \) in an action dataset \( c \).

According to the graphical interpretation, we describe the recognition system using a quadruple:

\[
\Phi_c = (\Lambda_{c}, \Omega_{c}, \Upsilon_{c}, G_{c}) \tag{3}
\]

where \( \Upsilon = \{ p(x_{c} | \omega_{c,1}), p(x_{c} | \omega_{c,2}), \ldots, p(x_{c} | \omega_{c,M}) \} \). In the action modeling process, \( \Phi_c \) involves three major steps: (1) extract salient postures \( \Omega_c \) from the training data, (2) model each posture by likelihood functions \( \Upsilon_c \), and (3) construct the action graph \( G_c \). Each salient pose is a set of similar poses, which is obtained by clustering the sample poses into \( M \) clusters. We cluster these poses into \( M \) salient postures by the k-means algorithm. We assume that the distribution of the points for a posture can be approximated by statistically independent. The posture model can be represented as the joint distribution of the points.

\[
p(\omega_{c} | \alpha_{c}) = \prod_{i=1}^{n} \sum_{j=1}^{c} \pi_{j,\omega_{c}} N(p)| \mu_{j,\omega_{c}}, \Sigma_{j,\omega_{c}} \tag{4}
\]

where \( N(.) \) is a Gaussian function. In the recognition task, we present a modified action graph based on ATC to recognize human action.

The action recognition process is to seek the most likely action \( \hat{\lambda} \) from a set of action models \( \Lambda = \{ \Lambda_1, \Lambda_2, \ldots, \Lambda_L \} \). Let \( X = \{ x_1, x_2, \ldots, x_T \} \) be a set of pose sequences derived by the input action sequence with \( T \) frames. The action recognition process can be formulated as

\[
\hat{\lambda} = \arg \max_{\lambda} p(\lambda, X, \lambda)
\]

\[
= \arg \max_{\lambda} p(x_1, x_2, \ldots, x_T | s_{c_1}, s_{c_2}, \ldots, s_{c_T}, \lambda_c) p(c_i)
\]

\[
= \arg \max_{\lambda} p(x_{c_1}, \ldots, x_{c_T} | s_{c_1}, \ldots, s_{c_T}, \lambda_c, c_i) \times p(s_{c_1}, \ldots, s_{c_T}, \lambda_c) p(c_i)
\]

where \( s_{ci} = \{ s_{ci,1}, \ldots, s_{ci,n} \} \) represents the corresponding posture sequence derived from \( X \) and \( p(c_i) \) is a prior value based on the confidence of ATC classification. We here assume \( p(c_i) = 1 \).

Assume that \( x_{c_i,t} \) statistically depends only on \( s_{c_i,t} \), \( x_{c_i,t} \) is statistically independent of \( \lambda_c \), given by \( s_{c_i,t} \), and \( s_{c_i,t} \) only depends on its previous state \( s_{c_i,t-1} \).

**Table 1. Chinese Liyuan Basic Step database.**

- **Type 1**: Walk Step + Hand Lift + Arrange Hat
- **Type 2**: Hand Lift + Foot Stamp + Field Across
- **Type 3**: Wave Over Elbow + Lift hand + Seven Step Jolt
- **Type 4**: Puppet Fall
- **Type 5**: Xiang-Gong-Mo + Hand Swing + Bolting Pose
- **Type 6**: Etitude + Hop
Thus, we can reformulate Eq. (2) as

$$\hat{\lambda}^* = \arg\max_{\lambda \in \Lambda, S \in \Omega} \prod_{i=1}^{N} p(x_{c_i}, t | s_{c_i}, c_i) \times p(\lambda_{c_i}, c_i)$$

(6)

where $p(x_{c_i}, t | s_{c_i}, c_i)$ expresses the probability of observation $x_{c_i, t}$, which is derived from salient posture $s_{c_i, t}$.

### 3.6 Evaluation of Chinese Liyuan Basic Steps

We evaluate our approach on a self-collected action database: the “Chinese Liyuan Basic Step dataset.” It is derived from the "Pear Orchard opera," a mixture of songs, dances, and Chinese Nanyin music. It was performed at the imperial court during the Tang Dynasty. The dance step of the Chinese Liyuan dance involves complex movements. We use six dance types for evaluating our action recognition approach. Each dance type is a combination of 1~3 basic dance steps, and each basic dance step is a series of complex movements. Table 1 shows the details of each collected dance data type. The experimental result shows that our approach achieves 100% recognition accuracy. In addition, Table 2 demonstrates the computational time for each dance type. As shown in the table 2, even though the length of each action in our database is longer than 9 seconds, our system successfully achieves the recognition task very quick which enable the i-marionette to interact with humans.

The reason that our approach achieves 100% recognition accuracy is because the discrepancy of each dance data is highly dissimilar, even if the dance movement is complex. The video of each self-collected dance data is shown in the attached webpage: [http://www.csie.ntu.edu.tw/~d97944010/research/mm2012/](http://www.csie.ntu.edu.tw/~d97944010/research/mm2012/)

Moreover, we had also evaluated our approach on a more complex action database, e.g., street dance database. As shown in [http://www.csie.ntu.edu.tw/~d97944010/research/icpr2012/](http://www.csie.ntu.edu.tw/~d97944010/research/icpr2012/), the proposed approach not only successfully achieves high recognition accuracy (97%) but also performs in nearly real-time.

### 4. i-MARIONETTE

The marionette is a sophisticated and challenging mechanical system. We aimed at the creation of a fully automated marionette, which is capable of performing various human actions. In this section, we introduce our performance stage, our intelligent marionette called the “i-marionette,” and its control equipment.

#### 4.1 i-Marionette

Our i-marionette has a humanoid body shape (Figure 7a) and a realistic female face (Figure 7b). The entire body of the marionette is composed of fifty body parts, consisting of a head, a torso, a pelvis, two upper arms, two lower arms, two hands, two thighs, two legs, and two feet. To construct the appearance of our i-marionette, we use a commercial mannequin for the basis of the i-marionette’s body shape and set the hook in each body part. Each hook position is used to truss the string for suspending each body part. Figure 8 illustrates the hook positions of the i-marionette. In addition, because each body part is hollow, we employ trestles for placing the joints in hollow body parts. We
also limit each joint’s movement by referring to the joint’s kinematic limitation. Figure 9 shows the joint setting in the hollow body parts. To construct humanoid movement for the i-maronette, we recorded an enormous human motion database including a huge number of 3D human motion data derived by the Kinect SDK and captured by a Kinect sensor. The 3D human motion data drives the i-maronette to perform various human actions.

4.2 Performance Stage and Control Device

Figure 10 shows the performance stage with the i-maronette and its control device. Along the stage, we placed three Kinect sensors to form a triangle area to sense the triggered action from the performer.

In the control device (Figure 11), we use a set of trusses to sustain 29 motors and the i-maronette. These motors control the i-maronette by dragging strings suspended on the i-maronette’s body. We divide human body motion into two parts: global motion and local motion. Global motion refers to the translation and the rotation of the entire body. In addition, the local motion denotes the body joint angle for each limb. For the above reasons, the architecture of our control device is designed by two layers of motor equipment (Figure 12). The first layer controls the global motion of the i-maronette, and the second layer manipulates the local motion. To send a package of commands to all of the motors, the motors are connected through RS485 wire. The layout of the motor controllers is shown in Figure 13.

5. INTERACTIVE ARTISTIC PUPPETRY

In traditional puppetry, the marionette is manipulated by a puppeteer. As shown in Figure 14a[16], there has been a conflicting relation between the marionette and puppeteer.

Figure 14a[16], there has been a conflicting relation between the marionette and puppeteer.
Figure 15. The two roles in our interactive artistic puppetry: (a) a puppeteer who implies modern technology; (b) a marionette that implies traditional culture.

Figure 16. The marionette, implying traditional culture, arrives from a virtual world.

The marionette is like a follower who is passively dominated by the puppeteer. In this respect, the puppeteer is an active guide who controls the marionette to act the way the puppeteer wants. While the puppeteer manipulates the marionette, he looks like a dominator. However, the puppeteer is also influenced by the marionette. When the puppeteer tries to manipulate the marionette well, the marionette causes the puppeteer to pay more attention to it. Therefore, the puppeteer becomes a “human marionette” (Figure 14b) who is influenced by a “marionette puppeteer”. We were inspired by the above observations, and we started to think about the following issues: what if the marionette becomes fully automatic? What if the marionette becomes an active manipulator and guides the puppeteer? Is the puppeteer influenced by the marionette?

This artwork attempts to explore the conflict relationship between the puppeteer and the marionette. Our interactive artistic puppetry is performed by two characters. The first one is a performer (puppeteer), implying modern technology in the real world (Figure 15 a), and the second one is the i-marionette (Figure 15 b), implying traditional culture in a virtual world. As shown in Figure 15, the performer wears a modern technology dress, and the i-marionette wears traditional clothing. Through the proposed action recognition approach, the marionette can recognize human behavior in real time. In the beginning, the i-marionette, which is coming from the virtual world (traditional culture), arrives in the real world (modern technology), see Figure 16. The background music is a traditional opera (the Chinese Liyuan opera), which is represented by the marionette. Subsequently, the puppeteer (performer) arrives, and then she walks up to the marionette and puts a spell on the marionette, see Figure 17a.

Figure 17. The marionette was dominated by the puppeteer: (a) the marionette was gradually losing its consciousness; (b) the marionette begins to follow the puppeteer’s action.

Figure 18. The puppeteer is mimicking the marionette’s action

The marionette gradually loses its consciousness. Then, the puppeteer attempts to force the marionette to mimic her actions, see Figure 17b. Surprisingly, the marionette unconsciously follows the puppeteer’s action. After a while, the marionette wakes up and shakes off the puppeteer’s control. Meanwhile, as a role-swap, the puppeteer begins to mimic the marionette’s action (Figure 18), which means that the puppeteer is affected by the marionette. Through the interactions between the marionette and the puppeteer, puppetry represents an interactive circle: humans invent technology, the technology controls the marionette, and the marionette influences the human.
Figure 19. At the end of the puppetry show, the puppeteer sends the marionette back to the virtual world.

Figure 20. Interface of the mimicking-marionette game.

Figure 21. Mimicking-marionette game: a player was mimicking the i-marionette’s action.

Figure 22. Similarity score result.

Figure 19 shows that the puppeteer sending the marionette back to the virtual world at the end of the puppetry show.

6. MIMICKING-MARIONETTE GAME
The mimicking-marionette is a role swap game (Figure 20) that transforms the i-marionette into a puppeteer and the player into a marionette. In the game, the i-marionette performs a contemporary dance for the player to mimic and follow its actions. The game interface is shown in Figure 21. The i-marionette in the small window is used to help the players modify their postures. Figure 22 shows the window of the measured result of the action similarity. The similarity result is represented as a score in the game. The system demonstrates the measured similarity as a score. A high score indicates that the actions of the player and the i-marionette are similar. In contrast, a lower score indicates that the actions of the player and the i-marionette are not similar. The major motivation of the mimicking-marionette game is not only entertainment but also to allow the player to experience the sensation of being affected by the marionette. As the player earns a high similarity score, the player appears to be the winner of the game. However, the player has been affected by the marionette. That is to say, humans develop the mimicking-marionette game, humans play the game, and the game influences the humans. In the end, the human was manipulated by the game.

7. CONCLUSION
This paper presents a human-marionette interaction system based on human action recognition. We applied our human-marionette interactive system for two applications: interactive artistic puppetry and a mimicking-marionette game. The main idea behind these two applications is to explore the peculiar relationship between the puppeteer and the marionette. The interactive artistic puppetry presents a conflict scenario between two different cultural worlds: The performer implies modern technology based in the real world. In contrast, the i-marionette implies traditional culture based in a virtual world. Through the interactions between the marionette and the puppeteer, puppetry represents an interactive circle: humans invent technology, the technology controls the marionette, and the marionette affects the human. In the other application, the mimicking-marionette game, which is similar to a role-swap game, the player pretends to be a marionette and experiences the sensation of being affected by the marionette. As the player gains a high similarity score, the player appears to be the winner of the game. However, the player has been manipulated by the marionette. That is to say, humans develop the mimicking-marionette game, a human plays the game, and the game influences the human. In the end, the human was manipulated by the game. To recognize and learn complex and sophisticated human actions, we present an approach, Action Trait Code, for human action classification and employ an
effective graph model based on ATC classification for learning and recognizing human actions.

8. ACKNOWLEDGEMENTS
This work was supported in part by the National Science Council, Taiwan, under grant NSC 101-2622-E-002-005-CC2. We would like to thank the colleagues in the Digital Art Center, Taipei, including Wen-Hao Huang and Ruey-Ning Shih, for their technical supports in this work. Special thanks go to the dancer Ho-Wen Hsiao, for her outstanding performance.

9. REFERENCES
[16] Creative Commons Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0).