

Searching 3D Motion Patterns of Vietnamese Traditional Dances

Le Thi Thanh Van¹, Vu Ngoc Quang¹, Pham Thanh Huyen², Ma Thi Chau¹, Le Thanh Ha¹

¹ HMI Laboratory, University of Engineering and Technology, Vietnam National University, Hanoi, Vietnam

² Faculty of Information technology, Ha Long University, Quang Ninh, Vietnam

Correspondence: Le Thi Thanh Van, lvank52thb@live.com

Date received: 23/09/2021, Date revised: 29/10/2021, Date accepted: 15/11/2021

Digital Object Identifier: 10.32913/mic-ict-research-vn.v2021.n2.1020

Abstract: Vietnam has many traditional dances in old theatres such as Xoan singing, “tuồng” or “chèo”. They all urgently need to be preserved in digital formats, especially in 3D motion capture format for dances. In digital formats, they bring many values such as the ability to automatically classify and search for content of dances’ movement. In this paper, we propose a system for 3D movement search of Cheo dance’s postures and gestures. The system applies sliding window technique, Dynamic Time Warping algorithm and a novel feature selection method named CheoAngle. Results show that the proposed system reach good scores in several metrics. We also compare CheoAngle with other feature selection methods for 3D movement and show that CheoAngle give the best results.

Keywords: *Motion search, motion recognition, similarity matching, dynamic time warping*

Title: **Tìm kiếm mẫu chuyển động 3D của múa truyền thống Việt Nam**

Abstract: Việt Nam có nhiều thể loại múa truyền thống trong nghệ thuật sân khấu cổ truyền như hát Xoan, tuồng, chèo. Tất cả những hình thức nghệ thuật này đều cần được bảo tồn ở dạng số hóa, đặc biệt là dạng chuyển động 3D cho múa. Trong dạng số hóa, chúng có nhiều giá trị như khả năng phân loại tự động và tìm kiếm nội dung chuyển động múa. Trong bài nghiên cứu này, chúng tôi đề xuất hệ thống tìm kiếm chuyển động 3D cho các tư thế và động tác của múa Chèo. Hệ thống áp dụng kỹ thuật sliding window, thuật toán Dynamic Time Warping và phương pháp chọn đặc trưng mới (tên là CheoAngle). Những kết quả thực nghiệm cho thấy hệ thống đề xuất cho kết quả tốt ở một vài phép đo. Chúng tôi cũng so sánh CheoAngle với những phương pháp chọn đặc trưng khác cho chuyển động 3D và chỉ ra rằng CheoAngle cho kết quả tốt nhất.

Keywords: *Tìm kiếm chuyển động, nhận diện chuyển động, so sánh tương đồng, dynamic time warping*

I. INTRODUCTION

Vietnamese traditional dance is one of basic types in Vietnamese long-established theatres [1–3]. They combined many factors, which consists of music, songs, art and literary screenplay, in order to represent a character’s personalities, psychological states... “chèo” (Cheo dance) is a particular-popular traditional dance form whose postures and motions mimicked life’s activities, plants and animals [1].

Need of digitalizing 3D motion for the Vietnamese traditional dance is one of necessary tasks. The traditional dances in 3D format help researchers to analyze and check basic postures (each posture is an unmoving position of body parts at a time (see Figure 1, 2 and 3) at different views. Thus, teaching dance also become more easily and feasibly while combining 3D technology. However, until



Figure 1. Arm cluster in Cheo dance. From left to right, 5 basic postures include: “múa chạy đàn”, “dâng hoa”, “dâng rượu”, “múa cướp bông”, “tấu nhạc”. Source: disanso.vn

now, we do not have any system which supports to manage 3D dataset of the traditional dance efficiently. To manage the data, the system must have operations such as search, edition, insertion and deletion. Accordingly, search is one of the most important operations. Indeed, this paper suggests a way to store 3D data of the traditional dance, define 3D basic movements of the traditional dance and build



Figure 2. Leg cluster in Cheo dance. From left to right, 5 basic postures include: “chữ V”, “chữ chi”, “quả trám”, “chữ đinh”, “đệm gót”. Source: disanso.vn

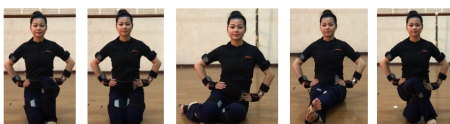


Figure 3. Sit cluster in Cheo dance. From left to right, 5 basic postures include: “chân chống chân quý”, “hai đầu gối cùng quý”, “ngồi hai chân co về một bên”, “đuôi thẳng hai chân”, “hai chân bắt chéo”. Source: disanso.vn

an algorithm for searching 3D movement of traditional dances in order to support making a system which manage traditional dance data in 3D space.

Cheo dance has the most systematic teaching [4] in types of Vietnamese traditional dance. It divides postures into 3 clusters: arm (see Figure 1), leg (see Figure 2) and sit (see Figure 3). Each cluster has 5 basic postures and their variants. For example, in the posture “dâng hoa” [4], both arms put on two body sides and move the same direction. Accordingly, any motion satisfying these two criterions will belong to the posture “dâng hoa” (see Figure 4). This motion is called as a variant of the posture “dâng hoa”. However, each posture has several special gestures which often mimic daily activities. For instance, gesture “gánh” of posture “tấu nhạc” [4] imitates an action “carry rice” of farmers. This gesture requires a performer to raise up one hand (at shoulder-equal position) and put down the other while arms and legs are swinging up and down. Additionally, combination of 3 clusters is in need of attention. At the same time, a performer only can use one leg posture or one sit posture. Unlike leg and sit postures, one arm posture can combine any leg or sit posture concurrently. To simplify, the leg and sit cluster is grouped into the same leg-sit cluster. It is worth noting that Cheo dance has basic postures (or movements that position of body features is changed from time to time) more complex than other traditional dances. For instance, in Hat Xoan Phu Tho songs [3], the palm moves in/out or ups/downs; the leg moves horizontally or around.

From above observations, in this paper, we have chosen 3D dataset of Cheo dance [5] as a representative in order to build a proposal algorithm. It was captured from an Xsens MVN motion capture system (a product of Xsens

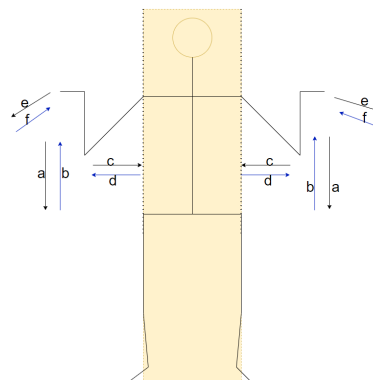


Figure 4. Variants of posture “dâng hoa”. A variant is a motion of two arms in the same direction – namely a, b, c, d, e, f – at two body sides in the extent which is the non-yellow region. The same motion is displayed by the same arrow color (black or blue).

Technologies B.V.) at format BVH (biovision hierarchy animation file). This file stores 2 parts: hierarchy of joints of a human skeleton and their 3D coordinates’ joints of the skeleton in each frame. Here, BVH file displays each frame as one time and uniquely interprets motion of one performer.

Problem of dance detection and recognition at 3D space has been studied in a long time. These works constructed algorithms for little gestures [6–13]. In fact, dance genres have no much gestures. So that, designing new algorithms for gestures recognition becomes more easily. For instance, Björn Rennhak and colleagues suggested a technique “turning motions” [6] which would detect poses in order to interpolate trajectory from state to state (in skills of Learning from Observation). This technique was applied for a traditional Japanese folk dance which only consists of 8 simple gestures which were captured by Vicon Motion iQ System equipment with 38 markers. The technique “turning motions” found out intersection points of forearms after projecting them orthogonally from 3D to 2D space. In spite of focusing parts of causing motion, this method was not likely to classify for same-position motions. Moreover, deep learning approach is also applied for dance dataset [8, 9, 12, 13]. This approach tends to 2 branches: 3D reconstruction from 2D images/videos, 3D dataset. However, to use deep learning algorithms, researchers must collect a huge number of samples for training process. Indeed, this paper suggests a way to store 3D data of the traditional dance, define 3D basic movement of traditional dances in order to support making a system which manages traditional dance dataset in 3D space.

II. RELATED WORKS AND METHODS

1. Problem

Problem of searching 3D motion patterns of Vietnamese traditional dances must find out each pattern's start point and end point and classify patterns in 3D space. Searching these points can be solved by many techniques. However, sliding window technique (see II.3.a) is likely to scan almost appearance of patterns. So that, we are interested in the classification step. In fact, pattern classification in 3D motion is considered as a problem of comparing similarity level between two motions. This problem is defined as below:

A given pattern set $A = \{A_1, A_2, \dots, A_n\}$, where A_i is a pattern, and a movement data X . The problem requires to search patterns A_i in X .

Here, a pattern is a basic posture or its certain variant at 3D space while a movement data can be a dance or compound of several postures or variants in 3D space.

2. Related works

In 3D movement, motions are analyzed and stored as human skeleton joints in the three-dimensional space in time order. A pattern is a 3D motion segment which shows a complete meaning action. To search for patterns in 3D movements, we need to detect and classify patterns.

a) Pattern detection

Patterns detection is an important step of searching patterns in time series. Approach Time series segmentation [14] has already used in a long time due to its superior features. Three main approaches consist of sliding window, top-down, bottom-up. The result of sliding window is segments of data which can contain patterns for the pattern classification step. We call these segments are proposals.

b) Pattern classification

Patterns classification is to compare predefined patterns to a proposal and choose the pattern which matches the best with the proposal. To do this, we need a technique to compare similarity between them. According to Anna Sebernegg [15], there are many approaches for this problem, such as time alignment, trajectory-based and rule-based. However, time alignment is the most popular due to its feasibility. Several representative algorithms with this approach are Dynamic Time Warping (DTW), hidden Markov model (HMM), etc. . .

c) Feature extraction

Feature extraction of the 3D motion data affects an algorithm's performance significantly. In 2017, Yusuke Goutsu, Wataru Takano and Yoshihiko Nakamura found two

matters with 3D motion features [16]. Firstly, local features, which got from some parts of the body, would give better results than global features, which collected the whole body. Secondly, the local feature often attaches a certain aim of an action. In 2019, Sheng Li and colleagues realized that only some joints of a human skeleton impacted a motion [17]. Then, this idea was applied in a 3D compression algorithm (MJS).

3. Related methods

a) Sliding window

Sliding window technique uses a window (or kernel) to slide through positions on an array. In this paper, the technique will be used for creating a list of proposals for each posture group.

b) Dynamic Time Warping algorithm (DTW)

DTW algorithm [18–21] are used to compare a similarity ability between a pattern and a proposal. In DTW, the series X and Y can be shrunk according to the following fixed rules:

- The first positions of 2 series are mapped
- The last positions of 2 series are mapped
- A position at X can be mapped into multiple-position at Y , and vice versa
- All positions must have their mapping
- Do not exist values such that $i < j$, $k < l$ and X_i is mapped to Y_l , X_j is mapped to Y_k . From the above rules, we have a distance between two series in the DTW algorithm:

$$DTW(X, Y) = \begin{cases} 0 & \text{if the length of } X \text{ and } Y \text{ are equal,} \\ \infty & \text{if the length of } X \text{ or } Y \text{ is } 0, \\ dist(X_1, Y_1) + \min(& \\ & DTW(Res(X), Y), DTW(X, Res(Y)), \\ & DTW(Res(X), Res(Y)) \\ \text{other cases} & \end{cases} \quad (1)$$

In equation (1), $dist(X_1, Y_1)$ is the distance between X_1 and Y_1 . It can be calculated by using the Euclidean distance, the Manhattan distance, etc. We choose Euclidean distance in this paper. We use a modification for the algorithm that is to use a fixed window length w such that X_i only maps to Y_j if $|i - j| < w$. In this way, it is likely to reduce significantly the running time of the DTW algorithm while the result is not affected much.

III. PROPOSED SYSTEM OF SEARCHING 3D MOTION PATTERNS

A dance is a consecutive sequence of several basic postures and their variants. In this paper, we consider each



Figure 5. Some pattern instances of ARM2. From left to right, pattern instances include posture “dâng hoa (cao)”, posture “dâng hoa (hạ)”, gesture “bay”, gesture “chống sườn”, gesture “đưa thoi”.

posture group consists of a basic posture and its whole variants (including named gestures of the posture). It means that each posture group at 3D space is called as a pattern. For instance, in Cheo dance, we have 5 patterns in the arm cluster: “múa chạy đàn” (ARM1), “dâng hoa” (ARM2), “dâng rượu” (ARM3), “múa cướp bông” (ARM4) and “tấu nhạc” (ARM5); 10 patterns in leg-sit cluster: “chữ V” (LEG1), “chữ chi” (LEG2), “quả trám” (LEG3), “chữ đỉnh” (LEG4), “đệm gót” (LEG5), “chân chống chân quỳ” (SIT1), “hai đầu gối cùng quỳ” (SIT2), “ngồi hai chân co về một bên” (SIT3), “đuỗi thẳng hai chân” (SIT4) and “hai chân bắt chéo” (SIT5). Now, as a case in point, let’s look at the dance “Múa phối kết hợp”. This dance uses 4 arm patterns, 5 leg patterns and 1 sit patterns with the different contribution (see Table I and II). In addition, each representative of a pattern is called as a pattern instance or instance. Thus, each pattern has one or more pattern instances (see Figure 5). A 3D movement is a dance, mixing between basic postures (or its variants) or a complex gesture at 3D space.

We can now define the problem of searching 3D motion patterns of Vietnamese traditional dances. Its input consists of a 3D movement and a set of pattern instances. Its output is a list of recognized patterns and their positions in the input. Our proposed system stores a list of pattern instances of basic postures and a list of angles for each pattern (called a list of pattern angles). Accordingly, the system will use those to recognize patterns through 2 main steps (Figure 6): pattern detection and pattern classification.

1. Choose a list of angles as features of a pattern (CheoAngle)

It is easy to see that angles calculation from joints will help us skip a data normalization step. The fact that Cheo data use many joints to display human motion in 3D space. As joints’ coordinates are local, we must convert these local coordinates into global ones through tool “bvh converter”¹. Then, we choose 19 joints for calculation (see Figure 7). In addition, angles of each joint at 2-axels motion data do not depend on these series’ coordinate system. Moreover, to decrease noises, with patterns of arm groups, we only consider the above half of the body (from hips to head);

¹<https://github.com/tekulvw/bvh-converter>

with patterns of leg and sit groups, we focus on the under half of the body (from hips to feet). Besides, based on the idea of a robot arm in kinematics [22], in the arm cluster, the line created from two shoulders is a base, each wrist is an end effector; in the leg-sit cluster, the line created from two hips are a base, each ankle is an end effector. Focusing on these joints, we will build a list of angles for each group basing on their impacts. Moreover, we also can get additional angles from other joints. These angles are calculated following the formula

$$\theta = \arccos\left(\frac{\vec{AB} \cdot \vec{A'B'}}{|\vec{AB}| \cdot |\vec{A'B'}|}\right) \quad (2)$$

, where \vec{AB} and $\vec{A'B'}$ are two vectors.

To facilitate for analysis, we use joint name in the vector notion. We assume that vector A_B connects joint A to B; vector C_D connects joint C to D. Accordingly, angle $\angle(A_B, C_D)$ is constructed by the two vectors A_B and C_D. For instance, the arm pattern ARM5 (group “tấu nhạc”) requires to raise up one hand and put down the other (see Figure 8). A list of angles for ARM5 needs satisfying criterions: features of the posture “tấu nhạc”, differentiation between the posture “tấu nhạc” and the other ones. The posture “tấu nhạc” is different from the other ones on rules: pattern ARM1’s attributes are put hands on the chest at the middle of body and move them the same direction; hands in pattern ARM2 must be put along the body and displaced in a similar manner (see Figure 4); in pattern ARM3, hands are in the same body side and move in the similar manner; pattern ARM 4’s attributes are that hands in the same extent and move in the opposite direction. Thus, we have the list of angles for ARM5 comprises $\angle(LSHOULDER_LWRIST, RSHOULDER_RWRIST)$, $\angle(LWRIST_RWRIST, LSHOULDER_RSHOULDER)$ and $\angle(LELBOW_LWRIST, RELBOW_RWRIST)$. In the first angle, LSHOULDER_LWRIST and RSHOULDER_RWRIST show one arm goes up and the other goes down. The other angles fulfill the second criterion.

2. Detect patterns

This section will give details about the generation of proposals for each instance. Here, this section consists of

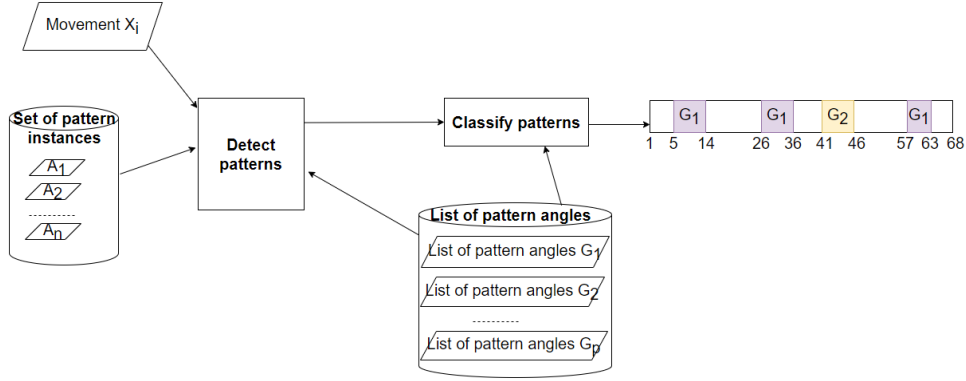


Figure 6. The system architecture. Accordingly, movement X_i is a whole dance or an excerpt of a dance at 3D space in time order (called time series); output is a list of pattern names (G_1, G_2, \dots) and their positions on a time series (in this figure, its length is 68); a set of pattern instances (A_1, A_2, \dots, A_n) is different samples of each posture group (called pattern); a list of pattern angles has angles which are calculated for a pattern.

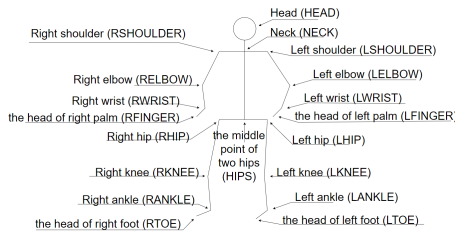


Figure 7. Joints of a human skeleton are used in the proposed system.

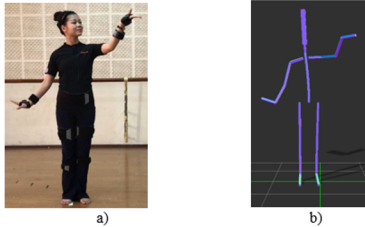


Figure 8. The posture “tấu nhạc”: a) 2D space b) 3D space. Source: disanso.vn

3 main steps: make a list of proposals, filter proposals, normalize proposals.

a) Make a list of proposals

The system makes a list of proposals for each instance A_j , which appears in the movement X_i , through 3 main steps (see Figure 9): extract feature, general proposal $proposal_k$, measure similarity. Steps are described more details as below:

- Extract feature: for both movement X_i and instance A_j , we extract features by calculating angles (using equation (2)) in the same list of pattern angles. Take the pattern instance “gánh” (has 76 frames) of pattern ARM5 and dance “múa phối kết hợp” (has 6906

frames) as an example. From the list of angles for ARM5 (see III.1), we calculate value of 3 angles and push into an array at each frame of this instance. As an effect, we obtain the instance feature as a 76×3 matrix. In like manner, we also possess the movement feature as a 6906×3 matrix.

- General proposal $proposal_k$: proposals $proposal_k$ will be created by using the sliding window technique with a proper window size following the length of the instance and a sliding step a . In this case, the sliding step’s default is 5 (see Figure 10). If the instance length is smaller than the default value, the sliding step will be the instance length.
- Measure similarity: the system sequentially compares features at each frame in A_j to features at each frame in a proposal $proposal_k$ (use equation (1)). The system returns a cost $cost_{A_j,k}$ of the optimal path between A_j and $proposal_k$ (called the similarity cost). If the similarity cost is 0, A_j is like $proposal_k$. Otherwise, the more the cost is, the more they are different.

b) Filter proposals

Filter proposals in the same instance A_j which has the length n_{A_j} , coefficient $\beta_{A_j} > 0$ and the number of angles n_θ as steps below:

- If the instance is a posture which has mostly no change of joints’ position, $thresh\theta := n_{A_j} * \alpha$ where α is constant. Otherwise, $thresh\theta := n_{A_j} * \beta_{A_j} * n_\theta$.
- Assume that A_j has a set of proposals $proposal_k$ and $cost_{A_j,k}$ is the similarity cost between A_j and $proposal_k$. We choose $proposal_k$ if $cost_{A_j,k} < thresh\theta$.

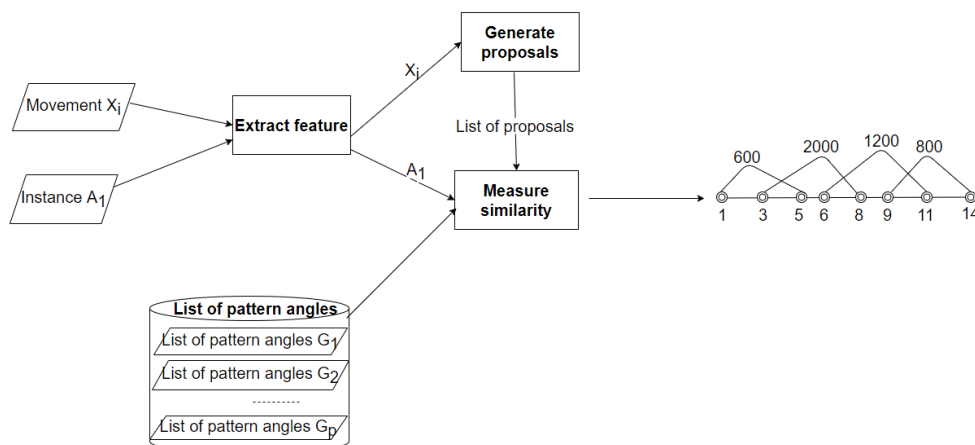


Figure 9. The workflow of the step Make a list of proposals for a pattern instance A_1 .

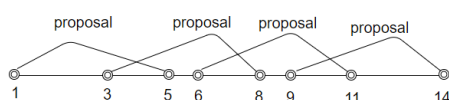


Figure 10. Several proposals are generated with the length 5 and sliding step 3

c) Normalize proposals

The system is only interested in proposals of instances in the same pattern. Two tasks need to do sequentially as:

- Merge proposals, which overlap each other in the same instance, into the unique one. As an illustration, in the Figure 10, we assume that the first and second and forth proposal are chosen. Since the first and second proposal is overlapped each other, we have to unify them into a unique one, which is a segment from frame 1 to 8. This proposal is written as $\langle 1, 8 \rangle$.
- Merge proposals, which duplicate and overlap each other of instances in the same pattern, into discrete ones. For example, pattern ARM1 has instances A1, A2 and A3. Accordingly, A1 consists of proposals: $\langle 1, 4 \rangle, \langle 6, 8 \rangle, \langle 12, 18 \rangle$; A2 comprises proposals: $\langle 4, 8 \rangle, \langle 12, 15 \rangle$; A3 encompasses proposals: $\langle 1, 8 \rangle$. After process, ARM1 will have discrete proposals: $\langle 1, 8 \rangle, \langle 12, 18 \rangle$.

3. Classify patterns

The system will do 2 steps: reorder proposals and attach a label to each pattern.

a) Reorder proposals

The system performs to reorder proposals in according to the following steps:

- The system lists the head and tail positions of the whole proposals of groups (called original proposals) in order to put on a list L.
- Duplicated elements in L are removed and the remain of L is ordered increasingly before putting on a list L2.
- Based on L2, create all possible proposals. From that, choose proposals such that they are related to at least an original proposal. Finally, we have a list of new proposals.

b) Attach each label to a pattern

Assume that there is a set of n patterns $\{group_k\}_n$ which has the same proposal SG . In pattern $group_k$, compare instances of A_j the pattern to SG , then calculate the similarity: $sim_{group_k} = \max\{sim_{group_k, A_j}\}$ such that $sim_{group_k, A_j} = \frac{n_{optimalSG}}{l_{SG}}$, where $n_{optimalSG}$ is the number of different positions of SG on the optimal path, l_{SG} is the length of SG . The label of SG is $\max\{sim_{group_k}\}$. In a particular case of choosing the label between the first arm group (posture “múa chạy đàn”) and the forth arm group (posture “cướp bông”), the system calculates angles in the same list of angles (called ARM1_4) which is different with ones of arm group 1 and 4. This procedure is also applied for arm group 3 and 4. For leg-sit cluster, at one time, only one group can appear. To enforce this rule, the system must classify labels which appear leg and sit groups in the same proposal.

IV. EXPERIMENTS AND RESULTS

1. Dataset

We use 3D movement data of Cheo dance [5] for experiments. This 3D dataset is a part of ANIAGE (High Dimensional Heterogeneous Data based Animation

Techniques for Southeast Asian Intangible Cultural Heritage Digital Content)², conducted by HMI Laboratory³. The dataset is in BVH format that stores human skeleton motion at 3D space. We use this dataset to construct a pattern database of 70 patterns and a test dataset of 7 files. This test dataset is called HMI-3DCheo.

For patterns, basic postures are represented by 5 frames while their variants have variable lengths. The two reasons for this rule. First, a posture is body features not vary over time. It means that a posture only needs 1 frame. However, the DTW algorithm calculates total cost of the optimal path for two time series. In equation (1), this algorithm must get values of 3 neighbors for the considered element. It means that each series needs at least 3 frames. To avoid noise, we choose the length as 5 frames. Second, a variant is a meaning motion in dance. In fact, gestures have 3 stages: start, main, end. The stage “main” is very important because it is likely to decide the gesture detection. Hence, we tried to get this stage for each instance as possible as we could. As an evidence, gesture “bay” mimics a flying action of a bird (see Figure 5). It takes 130 frames to describe the flying action.

HMI-3DCheo consists of “múa phối kết hợp” (6906 frames), compound of gestures “quay lưu không nam và nữ” (3149 frames), compound of gestures “đi nữ lệch và nam ngang” (2744 frames), gesture “đi quả trám” (1037 frames), gesture “lão say” (1604 frames), excerpt “Sứ Vân giả dại” (“xe tơ – dệt vải”, 6218 frames), “Thị Kính hát sù râu ba than” (6804 frames). Distribution of patterns in the test data is shown in the Table I and II.

2. Metrics

We use precision, recall and F1-score to evaluate experimental outcomes. They are calculated for an instance A_j [?] as follows:

$$precision_{A_j} = \frac{TruePred_{A_j}}{TotalPred_{A_j}} \quad (3)$$

$$recall_{A_j} = \frac{TruePred_{A_j}}{Real_{A_j}} \quad (4)$$

$$F1 - score_{A_j} = \frac{2 * precision_{A_j} * recall_{A_j}}{precision_{A_j} + recall_{A_j}} \quad (5)$$

where $TruePred_{A_j}$ is the length of true A_j predicted segments on the whole test data; $TotalPred_{A_j}$ is the total of A_j predicted segments on the whole test data; $Real_{A_j}$ is the total of real A_j segments on the whole test data.

²<https://www.euh2020aniage.org/>

³<https://hmiuet.wordpress.com/projects/>

3. Comparison methods

Since each 3D movement of the dance is represented by a time series, we need to compare with other methods for time series such as edit distance on real sequence (EDR) [21, 23–25] and Longest common subsequence (LCSS) [20, 21, 26]. Some system constraints must be modified for these methods. In the step Make a list of proposals (see III.2.a), cost matrices of EDR and LCSS are edited. Assume that we have a proposal $X = \{x_i\}_{i=1}^m$ and an instance $A = \{y_j\}_{j=1}^n$. Accordingly, $x_i = \{\theta_k\}_{k=1}^p$ and $y_j = \{\omega_k\}_{k=1}^p$ are features (each feature consists of p angles). The condition for x_i and y_j being alike in LCSS is $\sqrt{\sum_{k=1}^p (\theta_k - \omega_k)^2} \leq \frac{\phi * \pi}{180} * \sqrt{n}$ and not being alike in EDR is $\sqrt{\sum_{k=1}^p (\theta_k - \omega_k)^2} > \frac{\phi * \pi}{180} * \sqrt{n}$, where ϕ is the acceptable angle deviation. At the method LCSS, the longest common substring in matrix T is named ω . At the method EDR, the least edition operation on X and A in matrix T is named n_{EDR} . In step Filter proposals (see III.2.b), LCSS only selects proposals which satisfy $n_\omega > 0.4 * n$ while EDR filter following the condition: $n_{EDR} < 0.4 * \max(n, m)$. Consequently, the step Attach each label to a pattern (see III.3.b) is modified as $sim_A^X = \frac{n_\omega}{n}$ at LCSS and $sim_A^X = \max(n, m)$ at EDR. In addition, in the EDR method, label of SG is $\min\{sim_A^X\}$.

On the other hand, we cannot apply a deep learning model for this problem. The reason is too little training data. In fact, we have only 70 instances of 15 patterns for training process, which means an average of 4.6 instances per pattern. As a result, a training model would likely overfit the small training dataset.

4. Results and discussion

We performed experiments on the dataset HMI-3DCheo. This paper pays attention on the efficiency of DTW algorithm and the feature selection technique CheoAngle.

The Table V and IV show experimental results in order to compare values among methods. Generally, the precision and recall averages of DTW better than ones of LCSS and EDR. In the arm cluster, recall averages of DTW, LCSS and EDR are 49.78%, 46.74%, 35.82%. In leg-sit cluster, precision averages of DTW, LCSS and EDR are 66.36%, 61.47%, 60.82%; their recall averages are 62.55%, 54.99%, 52.72%.

To prove the efficiency of CheoAngle, this paper bases on two comparison methods: ShengLi [17] and JointRelativeDist [27]. The method ShengLi considers joints which cause a movement. Accordingly, the paper only applies this way to build a list of angles for each cluster (arm, leg, sit) while CheoAngle makes a list of angles for each posture

Table I
DISTRIBUTION OF PATTERNS OF ARM GROUP FOR TEST DATA

Pattern Test data	ARM1	ARM2	ARM3	ARM4	ARM5
múa phối kết hợp (6906 frames)	35.9%	23.8%	10.0%	0.0%	21.4%
quay lưu không nam và nữ (3149 frames)	0.0%	100.0%	0.0%	0.0%	0.0%
đi nữ lệch và nam ngang (2744 frames)	36.0%	64.0%	0.0%	0.0%	0.0%
đi quả trám (1037 frames)	0.0%	100.0%	0.0%	0.0%	0.0%
lão say (1604 frames)	0.0%	12.0%	32.4%	0.0%	29.3%
xe tơ – dệt vải (6218 frames)	16.3%	22.0%	23.5%	9.2%	26.2%
Thị Kính hát sử râu ba than (6804 frames)	23.2%	32.1%	38.9%	2.2%	1.5%
HMI-3DCheo	21.3%	39.8%	18.7%	2.5%	12.9%

Table II
DISTRIBUTION OF PATTERNS OF LEG AND SIT GROUP FOR TEST DATA

Pattern Test data	LEG1	LEG2	LEG3	LEG4	LEG5	SIT1	SIT2	SIT3	SIT4	SIT5
múa phối kết hợp (6906 frames)	37.0%	37.2%	2.8%	7.1%	6.0%	9.7%	0.0%	0.0%	0.0%	0.0%
quay lưu không nam và nữ (3149 frames)	20.9%	9.1%	14.1%	27.9%	23.2%	0.0%	0.0%	0.0%	0.0%	0.0%
đi nữ lệch và nam ngang (2744 frames)	27.2%	71.8%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
đi quả trám (1037 frames)	7.1%	24.4%	34.8%	0.0%	11.2%	0.0%	0.0%	0.0%	0.0%	0.0%
lão say (1604 frames)	14.7%	20.2%	30.2%	31.2%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
xe tơ – dệt vải (6218 frames)	11.8%	6.8%	0.06%	0.1%	5.5%	0.8%	5.9%	51.5%	16.1%	0.2%
Thị Kính hát sử râu ba than (6804 frames)	22.6%	56.3%	1.5%	0.0%	0.9%	0.7%	16.4%	0.0%	0.0%	0.0%
HMI-3DCheo	23.0%	33.9%	5.6%	6.6%	5.9%	2.7%	5.2%	11.2%	3.5%	0.05%

Table III
TRIAL OUTCOMES ON THE ARM CLUSTER

Methods	Metrics	ARM1	ARM2	ARM3	ARM4	ARM5
DTW	Precision (%)	65.7	85.6	51.2	1.6	66.7
	Recall (%)	75.6	72.3	29.4	4.1	67.5
	F1-score (%)	70.3	78.4	37.3	2.3	67.1
LCSS	Precision (%)	75.4	71.1	63.1	0.0	58.3
	Recall (%)	74.3	28.0	49.9	0.0	81.5
	F1-score (%)	74.8	40.2	55.7	0.0	68.0
EDR	Precision (%)	91.6	60.2	74.8	0.0	65.3
	Recall (%)	70.2	22.8	35.6	0.0	50.5
	F1-score (%)	79.5	33.1	48.3	0.0	57.0

group. The method JointRelativeDist relatively calculates a distance between a joint pair of 2 skeletons. From this idea, we construct a way to choose features for a certain skeleton. Assume that $P_1 = \{u_k\}_{k=1}^P$ is a considered skeleton. The Euclidean distance between two points $u_k = (u_k^x, u_k^y, u_k^z)$ and $u_o = (u_o^x, u_o^y, u_o^z)$ on P_1 is:

$$D_{P_1}(u_k, u_o) = \sqrt{(u_k^x - u_o^x)^2 + (u_k^y - u_o^y)^2 + (u_k^z - u_o^z)^2} \quad (6)$$

From the equation (6), we describe a feature of the skeleton P_1 whose is obtained from the method JointRelativeDist is:

$$F_{P_1} = \{D_{P_1}(u_k, u_o)\}_{i=1}^q \quad (7)$$

where q is the number of point-need-calculated pairs of the skeleton. Here, the paper defines joint pairs for arm,

Table IV
TRIAL OUTCOMES ON THE LEG-SIT CLUSTER

Methods	Metrics	LEG1	LEG2	LEG3	LEG4	LEG5	SIT1	SIT2	SIT3	SIT4	SIT5
DTW	Precision (%)	83.8	72.5	34.1	53.2	35.4	66.4	74.6	99.0	96.5	48.1
	Recall (%)	44.7	41.0	30.0	69.1	34.8	94.4	23.4	93.5	94.6	100.0
	F1-score (%)	58.3	52.4	31.9	60.1	35.1	78.0	35.7	96.2	95.5	65.0
LCSS	Precision (%)	81.0	56.4	25.0	56.8	34.9	82.0	74.5	95.1	86.8	22.2
	Recall (%)	65.0	87.0	1.8	41.0	33.5	88.6	7.7	97.8	96.7	30.8
	F1-score (%)	72.1	68.4	3.4	47.6	34.1	85.2	14.0	96.4	91.5	25.8
EDR	Precision (%)	82.4	63.2	62.5	36.5	32.3	90.2	29.3	97.7	91.9	22.2
	Recall (%)	54.5	83.6	11.2	39.8	33.9	80.9	1.6	91.1	99.8	30.8
	F1-score (%)	65.6	72.0	19.0	38.1	33.1	85.3	3.1	94.3	95.7	25.8

leg and sit cluster instead of posture groups.

To compare the proposed method CheoAngle with ShengLi and JointRelativeDist following F1-score, we can see that this way is efficient on arm and leg-sit cluster. In the arm cluster (Table V), values F1-score of patterns ARM1, ARM2, ARM3 and ARM5 are higher than ones of ShengLi and JointRelativeDist. In particular, the pattern ARM4 has the highest result at JointRelativeDist. In the leg-sit cluster (Table VI), 8/10 patterns at CheoAngle have F1-score values higher than ones of the method remains. In particular, F1-score values of the pattern ARM1 and ARM2 at ShengLi is higher than others.

On the contrary, the result for pattern ARM4 is low for all methods. There might be several reasons for this result. First, the number of instances for pattern ARM4 are very small compared to others (see Table I and II). Meanwhile, pattern ARM4 has many variants in which hands are extended and moved in opposite direction. These invariants are also similar to variant of pattern ARM1 and ARM3. As a result, it is very easy to mistake an instance of ARM4 for ARM1 and ARM3. Second, the contribution of this pattern in the test dataset is very little (see Table I and II). Moreover, the appearance of pattern ARM4 in the test dataset often complicated cases while the ARM4 instances are basic samples of the basic posture “múa cướp bông”. As an example, in the excerpt “Súy Vân giả dại”, action “xâu kim” (thread needle), at the middle of the body, uses a ARM5 variant (prepare for needle thread) and a ARM4 variant (threading needle). And so, the system can consider this ARM4 variant as a ARM1 case. This means that if ARM4 predictions are incorrect, ARM1 and ARM3 outcomes also are affected seriously. Similarly, we can explain the unlogic results of leg-sit patterns. Moreover, we have difficulty distinguishing leg-sit patterns. It is because their attributes are not much

dissimilar. To illustrate, we analyze LEG2 and LEG5. In gesture “chân chữ chi” (LEG2), a performer must lift the heels (LEG5).

V. CONCLUSIONS

Results at Section IV proves the efficiency of applying several techniques. The technique sliding window supports to search patterns on the global scope. The algorithm DTW makes patterns classification to become simple. In addition, the way to select need-to-calculated angles of motions also contributes an important role to predictions as well as the data scale.

Besides advantages, the proposed algorithm has a small group of limitations. The first limitation is the dependency on pattern instances’ length. Instances have short or long length depends on dance rhythm and skills of the dancer. Thus, the pattern’s length affects the proposals filter. The second limitation is the window size and the stride. If the stride is short, it takes a long time to run the algorithm. Otherwise, the long stride will skip patterns that we need to search. Meanwhile, the shorter window size is, the more difficult the system classify patterns due to getting only a part of information in the pattern. In contradiction, the longer window size is, the more easily the system predicts pattern incorrectly.

In the future, we will optimize lists of pattern angles and complete instance and test dataset. For pattern angles optimization, we improve techniques. In classification process, a group having few highlight features is easy to predict incorrect patterns. Moreover, several patterns barely have no movement, but some joints change positions very slightly. This will cause the difficulty for comparison between groups which are in the same posture. Thus, we will add the information of rotation degree of joints in order to

Table V
FEATURES COMPARISON ON THE ARM CLUSTER

Methods	Metrics	ARM1	ARM2	ARM3	ARM4	ARM5
CheoAngle	Precision (%)	65.7	85.6	51.2	1.6	66.7
	Recall (%)	75.6	72.3	29.4	4.1	67.5
	F1-score (%)	70.3	78.4	37.3	2.3	67.1
ShengLi	Precision (%)	61.0	93.8	27.1	0.0	44.2
	Recall (%)	76.5	46.4	30.4	0.0	78.0
	F1-score (%)	67.9	62.1	28.7	0.0	56.4
JointRelativeDist	Precision (%)	84.7	96.1	8.9	11.8	46.4
	Recall (%)	48.6	38.8	4.9	33.3	42.8
	F1-score (%)	61.8	55.3	6.3	17.4	44.5

Table VI
FEATURES COMPARISON ON THE LEG-SIT CLUSTER

Methods	Metrics	LEG1	LEG2	LEG3	LEG4	LEG5	SIT1	SIT2	SIT3	SIT4	SIT5
CheoAngle	Precision (%)	83.8	72.5	34.1	53.2	35.4	66.4	74.6	99.0	96.5	48.1
	Recall (%)	44.7	41.0	30.0	69.1	34.8	94.4	23.4	93.5	94.6	100.0
	F1-score (%)	58.3	52.4	31.9	60.1	35.1	78.0	35.7	96.2	95.5	65.0
ShengLi	Precision (%)	87.9	69.1	59.8	36.2	61.8	25.1	5.8	95.6	87.5	0.0
	Recall (%)	58.1	70.7	3.5	32.4	21.4	94.4	11.2	16.2	73.0	0.0
	F1-score (%)	69.9	69.9	6.6	34.2	31.8	39.6	7.6	27.7	79.6	0.0
JointRelativeDist	Precision (%)	77.0	56.9	8.2	24.3	59.2	7.5	0.0	0.0	0.0	0.0
	Recall (%)	62.7	11.1	0.8	2.7	4.6	87.7	0.0	0.0	0.0	0.0
	F1-score (%)	69.1	18.6	1.5	4.9	8.6	13.8	0.0	0.0	0.0	0.0

complement patterns comparison. For the instance dataset, we must add almost representatives for patterns: ARM2, ARM4, LEG1, SIT2, SIT4, SIT5. For the test dataset, we must give dances or compounds of basic postures for patterns: ARM2, ARM3, ARM4, LEG3, LEG4, LEG5, SIT1, SIT2, SIT3, SIT4, SIT5.

REFERENCES

[1] Lê Ngọc Canh, *Nghệ thuật múa Chèo*. Nhà xuất bản Sân khấu, 2003.

[2] Nhà hát tuồng Việt Nam, “Múa tuồng (vũ đạo),” 2020. [Online]. Available: <http://vietnamtuongtheatre.com/mua-tuong-vu-dao/>

[3] Nguyễn Bích Thủy, “Nghệ thuật trình diễn hát xoan phú thọ,” 2018. [Online]. Available: http://svhttdl.phutho.gov.vn/tin/nghe-thuat-trinh-dien-hat-xoan-phu-tho_1010.html

[4] Trần Thị Ngọc, *Giáo trình múa Chèo*. Trường đại học Sân khấu và Điện ảnh Hà Nội, 1998.

[5] HMI Laboratory, VNU University of Engineering and Technology, “3d dataset for the traditional cheo dance,” 2021, accessed: 2021-11-16. [Online]. Available: <http://disanso.vn/vidman>

[6] B. Rennhak, T. Shiratori, S. Kudoh, P. Vinayavekhin, and K. Ikeuchi, “Detecting dance motion structure using body components and turning motions,” *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2264–2269, 2010.

[7] E. Protopapadakis, A. Grammatikopoulou, A. Doulamis, and N. Grammalidis, “Folk dance pattern recognition over depth images acquired via kinect sensor,” *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-2/W3, pp. 587–593, 2017. [Online]. Available: <https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W3/587/2017/>

[8] Z. Li, “Three-dimensional diffusion model in sports dance video human skeleton detection and extraction,” *Advances in Mathematical Physics*, vol. 2021, pp. 1–11, sep 2021. [Online]. Available: <https://doi.org/10.1155/2021/3772358>

[9] X. Hu and N. Ahuja, “Unsupervised 3d pose estimation for hierarchical dance video recognition,” *CoRR*, vol. abs/2109.09166, 2021. [Online]. Available: <https://arxiv.org/abs/2109.09166>

[10] F. A. Damastuti, A. K. Nurindiyani, and D. Pramadihanto, “Gesture 3d modeling for traditional javanese dance,” in *2018 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*, 2018, pp. 69–73.

[11] J. Sutopo, M. K. A. Ghani, B. M. Aboobaidar, and Zuhawati, “Dance gesture recognition using space component and effort component of laban movement analysis,” *International Journal of Scientific & Technology Research*, vol. 9, no. 2, 2020.

[12] N. Jain, V. Bansal, D. Virmani, V. Gupta, L. Salas-Morera, and L. Garcia-Hernandez, “An enhanced deep convolutional neural network for classifying indian classical dance forms,” *Applied Sciences*, vol. 11, no. 14, 2021. [Online]. Available: <https://www.mdpi.com/2076-3417/11/14/6253>

- [13] E. Protopapadakis, A. Voulodimos, A. Doulamis, S. Camarinopoulos, N. Doulamis, and G. Miaoulis, "Dance pose identification from motion capture data: A comparison of classifiers," *Technologies*, vol. 6, no. 1, 2018. [Online]. Available: <https://www.mdpi.com/2227-7080/6/1/31>
- [14] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "Segmenting time series: A survey and novel approach," in *Data mining in time series databases*. World Scientific, 2004, pp. 1–21.
- [15] A. S. . P. K. . H. Kaufmann, *Motion Similarity Modeling A State of the Art Report*, 8 2020.
- [16] G. Yusuke, T. Wataru, and N. Yoshihiko, "Classification of multi-class daily human motion using discriminative body parts and sentence descriptions," *Int. J. Comput. Vision*, vol. 126, no. 5, p. 495–514, May 2018. [Online]. Available: <https://doi.org/10.1007/s11263-017-1053-3>
- [17] S. Li, J. Tingting, T. Yonghong, and H. Tiejun, "3d human skeleton data compression for action recognition," in *2019 IEEE Visual Communications and Image Processing (VCIP)*, Dec 2019, pp. 1–4.
- [18] D. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *KDD Workshop*, 1994, pp. 359–370.
- [19] E. Keogh and C. A. Ratanamahatana, "Exact indexing of dynamic time warping," *Knowledge and Information Systems*, vol. 7, no. 3, pp. 358–386, mar 2005. [Online]. Available: <https://link.springer.com/article/10.1007/s10115-004-0154-9>
- [20] M. Vlachos, D. Gunopulos, and G. Kollios, "Discovering similar multidimensional trajectories," *Proceedings 18th International Conference on Data Engineering*, pp. 673–684, 2002.
- [21] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, and E. J. Keogh, "Indexing multi-dimensional time-series with support for multiple distance measures," in *KDD '03*, 2003.
- [22] H. Václav, "Robot kinematics," 2005. [Online]. Available: <http://people.ciirc.cvut.cz/~hlavac/TeachPresEn/51Robotics/11KinematicsRobot.pdf>
- [23] L. C. . M. T. Özsü , Vincent Oria, "Robust and fast similarity search for moving object trajectories," *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 491–502, 2005.
- [24] L. Allison, "Dynamic programming algorithm (dpa) for edit-distance," Faculty of Information Technology (Clayton), Monash University, Australia, 1999. [Online]. Available: <http://users.monash.edu/~lloyd/tildeAlgDS/Dynamic/Edit/>
- [25] L. V. I. and I. V., "Binary codes capable of correcting deletions, insertions and reversals," *SPhD*, vol. 10, p. 707, 1966. [Online]. Available: <https://ui.adsabs.harvard.edu/abs/1966SPhD...10..707L/abstract>
- [26] G.-M. I. of Electronics and C. Science, "Longest common subsequences," France, 1998. [Online]. Available: <https://www-igm.univ-mlv.fr/~lecroq/seqcomp/node4.html>
- [27] J. K. T. Tang, H. Leung, T. Komura, and H. P. H. Shum, "Emulating human perception of motion similarity," *Comput. Animat. Virtual Worlds*, vol. 19, no. 3–4, p. 211–221, sep 2008.



Le Thi Thanh Van received Master degree of Computer Science from University of Engineering and Technology, Vietnam National University, Hanoi in 2021. Her research includes computer vision, machine learning.
Email: lvank52thb@live.com



Vu Ngoc Quang received Master degree of Computer Science from Japan Advanced Institute of Science and Technology, Japan in 2016. He is working as Research Assistant in HMI Laboratory, University of Engineering and Technology, Vietnam National University, Hanoi.

His research includes machine learning and brain-computer interface.

Email: quang.vn@outlook.com



Pham Thanh Huyen received Master's degree in Computer Science at Hanoi National University of Education in 2010. She is a Ph.D student in the University of Engineering and Technology, Vietnam National University, Hanoi. She is a lecturer of Ha Long University in Quang Ninh province.

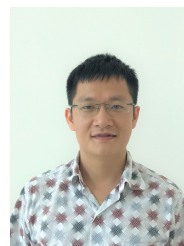
Her research includes database, data mining, machine learning.
Email: phamthanhuyen@daihochalong.edu.vn



Ma Thi Chau received Ph.D degree from University of Engineering and Technology, Vietnam National University, Hanoi in 2014.

Her research includes Computer Graphics and 3D Reconstruction.

Email: chaumt@vnu.edu.vn



Le Thanh Ha received Ph.D degree from Korea University, Korea in 2010.

He is the head of HMI Laboratory, University of Engineering and Technology, Vietnam National University, Hanoi.

His research includes Video Coding and Communication, Image Processing, Computer Vision and Satellite Image Processing.

ing.

Email: ltha@vnu.edu.vn