An Improved Linear Blending Skinning Algorithm Based on Human Joint Position Relation and Improved Weight Calculation Method

Hui-bai WANG, Jia-wei REN* and Hao YU
North China University of Technology, Beijing, China
*Corresponding author

Keywords: Skeleton Animation, LBS, Weight calculation.

Abstract. Based on real-time requirements and joint position information in motion, this paper improves LBS algorithm because of its highest computational efficiency. The weight calculation method in traditional LBS algorithm is improved into a pair of values—the ratio of the length of the vertex’s projected position on the bone to the length of the bone and Cosine of the Angle between the attachment of the vertex to the parent joint and the bone containing the parent joint—to overcome the wrap candy paper and articulatory place collapse. After extracting the skeleton structure of the model and calculating the weight, by monitoring the position change relation of the local key joints in the motion sequence, the traditional LBS algorithm and the improved LBS algorithm can be dynamically switched to reduce the computation as much as possible and improve the algorithm efficiency. Experimental results show that the improved algorithm proposed in this paper overcomes the defects of LBS algorithm effectively, and the method is simple and effective.

Introduction

Character Animation is mainly divided into Vertex Animation and Skeletal Animation. Among them, skeletal animation is the main research content in the field of character animation [1]. Compared with vertex animation, it takes up less space but can present the motion of the model smoothly, which makes it more widely used, especially in the application of high-precision skin model. The skeleton animation transforms the vertices from model space to skeleton space, affects the skin vertices to translate and rotate when the bones generate displacement and rotation transformation, and then inverts the transformed skin vertices from skeleton space to model space to convert the skin vertices back to model space.

Character animation is generally divided into the following parts:
(1) skeleton embedding: set a skeleton structure for the given model and make the skeleton match the skin.
(2) skin covering process: calculate the weight of skin vertex relative to individual nodes, and use skin covering algorithm to realize bone-driven skin model movement and skin deformation effect.
(3) set motion constraints: the skeleton is controlled to move by using pre-set motion constraints or Inverse Kinematics to generate motion sequences.

Related Technologies

This section will introduce the work of providing technical background and relevant theoretical knowledge for this paper from the aspects of skin technology and vertex weight setting.

Bone-based Skin Technology

Bone-based skin technology mainly includes LBS algorithm proposed by Magnenant et al. [2] and DQS algorithm proposed by Kavan et al. [3] Among them, LBS technology algorithm is simple and highly efficient, but it has the adverse effect of large volume loss of deformation results (wrap candy paper effect and joint collapse, etc.). DQS algorithm does not have the defects of LBS, but the deformation results have the adverse effect of joint bulge, and the calculation efficiency is not as high as LBS so that it is inferior to LBS algorithm in the context with high real-time requirements.
Weight Calculation Method

Skin weight calculation is also a very important work in character animation. Dionne et al. [4] proposed an automatic weight calculation method based on geodesic distance, and Ju et al. [5] proposed a skin template based on bounding box. Dicko[6] et al. realized the relocation of attributes such as skeleton structure and skin fixed point weight. However, these methods are dependent on the specified model template and cannot be applied to any model object that is supplied at once, which has limitations.

Model Skeleton Structure is Obtained

Laplacian point cloud contraction is a relatively famous point cloud smooth and contraction method. Its contraction process is also very effective for the extraction of point cloud curve skeleton, but it also has some limitations—local area will over-contraction. The global Laplacian point cloud contraction inevitably leads to this problem. Therefore, this paper adopts the local Laplace contraction method to extract the skeleton. The extraction method is as follows:

Mark all initial point clouds as unstable. The initial skeleton extraction radius is calculated according to the bounding box of point cloud. Laplacian contractions were performed for all point clouds in unstable states, and points in line with the extraction radius were marked as skeleton points. A skeleton branch is obtained by connecting the existing skeleton points. Update the extraction radius for the remaining unstable points and traverse all points to iteratively perform point cloud contraction and skeleton point extraction until all points are not in an unstable state. Get all the skeleton branches and output skeleton curve.

Node extraction is carried out by human body proportion. When measured by geodesic distance, the relative length of each part of the human body does not change much. As shown in figure 1, L is the geodesic distance between two hands. According to figure 1, the corresponding positions of each joint can be found on the skeleton curve.

![Figure 1. Skeleton scale of human body.](image)

Skin Weight Binding

The traditional weight calculation method is bone projection method. However, this method is only applicable to skin deformation of limbs and trunk with small motion range. When large joint movements occur, the skin bulges severely.

In the rigid change area, the deformation of the skin apex is affected by only one bone. The traditional way is to use 0 and 1 as criteria for whether or not it affects. When the weight is 1, it is affected; when it is 0, it is not affected. The method adopted in this paper is to adopt a set of values as the weight, the ratio of the length of the vertex's projected position on the bone to the length of the bone and Cosine of the Angle between the attachment of the vertex to the parent joint and the bone containing the parent joint. When the first value in the weight is greater than 0, it is equivalent to the weight 1 in the traditional method. The advantage of the sample representation is that in addition to being able to determine the skin vertex is affected by the bone, it can also determine the relative position of the skin vertex and the bone.
The non-rigid change area is usually the elbow joint and the knee joint. Due to the human body joint point can only be at most 180° bend and the direction is fixed, the skin that is susceptible to deformation problems is the skin on the inside of the bent joint, and the skin on the outside of the bent joint is negligible. As a result, the skin on the inside of the bent joint is affected by the two bones. For such nodes, the same weight calculation method is used. The difference is that the skeleton curve will be bent at an Angle on the inside of the bent joint. The skin on the medial side of the bent joint is affected by this Angle. According to the weight calculation method used in this paper, the weight of both bones will be calculated. The lateral skeleton has no such effect, and only affected by only one bone.

The specific method of weight calculation is as follows:

As shown in figure 2, V1 and V2 are skin vertices, J0 is the parent joint of J1, and J1 is the parent joint of J2. For the skin fixed point V1, relative to the bone J0J1, calculate the cosine of the included Angle V1J0J1 and the included Angle V1J1J0. When both cosine values are greater than 0, the Angle between V1J0J1 and V1J1J0 are acute. It means V1 is projected onto the skeleton and is affected by the skeleton J0J1. Let the distance of V1’s projection onto skeleton J0J1 to joint J1 be S, and the weight of V1 affected by skeleton J0J1 be P. The calculation of S and P is as follows:

\[ S = \text{distance}(J_0J_1) - \cos \angle V1J_0J_1 \times \text{distance}(V1, J_0) \]  
(1)

\[ P = \begin{cases} \frac{S}{\text{distance}(J_0J_1)}, & \cos \angle V1J_0J_1 \end{cases} \]  
(2)

**Improvement of LBS Algorithm**

Based on real-time consideration, this paper improves LBS algorithm on the basis of LBS algorithm. The sugarcoated paper effect caused by large Angle rotation and the collapse of joint parts caused by large joint bending are caused by the fact that the skin volume is not well controlled and supported, which causes the skin to collapse. To solve these problems, this paper uses the new weight calculation method in 3.1 to improve the LBS algorithm.

**Improvement of Wrapped Candy Wrappers**

Sugar-coated paper effect mainly occurs when the joint rotation Angle is too large. As shown in figure 3, when the skin rotates violently, V is affected by the two bones M1 and M2, and the newly generated skin vertex V may lose the information of the length of the vector in the model space, resulting in the skin distortion. Especially when rotation Angle more than 60° this change is especially striking.
So when the rotation Angle is less than 60°, using the traditional LBS algorithm to carry on the
skin. When rotation Angle is greater than 60°, adopt the new weight calculation method to
calculate the skin deformation. Algorithm is as follows:

(1) Comparing the obtained motion information with the initial skeleton information, if the
rotation Angle is less than 60°, the skin was covered by the traditional LBS skin algorithm, and
Skin weight \( P \) only takes the first number in a pair of data and round up; If the rotation Angle is
greater than 60°, into (2) or (3).

(2) In order to maintain the skin volume, the weight \( P \) obtained during skin binding should be
used to calculate the distance from the skin vertex to the bone. The first data of \( P \) is the ratio of the
length of the vertex projected to the parent node on the skeleton to the length of the skeleton and the
second data point is the Cosine of the Angle between the attachment of the vertex to the parent joint
and the bone containing the parent joint. For example, the weight of point \( V_1 \) in figure 2, multiply
the first number by the length of the bone can get value \( S \), the length of the vertex projected onto
the skeleton to the parent node; because the second data point is cosine of \( \angle V_1J_0J_1 \), we can figure
out the tangent of \( \angle V_1J_0J_1 \), \( S \) times the tangent of \( \angle V_1J_0J_1 \) is the distance from the vertex to the bone.

(3) When affected by two bones, the skin vertex is closer to the joint Shared by the two bones. At
this point, the distance from the vertex to the skeleton is changed to the distance from the vertex to
the node. Such as \( J_1 \) node in figure 2, using the weight of vertex and skeleton \( J_1J_2 \) to get the
distance, \( \frac{S}{\cos \angle V_2J_1J_2} \), from the vertex to the Shared node.

(4) The position of the vertex was obtained by traditional LBS calculation, and the distance
obtained by (2) and (3) was modified to maintain the volume and greatly reduce the degree of
sugar-paper effect.

**Improvement of Joint Collapse**

The cause of joint collapse is roughly the same as that of the sugar-coated paper effect, both of
which result in unnecessary scaling and translation during linear weighting, failing to maintain a
good skin volume. As shown in FIG. 4, \( V_1 \) and \( V_2 \) are the positions obtained after the vertex is
transformed under the influence of two different skeletons respectively, and \( V \) is obtained by simply
weighting \( V_1 \) and \( V_2 \). But \( V \) loses its length after the transformation, and the position of \( V \) prime
should be the position of its original length. So the joint collapses.

![Figure 4. Causes of Joint collapse.](image)

Prone to collapse are the elbow and knee joints. Similar to the solution to the sugar-coated paper
effect, the weights in this paper are used to calculate the distance from the vertex to the bone or the
Shared node to correct the position to keep the volume stable. The specific algorithm is as follows:

(1) Comparing the obtained motion information with the initial skeleton information, if the Angle
between the two bones sharing elbow joint or knee joint is more than 60°, the skin was covered by
the traditional LBS skin algorithm, and Skin weight \( P \) only takes the first number in a pair of data
and round up; If the angle is less than 60°, into (2) or (3).

(2) The same as algorithm of 5.1(2), calculate the distance from the vertex to the bone.
(3) The same as algorithm of 5.1(3), when affected by two bones, using the weight of vertex to get the distance from the vertex to the Shared node.

(4) The position of the vertex was obtained by traditional LBS calculation, and the distance obtained by (2) and (3) was modified to maintain the volume and greatly reduce the degree of the skin of the joint collapses.

Results

This experiment based on visual studio 2017 joint OpenGL graphics library development skin deformation process, computer hardware by Intel ® Xeon ® E5-2670 v2 @ 2.50 GHz CPU, memory, 64 g graphics for NVIDIA Quadro K5000. In the experiment, the skeleton system of the skin model was firstly obtained through the skeleton extraction algorithm, and 16 human body nodes were generated. According to the skeleton system and skin model, the skeleton and skin vertices are bound by the weight calculation method proposed in this paper. In the experiment, two situations where LBS algorithm is prone to distortion are simulated, and the algorithm in this paper is compared with LBS algorithm. The experimental results are shown in figure 5 and figure 6. Through the improvement of LBS algorithm in this paper, better experimental results are obtained.

Figure 5. When right arm rotated 180 ° LBS algorithm rendering (left) Algorithm rendering of this paper (right).

Figure 6. Elbow joint collapse bending at 90 ° LBS algorithm rendering (left) Algorithm rendering of this paper (right).

Summary

In this paper, based on local Laplace contraction, the skeleton structure of the skin model is obtained and 16 joints are identified. By improving the traditional weight allocation algorithm, the traditional LBS algorithm and the modified LBS algorithm are dynamically switched based on the joint position information (Angle of the joint system, joint rotation Angle, etc.). While ensuring the speed and computational efficiency as much as possible, the problem of skin distortion at joint joint was effectively solved, and the defects of traditional LBS algorithm in wrap candy paper effect and joint collapse were overcome, so as to achieve the objective of realistic deformation of human skin. Experimental results show that the improved algorithm proposed in this paper overcomes the defects of LBS algorithm effectively, and the method is simple and effective. Next, on the basis of
the weight calculation method proposed in this paper, we will try to simulate the muscle changes during human movement.

References

[1] Ismail I, Sunar M S, Basoriah, Dynamic Motion Control Editor for Virtual Huma, J. International Journal of Interactive Digital Media, 2013, 1(1) :14-17.

[2] Magnenat-Thalmann N, Laperrière R, et al, Joint-dependent local deformations for hand animation and object grasping,, J. Proceedings Graphics Interface, 1988: 26-33.

[3] Kavan L, Collins S, Zăra J, et al. Geometric skinning with approximate dual quaternion blending, J. ACM Transactions on Graphics, 2008, 27(4): 995-999.

[4] Dionne O, Lasa M. D. Geodesic voxel binding for production character meshes, J.ACM Siggraph/eurographics Symposium on Computer Animation, 2013: 173-180.

[5] Ju T, Zhou Q. Y, Panne M, et al. Reusable skinning templates using cage-based deformations, J. ACM Transactions on Graphics, 2008, 27(5): 32-39.

[6] Ali-Hamadi D, Liu T, Gilles B, et al. Anatomy transfer, J. ACM Transactions on Graphics, 2013, 32(6):188.

[7] Wang Xiaojie, The Research on Skeleton Extraction of Three-Dimensional Point Cloud, D Shandong University, 2017.

[8] Kavan L, Zara J. Spherical blend skinning: a real-time deformation of articulated models. Interactive 3D Graphics and Games, J.ACM Press, 2005:1-10.

[9] Merry B, Marais P, Gain J. Animation space: a truly linear framework for character animation, J. ACM Trans. Graph., 2006,2 5(4):1400-1423.