Blind Robust Watermarking Mechanism Based on Maxima Curvature of 3D Motion Data

Ling Du[1,2], Xiaochun Cao[1,⋆], Muhua Zhang[1], and Huazhu Fu[1]

[1] School of Computer Science and Technology,
Tianjin University, Tianjin, 300072, China
[2] School of Computer, Shenyang Aerospace University, Shenyang, 110136, China
{duling,xcao}@tju.edu.cn

Abstract. This paper presents a blind robust watermarking mechanism for copyright protection of 3D motion data. The mechanism segments motion data based on stable anchor-points captured by the maxima in spatio-temporal curvature and filtered by posterior attack model. For each segment, we make a randomized cluster division of 3D points based on a secret key. A watermark is then embedded within these clusters by Triangle Orthocenter based encoding approach. Experimental results show that the proposed watermarking scheme is robust against many possible attacks such as uniform affine transforms (scaling, rotation and translation), noise addition, reordering and cropping.

1 Introduction

With the rapid progress in motion capture (mocap) technology, 3D motion data are being widely used in animations, video games, movies, human motion analysis and other fields. 3D motion data has high scientific and commercial value, which makes its copyright protection becoming an important issue. Digital watermark technology [6] provides an effective method for digital copyright protection. So far, digital watermark techniques mainly consist of image watermarking [7][9][11][23], video watermarking [17][20][24], audio watermarking [3][13][21], mesh watermarking [4][5][14][15][19] etc. Although different watermarking methods have been developed for other kinds of media, they cannot be directly applied to 3D data. The most important reasons are dimensionality. Since other kinds of media are not generalized to handle problems related to higher dimensional data, developing watermarking methods for 3D motion data is more challenging.

3D human motion data consist of motion information related to human joints, and can be represented by a set of trajectories. Recently, for 3D motion trajectory data watermarking, Kim et al [10] present a algorithm based on multiresolution representation and spread spectrum. The algorithm not only can resist against random noises, but also has merits of spread spectrum such as the resilience to common signal processing as well as the robustness to time warping. In [22], original data is firstly transformed into the frequency domain by discrete cosine transformation, and then chose the most significant components to insert the

[⋆] Corresponding author.

M. Kirchner and D. Ghosal (Eds.): IH 2012, LNCS 7692, pp. 110–124, 2013.

watermark according to the amplitudes of the signal. Both methods in [10] and [22] implement the watermarking of motion data based on the spread spectrum method. However, they use principal component analysis (PCA) approach for segmentation, with only a rough semantic segmentation capability. Moreover, the time duration of motion is limited in their methods, and it only robust against similarity transformation attack, not robust against affine transformation.

Considering the download and transmission of motion data, Li and Okuda [12] propose a method based the progressive representation including the base frames and enhancement ones. The motion is sent frame by frame from the base frames to the enhancements until all the frames are restored. The progressive encoding method gives the frames of the motion an order which ranks frames by their importance. With such an order, watermark can be embedded into a characteristic of each frame. However, in terms of robustness, this method only tolerates random noises to some extent. In addition, the extraction algorithm needs information about the original frame, so it belongs to the non-blind watermarking algorithm as [10] and [22]. Motwani et al [16] propose a fragile watermarking algorithm for 3D motion curves. Their approach implements a prototype in spread spectrum domain by using a Haar wavelet transform on the 3D data and alters the wavelet coefficients. However, the algorithm also belong to non-blind watermarking algorithm.

In order to enhance the capacity of watermark robustness to affine transformation attacks, Agarwal and Prabhakaran [1][2] provide blind robust watermark algorithms of 3D motion data. They segment motion trajectory and identify clusters of 3D points per segment. The watermark can be embedded and extracted within these clusters by the proposed extension of 3D quantization index modulation. The watermarking schemes are robust against many types of attacks and works well. However, the motion segmentation method and ordering criteria of encoding points need more investigation to identify robustness against noise addition attacks. And for its Euclidian Distance based encoding method, the shifting direction of encoding points do not take the movement direction of joint into account. It may decrease the imperceptibility of the watermark scheme.

This paper presents a blind robust watermarking mechanism for the trajectory of human joints based on maxima curvature of 3D motion data. The technical contributions are identified as follows.

-Robust segmentation method based on maxima curvature and posterior attack model. Compared to current segmentation method, it is more robust against possible attacks due to the stability of anchor-points used for segmentation.

-Blind robust watermark encoding method. For bit encoding process inside the cluster, we propose a novel Triangle Orthocenter based encoding method. Compared to current encoding approach, it has better imperceptibility and robustness to noise addition attack.

The remainder of the paper is organized as follows. In section 2, we describe the scheme design for our watermarking method. In section 3, some typical motions with different parameter settings are employed to demonstrate the advantage of our approach with respect to other methods. Finally, we draw a conclusion for our work in section 4.

2 Scheme Design

In our work, the motion capture data is represented by a set of trajectories consisted of $M(1 \leq M \leq 19)$ joint trajectories of human body. The watermark is represented as multiple bits (series 0s and 1s). Fig.1 shows the watermark scheme proposed. Firstly, we extract the spatio-temporal curvature extremes for each trajectory as candidate anchor points, and then filter the stable anchor-points by posterior attack model. Secondly, motion trajectories are segmented based on the remaining stable anchor points. Randomized cluster division is done based on a secret key for each segment. Finally, the watermark is embedded within these clusters by Triangle Orthocenter based encoding approach. The watermark is extracted accordingly as shown in the right side of Fig.1.

2.1 Candidate Anchor Points

In order to ensure the embedded watermark can be extracted when a part of the marked motion trajectory encountering attacks, the original 3D motion trajectory is temporally divided into several parts before watermark embedding. Since segmentation can help pinpoint the presence of the watermark during the extraction, the point used for segmentation must be robust against motion editing operations such as noise additions and 3D transforms.

Motion trajectory of 3D motion captured data can be expressed by the position vector composed of positions for each time instants, as

$$\mathbf{r}(t) = [x(t), y(t), z(t)]^T, \tag{1}$$

where $x(t)$, $y(t)$, $z(t)$ represent the 3D coordinates of the joint at the time t. The quantitative measure of motion can be acquired by its velocity $\mathbf{v}(t)$, and acceleration $\mathbf{a}(t)$, which are given by the first and second derivatives of position.

$$\mathbf{v}(t) = \mathbf{r}^{'}(t) = [x^{'}(t), y^{'}(t), z^{'}(t)]^T, \tag{2}$$

$$\mathbf{a}(t) = \mathbf{r}^{''}(t) = [x^{''}(t), y^{''}(t), z^{''}(t)]^T. \tag{3}$$

Human observers are able to perceive dynamic instants that stem from discontinuities in velocity or acceleration and can be captured by the maxima of spatio-temporal curvature [18]. The curvature $k(t)$ at time t is given by

$$k(t) = \frac{\|\mathbf{v}(t) \times \mathbf{a}(t)\|}{\|\mathbf{v}(t)\|^3}, \tag{4}$$

where $'\times'$ represents the cross product and $\|\mathbf{v}(t)\|$ represents speed. Finally, the maxima in spatio-temporal curvature is captured as

$$\mathbf{P}(t) = \left\{ \mathbf{r}(t) \mid k^{'}(t) = 0 \right\}. \tag{5}$$

Because the maxima in spatio-temporal curvature of a trajectory is invariant to 3D affine transforms, we use these instants as candidate embedding anchor-points during the watermarking. Each dynamic instant represents an important

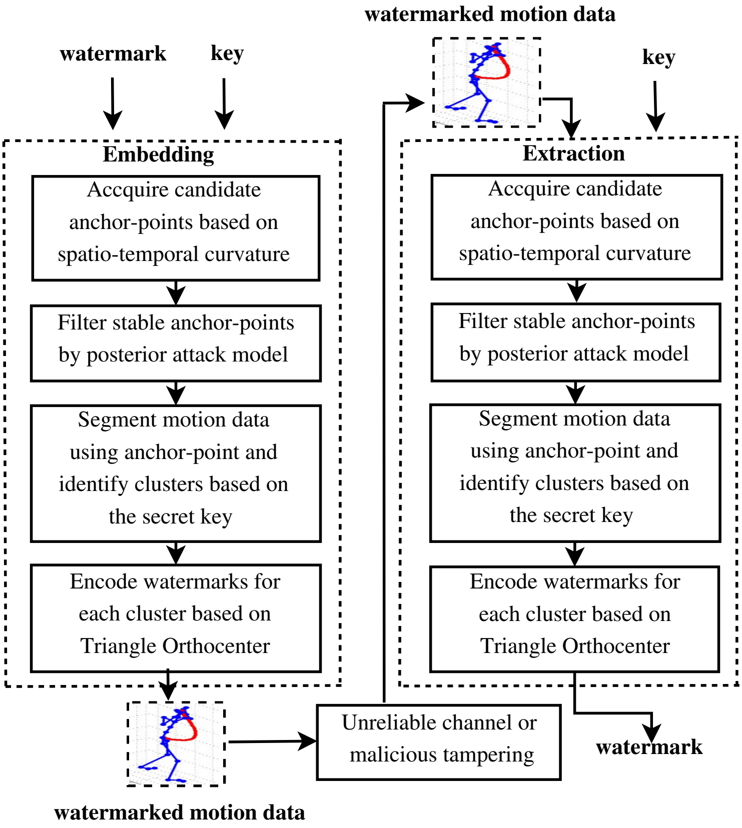

watermark
key
Embedding
Accquire candidate anchor-points based on spatio-temporal curvature
Filter stable anchor-points by posterior attack model
Segment motion data using anchor-point and identify clusters based on the secret key
Encode watermarks for each cluster based on Triangle Orthocenter
watermarked motion data
Unreliable channel or malicious tampering
watermarked motion data
key
Extraction
Accquire candidate anchor-points based on spatio-temporal curvature
Filter stable anchor-points by posterior attack model
Segment motion data using anchor-point and identify clusters based on the secret key
Encode watermarks for each cluster based on Triangle Orthocenter
watermark

Fig. 1. Watermark scheme design for 3D motion data

change in the motion characteristics. But for robust watermarking of 3D motion data, not all of these dynamic instants used for segmentation are robust against possible attacks such as noise addition.

2.2 Stable Anchor Points

In order to filter out those anchor points sensitive to noise, smoothing and other attacks, we introduce posterior attack model to obtain the stable anchor-points. Virtual attacks are done by adding Gaussian noise, smoothing and other attacks:

$$\widetilde{\mathbf{P}}_i(t) = f(\mathbf{P}(t), \mathbf{G}(\delta)), \tag{6}$$

where $\mathbf{P}(t)$ represents the candidate anchor-point captured by curvature maxima, $\mathbf{G}(\delta)$ represents combination of several attacks, $\widetilde{\mathbf{P}}_i(t)$ represents the candidate anchor-points after the i-th attack. We apply N times virtual attacks to candidate anchor-points with random attack parameters and combinations. The set of candidate anchor-points after virtual attacks is

$$\mathrm{P} = \left\{ \widetilde{\mathbf{P}}_i(t) \mid i = 1...N \right\}. \tag{7}$$

In our implementation, we conduct four groups of attacks with different intensities and ratio including Gaussian noise 10dB and 1dB, smoothing, on various ratios (1% to 50%) of attacked anchor points. By comparing the position of original candidate anchor-point $\mathbf{P}(t)$ to the candidate anchor-point $\mathbf{P}_i(t)$, we compute the posterior probability of attacks as

$$p(\mathbf{P}(t) \mid \mathrm{P}) = \frac{1}{N} \sum_{i=1}^{N} x_i(t), \tag{8}$$

where $x_i(t)$ is given by

$$x_i(t) = \begin{cases} 1, & \text{if } \mathbf{P}(t) = \widetilde{\mathbf{P}}_i(t) \\ 0, & \text{if } \mathbf{P}(t) \neq \widetilde{\mathbf{P}}_i(t) \end{cases}. \tag{9}$$

And then we select the points which have the larger posterior probability than the median of the posterior probability values, *Median*, for each candidate anchor-point as stable embedded anchor-points, which is given by

$$\mathbf{P}_{feature} = \{\mathbf{P}(t) \mid p(\mathbf{P}(t) \mid \mathrm{P}) > Median\}. \tag{10}$$

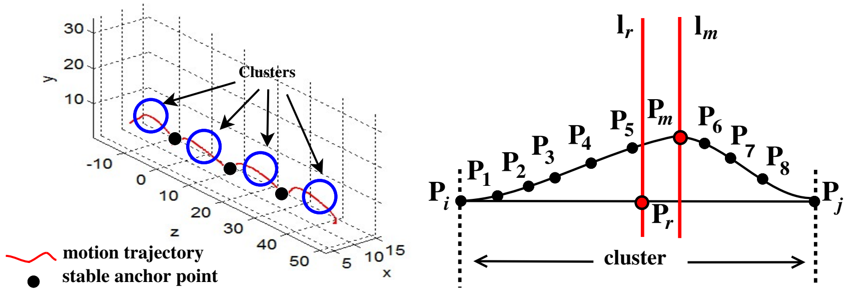
Finally, the motion trajectory data is segmented based on the remaining stable anchor points.

2.3 Watermark Embedding

In order to ensure the best robustness for our watermarking method, random choice for embedding position is an important and pivotal solution. Therefore, we divide each segment into clusters on the basis of a key *sk*. The clusters are chosen in a random fashion separated by a random number of points called embedding distance. Fig.2(a) shows the clusters based on segments of motion trajectory. Then for certain number of watermark copies, we randomly select the clusters to embed for each watermark copy.

For the watermark embedding inside each cluster, a new Triangle Orthocenter based approach is proposed to encode for every cluster in the trajectory. Firstly, we identify invariant point set and encoding point set which are used as reference and watermark bit hiding respectively. The choice of the invariant points is based on a scalar quantity which is invariant against uniform affine transformations, such as maximum Euclidian distance between the set of points. The encoding point set is the difference set between cluster set and invariant point set.

Secondly, we identify the order in which the watermark bits encoded. For a certain cluster, the order should not be dependent on the 3D information of



(a) (b)

Fig. 2. Identification of clusters (a) Clusters based on segments of a joint trajectory in 3D space. (b) Cluster in joint trajectory for dominant direction determination. $\mathbf{P}_i$ and $\mathbf{P}_j$ are the invariant points in the cluster. $\mathbf{P}_r$ is the midpoint of the line $\mathbf{P}_i\mathbf{P}_j$. $\mathbf{P}_m$ is the point within the cluster which is the most distant to the line $\mathbf{P}_i\mathbf{P}_j$.

the points, as it is vulnerable to noise adding attack. In addition, the order also should not be dependent on the occurrence of 3D point in time, as it is vulnerable to reordering attack. Therefore, we identify ordering criteria based on the occurrence order of 3D points and the dominant direction determined by statistics information of the points belonging to the cluster. The determination of dominant direction make our encoding method robust against global reorder attacks, which means completely inverting the point in a continuous manner in the trajectory. As shown in Fig.2(b), the statistics information are introduced in the following.

Point Numbers. The number of points separated by the line $\mathbf{l}_r$. It is described by N_l and N_r, which represents the number of points separated by line $\mathbf{l}_r$ respectively. In the example, $N_l = 5$ and $N_r = 3$.

Average Gradient. The average gradient of encoding points separated by the perpendicular line $\mathbf{l}_m$. It is described by Ag_l and Ag_r, which represents the module of average gradient separated by line $\mathbf{l}_m$ respectively. The *Average Gradient Ag* is described as

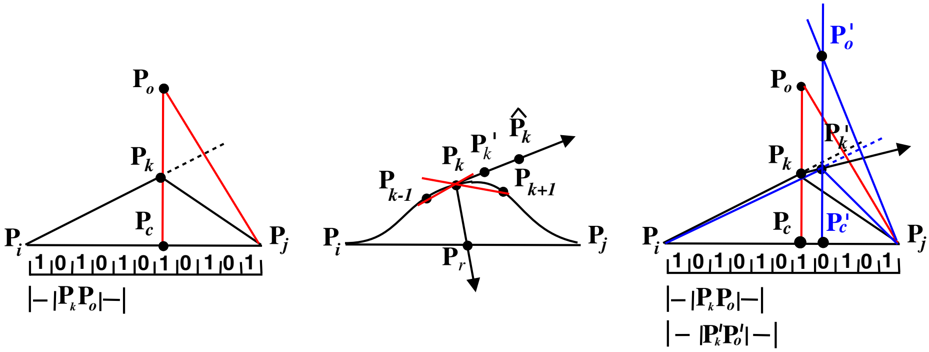
$$Ag = \frac{1}{n}\sum_{i=1}^{n} \left|gradient(\mathbf{P}_i)\right| , \tag{11}$$

where $gradient(\mathbf{P}_i)$ represents the gradient of point $\mathbf{P}_i$, n is the number of points in the left N_l or right N_r side of line $\mathbf{l}_m$.

Average Height. The average height of encoding points separated by the point $\mathbf{l}_m$. It is described by H_l and H_r, which represents average height separated by line $\mathbf{l}_m$ respectively. The *Average Height H* is described as

$$H = \frac{1}{n}\sum_{i=1}^{n} h_i . \tag{12}$$

where h_i represents the Euclidean distance described by the perpendicular distance from point $\mathbf{P}_i$ to the line joining $\mathbf{P}_i\mathbf{P}_j$.



(a)Watermark bit embedding (b) Shifting direction (c) Embedding after shifting

Fig. 3. Triangle Orthocenter based encoding scheme. $\mathbf{P}_o$ and $\mathbf{P}_o^{'}$ are the orthocenters of $\triangle \mathbf{P}_k\mathbf{P}_i\mathbf{P}_j$ and $\triangle \mathbf{P}_k^{'}\mathbf{P}_i\mathbf{P}_j$. $\mathbf{P}_c$ and $\mathbf{P}_c^{'}$ are corresponding pedals in the edge $\mathbf{P}_i\mathbf{P}_j$. $\mathbf{P}_{k-1}$, $\mathbf{P}_k$ and $\mathbf{P}_{k+1}$ are three continuous encoding points, $\hat{\mathbf{P}}_k$ is the predicted position of $\mathbf{P}_k$, $\mathbf{P}_k^{'}$ is the new shifted positions of $\mathbf{P}_k$ in our proposed.

For identifying dominant direction, because *Point Numbers* is robust against most attacks, we identify dominant direction by comparing the *Point Numbers* firstly. If $|N_r - N_l|$ is greater than a certain threshold, it is identified according to the value of N_r and N_l from high to low or vice versa. For example, if $N_r > N_l$, we choose $\mathbf{P}_i \rightarrow \mathbf{P}_j$ as dominant direction. Otherwise, if $|N_r - N_l|$ is less than a certain threshold, it means the point numbers N_r and N_l are at the same level. For this situation, we use $|Ag_l - Ag_r|$ and $|H_l - H_r|$ for further judgment. Once the dominant direction of points in the cluster has been identified, we hash a sequence of encoding points based on the key sk. Due to this randomness, for n encoding points in one cluster, the search space for the adversary is $n!$.

Finally, we encode the watermarking bit in each encoding point using Triangle Orthocenter based encoding approach. For the encoding progress, as shown in Fig.3(a), the Euclidean distance between $\mathbf{P}_k$ and $\mathbf{P}_o$ is used as the scalar quantity to encode a watermark bit information. Here we divide the line $|\mathbf{P}_i\mathbf{P}_j|$ into equal intervals and labeled as 0s and 1s, then measure the scalar quantity $|\mathbf{P}_k\mathbf{P}_o|$ against this scale and locate the interval, e.g. the 1 bit for the $\mathbf{P}_k$ point. If the bit (0 or 1) represented by the encoding point $\mathbf{P}_k$ is the same with the corresponding watermark bit, we move to the next encoding point. Otherwise, $\mathbf{P}_k$ is shifted along the motion direction predicted based upon DR (Dead Reckoning) technique [8], which can compute the predicted position of a joint based on the recent position, velocity and acceleration. For the encoding point $\mathbf{P}_k$, the corresponding predicted position $\hat{\mathbf{P}}_k$ at time t can be calculated as

$$\hat{\mathbf{P}}_k(t) = \mathbf{P}_{k-1}(t-1) + \mathbf{v}_{k-1}(t-1)\tau + 0.5\mathbf{a}_{k-1}(t-1)\tau^2, \qquad (13)$$

where $\mathbf{P}_{k-1}(t-1)$, $\mathbf{v}_{k-1}(t-1)$ and $\mathbf{a}_{k-1}(t-1)$ are the position, velocity and acceleration at the time instant $t-1$ for the last frame. In our example, we make a shift of $\mathbf{P}_k$ towards line $\mathbf{P}_k\hat{\mathbf{P}}_k$ to $\mathbf{P}_k^{'}$, as shown in Fig.3(b). Finally, it can be observed that since the encoding point $\mathbf{P}_k$ is displaced by $\mathbf{P}_k^{'}$ along the line $\mathbf{P}_k\hat{\mathbf{P}}_k$, the bit represented by it is changed from 1 to 0, as shown in Fig.3 (c). Because 3D trajectory is a curvilinear motion, and the $\triangle\mathbf{P}_k^{'}\mathbf{P}_i\mathbf{P}_j$ is obtuse triangle in most case, the Triangle Orthocenter based encoding approach with shifting towards predicted direction $\mathbf{P}_k\hat{\mathbf{P}}_k$ has better imperceptibility than the Euclidean Distance based encoding, which make the shift along line $\mathbf{P}_k\mathbf{P}_r$.

2.4 Watermark Extraction

Given the watermarked 3D motion trajectory data, we can extract a possible existing watermark by firstly identifying the segments using the maxima in spatio-temporal curvature and posterior attack model. Secondly, for each segment, we identify the cluster based on the secret key and extract all the embedded watermark bits in an inverse procedure of watermark embedding. Finally, we compare the detected watermark bit for all watermark copies K $(K \geq 3)$. Due to the randomness and unpredictability for the attack event, we vote for the majority bit value as the final detected data.

3 Experimental Results

In this section, we discuss experimental scheme and performance analysis for our proposed watermarking scheme from three aspects including imperceptibility, hiding capacity and robustness. For each aspect, we analyze the performance for different parameters settings. The watermarking scheme is implemented in MATLAB, and the source data are from CMU-MOCAP database. For the experiment analysis, the frames for motion data are from 448 to 1050 with frame rate being set as 120 fps. We set the watermark length as 32 bits with copy number of 3 for voting, and they are embedded into one cluster for each copy. Moreover, we conduct a comparison experiment between our method and Euclidean Distance based method for performance analysis.

3.1 Performance Analysis Metrics

In this section, we illustrate the metrics from the aspects of imperceptibility, hiding capacity and robustness used to evaluate our watermarking scheme.

For imperceptibility analysis, we use the following metrics:

$$DistortionforDistance = \frac{\sum_{i=1}^{m}\sum_{k=1}^{n} Euclidean(\mathbf{P}_{i,k}, \mathbf{P}_{i,k}^{'})}{m * n}, \qquad (14)$$

where $DistortionforDistance$ represents the error for the distance between original position $\mathbf{P}_{i,k}$ and modified position $\mathbf{P}_{i,k}^{'}$ in the k-th frame of i-th joint. m is frame number and n is joint number. Besides the distortion for distance, considering the motion direction of joint, define the angle metric as:

$$AverageShiftAngle = \frac{\sum_{i=1}^{m}\sum_{k=2}^{n-1}(\pi - \angle\mathbf{P}_{i,k-1}\mathbf{P}_{i,k}^{'}\mathbf{P}_{i,k+1})}{m * n}, \qquad (15)$$

where $\angle\mathbf{P}_{i,k-1}\mathbf{P}_{i,k}^{'}\mathbf{P}_{i,k+1}$ represents the trend of the movement direction.

Hiding capacity is measured by embedding rate, which is the ratio of the number of embedding bits to the total number of sampling points, as shown by:

$$ER = \frac{\sum_{i=1}^{s} Bits_i}{m} = \frac{\sum_{i=1}^{s} \lfloor \frac{S_i}{CSize} \rfloor (CSize - 2)}{m}, \tag{16}$$

where s is segment numbers, $Bits_i$ is the bit number that can be embedded in *i-th* segment, S_i is the length of *i-th* segment, and $CSize$ is cluster size.

The robustness can be evaluated by the following metrics:

Watermark Detection Rate (WDR). This is measured as the ratio of the number of watermark bits correctly detected to the original number of watermark bits encoded, which is defined as:

$$WDR = \frac{\sum_{i=1}^{wsize} \neg(w(i) \oplus w^{'}(i))}{wsize}, \tag{17}$$

where $w(i)$ and $w^{'}(i)$ represent *i-th* bit of original and detected watermark respectively, $wsize$ represents watermark length. Due to proper watermark replication technology, $w^{'}(i)$ can be acquired by voting from exacted bit copies.

Bit Error Rate (BER). This is measured as the ratio of the number of error bits extracted to the total number of watermark bits encoded in the given data set, which is defined as:

$$BER = \frac{\sum_{i=1}^{r*wsize} errorBits}{r * wsize}, \tag{18}$$

where $errorBits$ represents the number of error bits extracted, r represents the number of watermark copies, $wsize$ represents the length of watermark in bits.

3.2 Imperceptibility Analysis

As the distortion induced by encoding in our method is dependent on the scale (number of intervals), the imperceptibility analysis with varying number of intervals is shown in Fig.4. For both the distance and angle distortions, as we increase the number of intervals, the distortion is reduced. For the same scale, as shown in Fig.4(a), since we make a shifting of encoding points towards the predicted direction during our proposed encoding method, the $DistortionforDistance$ of Triangle Orthocenter based encoding (represented by TO) is consistently smaller than that of encoding based on Euclidean Distance (represented by ED). And for angle distortion, the difference of $AverageShiftAngle$ between our proposed and the original data without watermarking is also smaller, which is shown in Fig.4(b). Moreover, more intervals results in sensitivity to noises, and this will be discussed in section 3.4.

3.3 Hiding Capacity Analysis

Just as the cluster-based encoding scheme such as Euclidian Distance based strategy, we can encode a maximum of t-2 bits for a given cluster of size t.

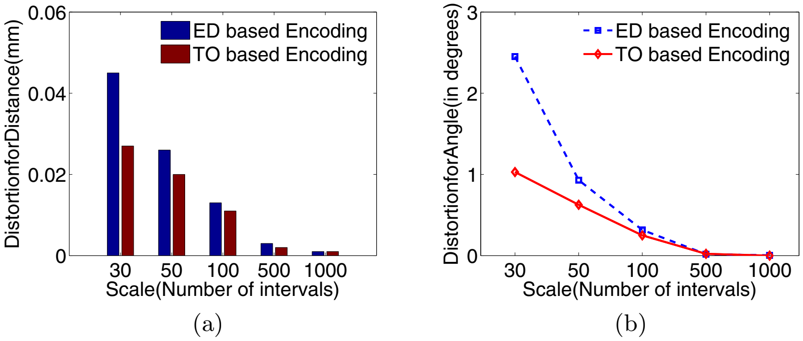


Fig. 4. Imperceptibility analysis for varying number of intervals("dance")

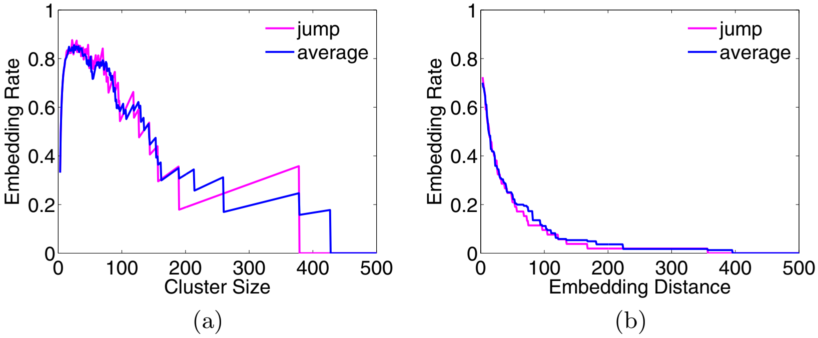


Fig. 5. Embedding rate analysis with respect to cluster size (a) or embedding distance (b)("average" means the value for different motion data)

As the hiding capacity is dependent on the cluster size and embedding distance, we test the influence of the two parameters on embedding rate. As shown in Fig.5(a), in the beginning, the gains from increased number of embedded bits per cluster dominate. However, this dominance fades quickly when the cluster is sizable. The zigzag effects are due to the integer and floor operation as in Eq.(16). Finally, when the cluster size is larger than any segment, the embedding rate is zero. With the optimal cluster size which maximizes the embedding rate, we test the performance of embedding rate with respect to embedding distance. As shown in Fig.5(b), embedding rate reduces as the distance increases, although not smoothly.

3.4 Robustness Analysis

In this section, the robustness against possible attacks including uniform affine transforms, cropping, noises addition and reordering of our proposed watermarking scheme is analyzed. And the noise attacks are done by adding four groups of attack including white Gaussian noise (10db to 1db) and the smooth attacks

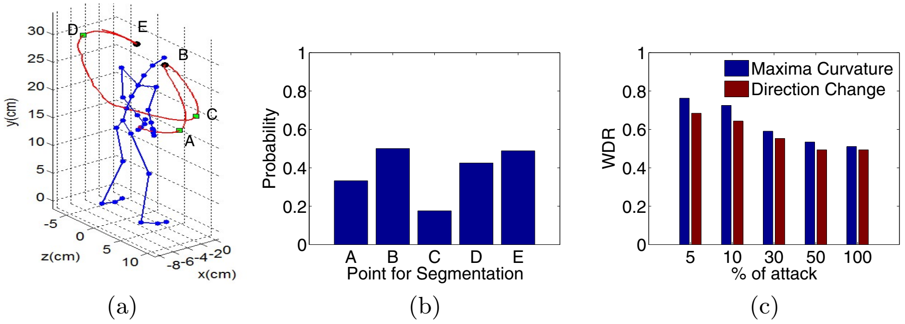


Fig. 6. Robustness analysis for segmentation

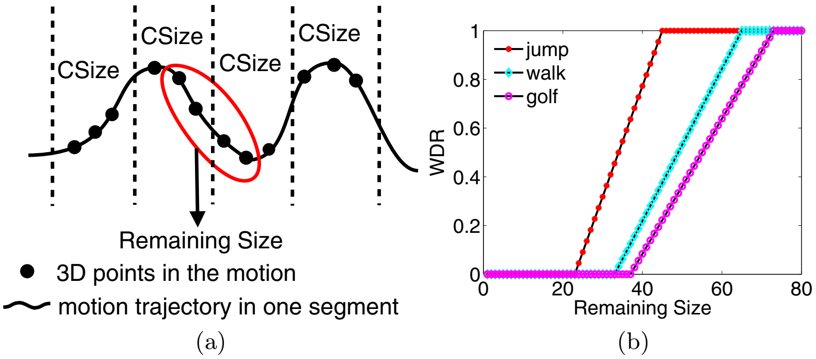


Fig. 7. Robustness analysis for cropping attacks (CSize for "jump"=22, CSize for "walk"=32, CSize for "golf"=36)

(represented by *smooth*1 and *smooth*2) on various ratios (5% to 100%). Here *smooth*1 and *smooth*2 are defined in [2]. The analysis for noise addition and reordering attacks take the golf motion (Scale=50) for example. We get the mean value for the four groups of attacks as the final result.

Segmentation Robustness. In order to identify robust segments, we make a segmentation of motion data with stable anchor-points captured by the maxima in spatio-temporal curvature and posterior attack model. While in [2], it records segments by identifying the change in angle direction of joint, which is shown in Fig.6(a). In order to measure its robustness, we conduct posterior attack and compute the probability for each point. As shown in Fig.6(b), the probabilities of points A, C and D are comparatively lower, and they may result in sensitivity to noises. However, in our segmentation method, the unstable points have been filtered by posterior attack model. The robustness analysis against noise attack with our "Maxima Curvature based segmentation" and "Direction Change based segmentation" is shown in Fig.6(c).

Uniform Affine Transforms. Since stable anchor point, and thus the segments and clusters are invariant to 3D affine transform action. In addition, the Triangle Orthocenter based encoding approach preserves collinearity and ratios of

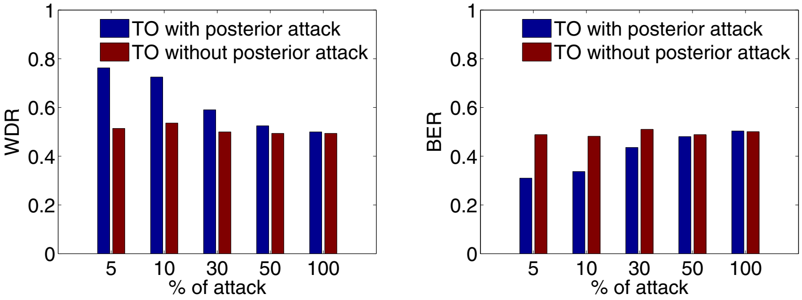


Fig. 8. Robustness analysis against noise addition attacks for Triangle Orthocenter based encoding with and without posterior attack model

distances between the points in clusters. The experiments verify that our scheme is completely robust against uniform affine transformation (translation, scaling and rotation).

Cropping Attacks. Since we embed watermarks into each joint trajectory independently, our scheme is robust against the joint cropping. That is, if one joint of the motion is preserved during the cropping, we can still detect the watermark. When the cropping occur inside one joint trajectory, the robustness against such attack depends on its presence in segments. Intuitively, in order to detect the watermark, the remaining part of a trajectory must be large enough to contain the watermarked points in one cluster. For analysis, assume watermarks are completely embedded in a cluster with the size represented by $CSize$, and the cluster is divided continuously in the segment, as shown in Fig.7(a). Fig.7(b) demonstrates the relation between watermark detection ratio (WDR) and the varying cropping size. Just as the Euclidean Distance based encoding method[2], the minimum remaining size $CropSize$ required to maximize the watermark detection ratio is limited by twice the cluster size $2 * CSize$.

Noise Addition Attacks. For robustness analysis for noise addition attacks, the comparison of our proposed watermarking method with and without posterior attack model is shown in Fig.8. The WDR increases from 0.50752 on average to 0.62064 by introducing the posterior attack model. Accordingly, the BER reduces from 0.49404 on average to 0.41344. Fig.9 illustrates the robustness analysis of our proposed Triangle Orthocenter based encoding (represented by TO) and Euclidean Distance based encoding (represented by ED) for both using the posterior attack model. As is shown, our scheme illustrates better robustness due to position independent ordering criteria of encoding points. The order based on position information in Euclidean Distance based encoding may be disturbed by noise addition. Moreover, the robustness against noise addition attacks can be improved by choosing smaller scales, which is shown in Fig.10. The robustness decreases for both method with larger scale, and our scheme is also better for a range of scales.

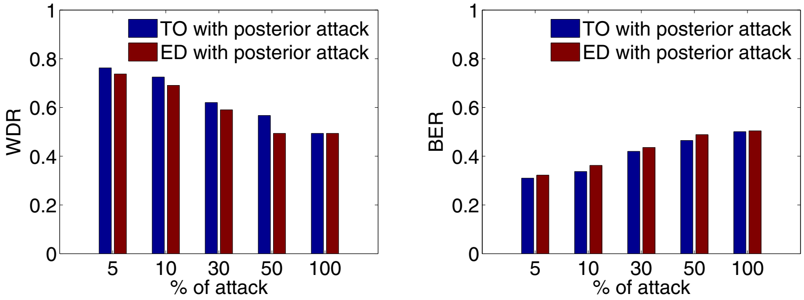


Fig. 9. Robustness analysis against noise addition attacks for Triangle Orthocenter based encoding and Euclidean Distance based encoding

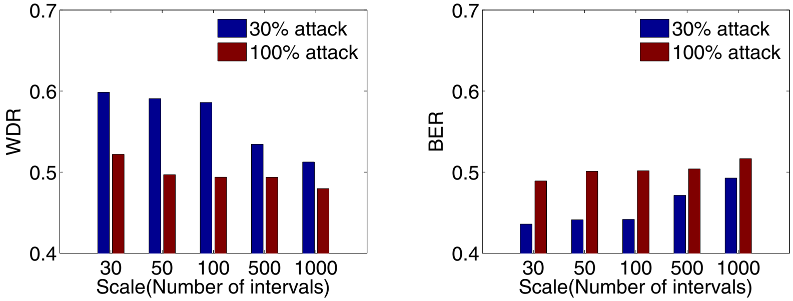


Fig. 10. Robustness analysis for noise addition attacks with varying number of intervals

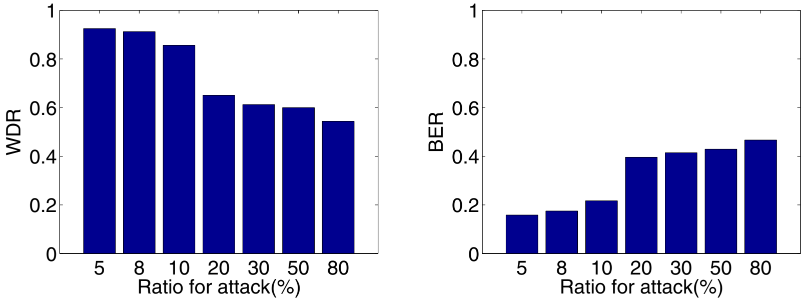


Fig. 11. Robustness analysis against reordering attacks with different attack ratios

Reordering Attacks. The reordering attacks are done either by completely inverting the point in the continuous section of the trajectory, or by randomly picking two neighboring points and swapping them. Because we determine the dominant direction of cluster based on the statistics information of encoding points, and the difference of these statistics information are almost robust against global reordering, so our proposed watermarking scheme is completely robust for global reordering in cluster level. For the random reorder attacks, Fig.11 illustrates robustness analysis for random reordering attacks to our watermarking scheme. Due to watermark replication technology adopted, we can see that with proper number of copies, our scheme is also highly robust against lower ratio partial random reordering attacks for the semantic preserved motion data.

4 Conclusions

This paper has presented a blind robust watermarking mechanism based on maxima curvature of 3D motion data. Analysis proves that the scheme has better imperceptibility by shifting the encoding point towards motion direction predicted based upon its recent position, velocity and acceleration. The hiding capacity has bounds based on cluster size. Moreover, experimental results show that our method is robust against many possible attacks such as uniform affine transforms, reordering and cropping. And particularly, it is more robust than the state of the art against noise addition attacks due to the stability of anchor-points used for segmentation and the position independent ordering criteria of encoding points in our proposed Triangle Orthocenter based encoding method.

Acknowledgments. This work was supported by National Natural Science Foundation of China (No.60905019), Tianjin Key Technologies R&D program (No.11ZCKFGX00800), Tsinghua-Tencent Joint Laboratory for Internet Innovation Technology, SKL of CG&CAD, and Strategic Priority Research Program of the Chinese Academy of Sciences(XDA06030601).

References

1. Agarwal, P., Adi, K., Prabhakaran, B.: Robust blind watermarking mechanism for motion data streams. In: Proceedings of the 8th Workshop on Multimedia and Security, pp. 230–235 (2006)
2. Agarwal, P., Prabhakaran, B.: Blind robust watermarking of 3d motion data. ACM Transactions on Multimedia Computing and Communication Applications 6(1), 1–32 (2010)
3. Arnold, M., Chen, X.-M., Baum, P.G., Doërr, G.: Improving Tonality Measures for Audio Watermarking. In: Filler, T., Pevný, T., Craver, S., Ker, A. (eds.) IH 2011. LNCS, vol. 6958, pp. 223–237. Springer, Heidelberg (2011)
4. Bors, A.: Watermarking mesh-based representations of 3-d objects using local moments. IEEE Transaction Image Process. 15(3), 687–701 (2006)
5. Cho, J., Prost, R., Jung, H.: An oblivious watermarking for 3-d polygonal meshes using distribution of vertex norms. IEEE Transaction Signal Process. 55(1), 142–155 (2007)
6. Cox, I.J., Miller, M.L.: The first 50 years of electronic watemrarking. Joumal of Applied Signal Proeessing 10(2), 126–132 (2002)
7. Dugelay, J.L., Roche, S., Rey, C.: Still-image watermarking robust to local geometric distortions. IEEE Transaction on Image Processing 15(9), 2831–2842 (2006)
8. Duncan, T., Gracanin, D.: Pre-reckoning algorithm for distributed virtual environments. In: Proceedings of the 2003 Winter Simulation Conference, pp. 1086–1093 (2003)

9. Gao, X., Deng, C., Li, X., Tao, D.: Geometric distortion insensitive image watermarking in affine covariant regions. IEEE Systems, Man, and Cybernetics Society 40(3), 278–286 (2010)
10. Kim, T., Lee, J., Shin, S.: Robust motion watermarking based on multiresolution analysis. Computer Graphics Forum 19(3), 189–198 (2000)
11. Lee, H., Kim, H., Lee, H.: Robust image watermarking using local invariant features. Optical Engineering 45(3), 1–11 (2006)
12. Li, S., Okuda, M.: Iterative frame decimation and watermarking for human motion animation. Vision and Image Processing, Special Issue on Watermarking 2010(7), 41–48 (2010)
13. Li, W., Xue, X., Lu, P.: Localized audio watermarking technique robust against time-scale modification. IEEE Transaction on Multimedia 8(1), 60–69 (2006)
14. Luo, M., Bors, A.G.: Principal component analysis of spectral coefficients for mesh watermarking. In: International Conference on Image Processing, pp. 441–444 (2008)
15. Luo, M., Wang, K., Bors, A., Lavoué, G.: Local Patch Blind Spectral Watermarking Method for 3D Graphics. In: Ho, A.T.S., Shi, Y.Q., Kim, H.J., Barni, M. (eds.) IWDW 2009. LNCS, vol. 5703, pp. 211–226. Springer, Heidelberg (2009)
16. Motwani, R., Bekris, K., Motwani, M., Harris, F.: Fragile watermarking of 3d motion data. In: International Conference on Computer Applications in Industry and Engineering, vol. (1), pp. 111–116 (2010)
17. Pankajakshan, V., Doerr, G., Bora, P.: Assessing motion coherency in video watermarking. In: ACM Multimedia and Security, pp. 114–119 (2006)
18. Rao, C., Yilmaz, A., Shah, M.: View-invariant representation and recognition of actions. International Journal of Computer Vision 50(2), 203–226 (2002)
19. Rondao-alface, P., Macq, B.: Blind watermarking of 3d meshes using robust feature points detection. In: International Conference on Image Processing, vol. 1, pp. 693–696 (2005)
20. Su, K., Kundur, D., Hatzinakos, D.: Statistical invisibility for collusion-resistant digital video watermarking. IEEE Transaction on Multimedia 7(1), 43–51 (2005)
21. Wang, X., Zhao, H.: A novel synchronization invariant audio watermarking scheme based on dwt and dct. IEEE Transaction on Signal Processing 54(12), 4835–4840 (2006)
22. Yamazaki, S.: Watermarking motion data. In: Proceedings of Pacific Rim Workshop on Digital Steganography, pp. 177–185 (2004)
23. Yang, S., Song, Z., Fang, Z., Yang, J.: A novel affine attack robust blind watermarking algorithm. In: Symposium on Security Detection and Information Processing, vol. 7, pp. 239–246 (2010)
24. Zhang, J., Ho, A.T.S., Qiu, G., Marziliano, P.: Robust video watermarking of h. 264/avc. 264/avc. IEEE Transaction on Circuits and Systems 54(2), 205–209 (2007)