

# Pose Space Deformation

Veronica Lee

Yifan Wang

## Abstract

*Skeletal animation is a technique used for character animation in computer graphics, where a character is represented as a hierarchical model consisting of bones and skins enveloping the bones. Skinning methods focus on how to appropriately envelope the bones with a skin mesh. In this paper, we discuss the implementation and comparison of two example-based skinning methods: skeletal subspace deformation (SSD) and pose space deformation (PSD). We also discuss our attempt to compute joint weights using the pose samples.*

## 1. Introduction

Skeletal animation is a technique used for character animation in computer graphics, where a character is represented as a hierarchical model consisting of bones and skins enveloping the bones. Skinning methods focus on how to appropriately envelope the bones with a skin mesh. There are three common approaches to skinning: 1) algorithmic, 2) physically-based, and 3) example-based. Purely algorithmic or physically-based skinning methods can have shortcomings because it is difficult to accurately model the complex biomechanics of skin deformation. Example-based methods require first creating a set of examples, but the results are likely to be more realistic.

Skeletal subspace deformation (SSD) [4] is a commonly used example-based method for skinning that uses a weighted transformation from the bind pose. Pose space deformation (PSD) [2] is another example-based skinning method that performs better than SSD, especially for poses at sharp angles. In this paper, we will discuss the implementation and comparison of SSD and PSD, primarily by following the implementation as described in [1]. We will also discuss our attempt to compute joint weights using the pose samples defined for PSD [5, 6].

## 2. Background Work

SSD is a method of linear blend skinning described by Thalmann et al [4] that is now used widely in many commercial software packages. This method is popular because the computations are simple and fast. However, it encoun-

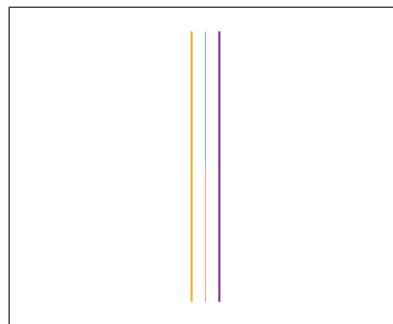


Figure 1. The bind pose.

ters strange behavior at sharp angles, including collapsing skin at bend elbows and rotated shoulders. We used the knowledge acquired from 6.837 lectures and homework assignment 2 to reimplement SSD in this paper.

More advanced example-based skinning methods like PSD tackle these issues by using a set of arbitrary poses defined where the strange deformations happen. The implementations in other papers [2, 3] have been shown to handle complex skinning effects like muscle bulges and major wrinkles. We referenced the PSD algorithm in [1, 2] to reimplement PSD in this paper.

Both SSD and PSD rely on joint weights to be defined for each vertex on the skin, in order to bind the skin mesh to the bone hierarchy. A method of computing joint weights from samples using a least-squares based estimation has been shown in [5, 6]. We referenced the least-squares estimation as described in [1, 5, 6] to reimplement the method in this paper.

## 3. Related Work

Weighted pose space deformation (WPSD) is an even more advanced example-based skinning method, which were derived from a hand model using medical images [3]. WPSD is able to generate realistic skinning even with a limited number of poses. [1] expands on this method by implementing parallel WPSD using fragment processors. This approach only works for poses with more than one degree of freedom, so we were not able to implement WPSD with the examples used for this paper.

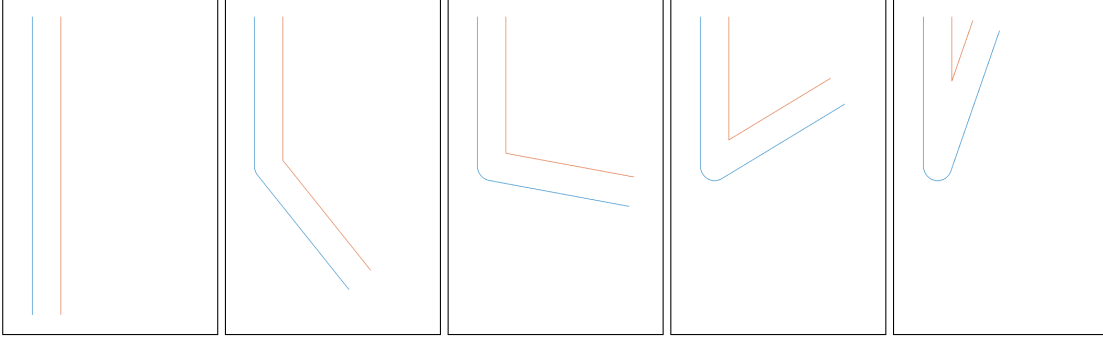


Figure 2. The five example poses defined at  $\theta = 180^\circ, 140^\circ, 100^\circ, 60^\circ, 20^\circ$ .

## 4. Methodology

We used MATLAB implement in code and generate images.

For the implementations of SSD and PSD in this paper, we created a simple bone structure consisting of two bones and one degree of freedom. The resulting bind pose is shown in Figure 1 (below). Let  $\theta$  be the angle between the two bones, so that  $\theta = 180^\circ$  in the bind pose. To test the implementations of SSD and PSD, we varied the value of  $\theta$ . We kept the upper bone (in blue in Figure 1) fixed, while moving the lower bone (in red in Figure 1).

Joint weights needed to be defined for both SSD and PSD. Let  $w(v) = (w_u, w_l)$  be defined as the weight distribution for a vertex  $v$  on the skin, where  $w_u$  is the weight corresponding to the upper bone and  $w_l$  is the weight corresponding to the lower bone. We defined the weights to be as follows:  $w(v) = (1, 0)$  if  $v$  is on the upper half of the upper bone,  $w(v) = (0, 1)$  if  $v$  is on the lower half of the lower bone,  $w(v)$  is a linear distribution from  $(1, 0)$  to  $(0.5, 0.5)$  for the vertices on the lower half of the upper bone, and  $w(v)$  is a linear distribution from  $(0.5, 0.5)$  to  $(0, 1)$  for the vertices on the upper half of the lower bone, so that the vertex connecting the two bones has weight  $(0.5, 0.5)$ . This setup also ensures that  $w_u + w_l = 1$  for every vertex on the skin.

## 5. Skeletal Subspace Deformation (SSD)

The general equation for SSD is

$$v(p_a) = S(v_0) \quad (1)$$

where  $p_a$  is an arbitrary pose,  $v(p_a)$  is a vertex of a deformed target of  $p_a$ ,  $v_0$  is the corresponding vertex in the bind pose, and  $S$  is the SSD function.

Using the bind pose and joint weights as defined in the methodology, SSD computes the position of each skin vertex by using the weighted blending of an affine transformation of each joint from the bind pose. Then we can define  $S$

to be

$$S(v_0) = \left( \sum_{j=1}^{n_{joint}} w_j T_j \right) v_0, \quad (2)$$

where  $n_{joint}$  is the number of joints,  $w_j$  is the predefined joint weight, and  $T_j$  is joint  $j$ 's transformation to the current vertex.

## 6. Pose Space Deformation (PSD)

The general equation for PSD is

$$v(p_a) = S(v_0 + D(p_a)), \quad (3)$$

where  $D(p_a)$  is the displacement as a function of the arbitrary pose.

PSD requires a set of arbitrary poses to first be defined where the strange deformations happen. We defined five example poses at  $\theta = 180^\circ, 140^\circ, 100^\circ, 60^\circ, 20^\circ$ . These predefined poses are shown in Figure 2. Given this set of example poses, we can interpolate to calculate  $D(p_a)$  in the ‘‘pose space’’ [2].

We first need to compute the displacements of the example poses in the rest coordinates to be able to interpolate on them. We can do this by using inverse SSD on each example pose  $k$ :

$$d_k = \left( \sum_{j=1}^{n_{joint}} w_j T_j \right)^{-1} v_k - v_0 \quad (4)$$

We used the radial basis function to interpolate the displacement for the arbitrary pose:

$$D(p_a) = \sum_{k=1}^{n_{pose}} \lambda_k \phi(\|x - x_k\|) \quad (5)$$

where the radial basis function was defined to be

$$\phi(x) = e^{-(\epsilon x)^2} \quad (6)$$

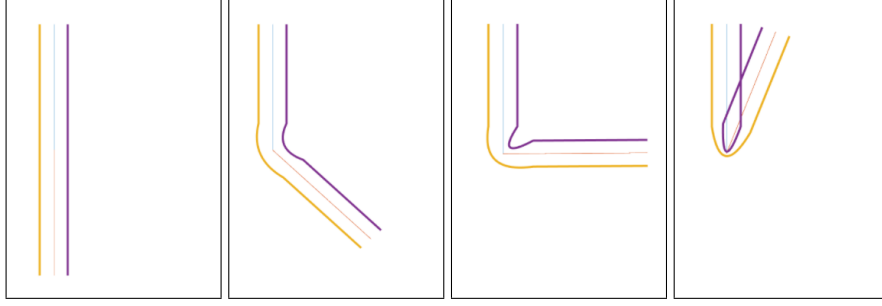


Figure 3. The results for SSD at  $\theta = 180^\circ, 135^\circ, 90^\circ, 20^\circ$ .

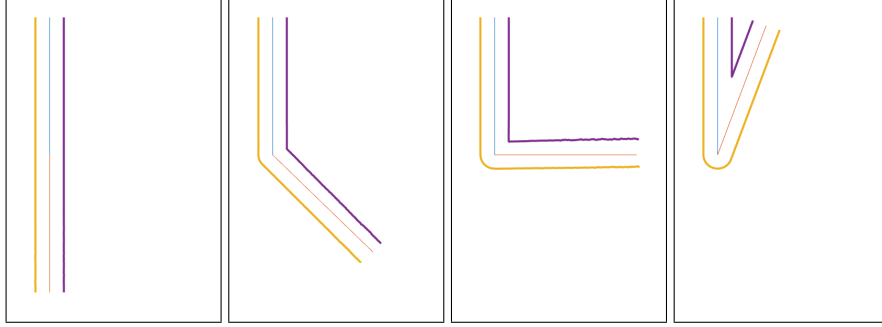


Figure 4. The results for PSD at  $\theta = 180^\circ, 135^\circ, 90^\circ, 20^\circ$ .

## 7. Computing joint weights from samples

We used least squares to calculate weights for each joint:

$$\tilde{v}_k - e_k = \sum_{j=1}^{n_{joint}} v_j w_j \quad (7)$$

where  $v_j = T_j v_0$ .

## 8. Results

The results of SSD are shown in Figure 3. As expected, the result at  $\theta = 180^\circ$  is perfect since that is the bind pose. However, for the other angles, it is very apparent that the skin collapses where the bones meet. At extreme angles like  $\theta = 20^\circ$ , the inner skin even collapses past the bone and folds into itself.

The results of PSD are shown in Figure 4. (A YouTube video showing the results of PSD for  $15^\circ \leq \theta \leq 190^\circ$  is also available at <https://youtu.be/v3lwH1p3HBU>). Even though the results are computed at angles that were not defined in the example poses (besides at  $\theta = 180^\circ$ ), the skin does not collapse, has very smooth transitions, and stays close along the bones.

The results of the computed joint weights are shown in Figure 5. The results do not look good because the least squares estimations can produce weights that are less than 0 or greater than 1, in which case the outcome will not converge.

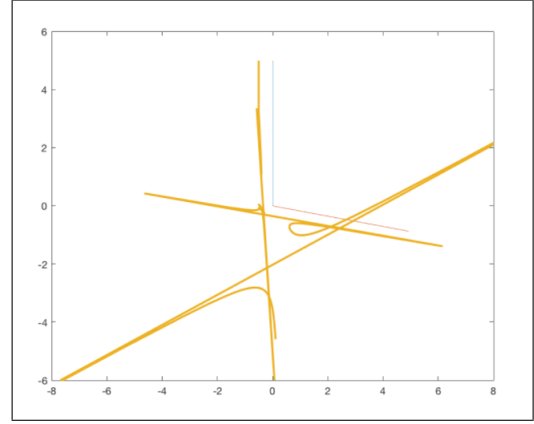


Figure 5. The results with computed joint weights using least squares estimation.

## 9. Conclusion

We were able to show the shortcomings of SSD at sharp angles and also demonstrate that, with appropriately defined example poses and appropriately chosen parameters for the radial basis function, PSD can fix those issues and produce smooth skinning.

Next steps include adding more degrees of freedom to our system and implementing WPSD.

## Acknowledgements

We would like to thank Professors Wojciech Matusik and Justin Solomon for their guidance throughout this project.

## References

- [1] Taehyun Rhee, J.P. Lewis, and Ulrich Neumann. Real-Time Weighted Pose-Space Deformation on the GPU (2006).
- [2] J.P. Lewis, Matt Cordner, and Nickson Fong. Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation (2000).
- [3] T. Kurihara, N. Miyata. Modeling Deformable Human Hands From Medical Images (2004).
- [4] Thalmann et al. Joint-dependent Local Deformations For Hand Animation and Object Grasping (1988).
- [5] X. C. Wang, C. Phillips. Multi-weight Enveloping: Least-Squares Approximation Techniques For Skin Animation (2002).
- [6] A. Mohr, L. Tokheim, M. Gleicher. Direct Manipulation of Interactive Character Skins (2003).