Physically Inspired Stretching for Skinning Animation of Non-rigid Bodies

Michal Piovarči* Martin Madaras† Roman Ďurikovič‡

Comenius University Bratislava, Slovakia

Abstract

We propose a physically inspired stretching model for non-rigid bodies with linear skeletons. Given an input model composed of linear skeleton segments, it extract scaling matrices that can be directly used in skinning animation. The stretching model evaluates stretching of the body cause by gravitational force and stretching of the body caused by muscle contraction. Our model is based on small deformation theory, which can be directly applied on cylindrical shapes. Since the input body may differ from a cylindrical shape, it is decomposed into several cylindrical parts and stretching factors are calculated for each part individually. Next, the body is stretched along the skeleton based on the function derived from the sum of skeleton curvature. Finally, a system for the visualization of a particle-based simulation using linear blend skinning is created and enhanced with out stretching model.

CR Categories: Computer Graphics [I.3.8]: Applications

Keywords: skinning animation, stretchable skinning, physically inspired stretching, simulation visualization

1 Introduction

When visualization of the particle-based simulation is needed, the simulation is performed and positions of all the particles can be visualized. This is good approach when the visualization is displayed locally - on the same machine as the simulation is performed. The situation changes, if we require to visualize the simulation remotely - on a client machine. Due the bandwidth limits, we might not be able to stream all the positions of particles during the simulation at every time step. In that case, it is possible to lighten the bandwidth using skinning animation for an approximation, instead of visualizing all the particles. In order to perform skinning animation, only the positions from one time step with skinning data have to be transferred and later only skeleton transformations have to be transferred at every time step during the animation.

However, classical skinning animations use matrices composed of rotations and translations only. The simulated body may also exhibits other changes in the shape, e.g. compression caused by gravitational force or muscle contractions. In our modification, stretching of the body caused by gravitational force and muscle contraction is performed. Since the amount of stretching is dependent on local change in volume and curvature around each skeleton segment, the change of the volume is approximated using Shape Diameter Function (SDF) [Shapira et al. 2008] and change in curvature is approximated by angle between neighboring skeleton segments. The body is approximated by cylinders and gravitational force is applied on the body using small deformation theory. Our stretching model needs non-rigid body with a linear skeleton as an input. The linear skeleton S is in this case a set of skeletal segments, where each skeletal segment s is defined as a vertex pair $(\mathbf{v}_i, \mathbf{v}_j)$, such that $\mathbf{v}_i \in \mathbb{R}^3, \mathbf{v}_j \in \mathbb{R}^3$. Next, the change of SDF of the body and curvature of the skeleton is calculated across all the data sets from the simulation. The measured data are then used for calculation of parameters used to fit our stretching model into the dataset.

A related work to skinning techniques and physically-based simulations is given in Section 2. Our elasticity inspired stretching model is described in Section 3. In Section 4, an example of application focused on visualization of physically-based particle simulation is given. Finally, Sections 5 and 6 give a view on the results and limitations of our model.

2 Related Work

Linear blend skinning and dual quaternion skinning [Magnenat-Thalmann et al. 2004; Kavan et al. 2008; Kavan and Sorkine 2012; Kim and Han 2014] are particularly widespread geometrical skinning techniques. Both techniques are capable of capturing the rotation of skeletal bones which makes them useful for reconstruction of rigid motion of a non-rigid body. The transformations of vertices at run time are a weighted sum of skeletal transformations. The required weight can be created manually by artist or they can be automatically generated [Jacobson et al. 2011]. However, the desired non-rigid motion is not captured as these techniques represent rotational and translational differences. Example based, physically inspired or physically based techniques were developed in order to capture the non-rigid transformations of soft bodies.

More complex geometrical methods try to reproduce physically based deformations with heuristics. In [Jacobson and Sorkine 2011], authors have developed a method that supports stretching and scaling of the mesh using skinning technique. The method supports stretching in the direction of each bone. Some methods [Yang and Zhang 2005b; Forstmann et al. 2007] try to reproduce non-rigid transformations using anatomy-based heuristics. All above mentioned methods produce transformations which depend on the animation skeleton and predefined constrains. Therefore, they are not suitable to replicate effects that are independent of skeletal transformations.

Example-based methods [Mohr and Gleicher 2003a; Wang et al. 2007; Shi et al. 2008; Wang et al. 2013] create the final animation from a set of input frames. The resulting skinning algorithms can deform models in real-time. Unfortunately, these methods are mainly based on the input poses. Therefore, non-rigid deformation not captured in the input set of frames cannot be reproduced and the stretching parameters cannot be changed during the animation.

Physically based simulations can credibly reproduce secondary motions and skin response to external stimuli based on anatomical data [Ng-Thow-Hing 2001; McAdams et al. 2011] or physical models [Kim and Pollard 2011; Hahn et al. 2012; Tan et al. 2012]. The

*e-mail: michal.piovarci@gmail.com
†e-mail: madaras@sccg.sk
‡e-mail: durikovic@fmph.uniba.sk

SCCG '15, April 22 – 24, 2015, Smolenice, Slovakia.
 ISBN 978-1-4503-3693-2/15/04…$15.00
DOI: http://dx.doi.org/10.1145/2788539.2788545

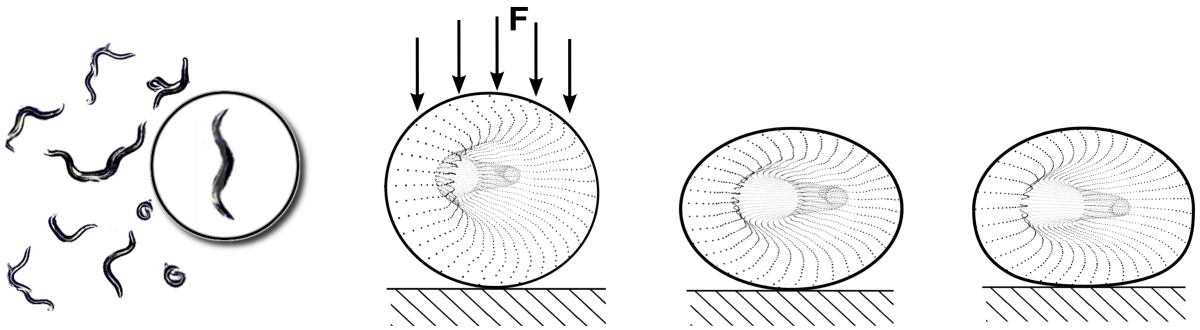


Figure 1: Non-rigid body gravitational compression. From left to right (a) non-rigid body is simulated; (b) rigid transformations are extracted and used to deform the body, the body is shown from inside; (c) due to gravitational force **F** the body is deformed using extracted rigid transformations as well as using our non-rigid stretching factors; (d) ground truth from physical simulation.

approaches are usually solved using finite elements method and are computationally too expensive to be performed in real-time. While the results of the animation could be performed offline, simulated models would require to transfer all vertex data for each time step of the simulation which might be impossible due to limited bandwidth.

Physically based methods try to reproduce the effects of physical simulations, while requiring only a fraction of time. For example [Kavan and Sorkine 2012; Vaillant et al. 2014] proposed methods witch try to emulate skin contact and muscle bulging which occurs at joints of vertebrates. Above mentioned techniques have in common that they react to the motion of the underlying skeletal structure. Therefore, external stimuli such as compression of soft body under gravitational compression are ignored.

3 Elasticity Inspired Stretching Model

The basic idea of our model is shown in Figure 1. During simulation of non-rigid bodies (see Figure 1(a)) we can extract the rigid movement as translations and rotations of an animation skeleton. However, when we apply this transformations the resulting mesh ignores non-rigid transformations caused by gravitational force **F** (see Figure 1(b)) and muscle stretching. After applying our stretching factors (see Figure 1(c)) we can achieve a result which is much closer to ground truth shown in Figure 1(d), without any additional computational power needed on the client-side.

Rigid transformations constructed from rotational differences in animation skeleton allow us to represent general movement of an animated non-rigid body. However, the body undergoes transformations which also affect its width, height, and length. Width and height of the body are primarily affected by gravitational pressure. The length of the body is mainly affected by the contraction of its muscles. We need to enhance skinning by transformations which would approximate these physical effects occurring during the simulation. The calculated transformation factors should satisfy the following criteria:

- The resulting skinning technique should perform in real-time.
- The factors influencing the skinning should be based on physical properties of the body.

Using our model we wish to replicate non-rigid material response to external forces, while maintaining the real-time performance of geometrically-based skinning approaches. Our options for base skinning techniques are limited to geometrically-based skinning due to the performance requirement of alternative approaches. Since we are not modifying the rigid motion of the body we have based our method on linear blend skinning. Linear blend skinning allows us to separate the rigid motion from the non-rigid transformations.

The transformation applied by linear blend skinning on a rigid body can be decomposed into translations and rotations. Assuming that $\mathbf{v}_i$ is position of vertex i at rest pose, $w_{i,j}$ is the influence of j-th bone on vertex i, m is the number of bones, and $\mathbf{T}_j$ and $\mathbf{R}_j$ are the translation and rotation translation vectors of j-th bone, new vertex position $\mathbf{v'}_i$ can be computed as:

$$\mathbf{v'}_i = \sum_{j=0}^{m} w_{i,j}(\mathbf{R}_j\mathbf{v}_i + \mathbf{T}_j). \quad (1)$$

We can extend this equation by adding an additional scale matrix $\mathbf{S}_j$ for each bone. This matrix will represent the elastic response of the body to external and internal stimuli and the updated equation will look as follows:

$$\mathbf{v'}_i = \sum_{j=0}^{m} w_{i,j}(\mathbf{R}_j\mathbf{S}_j\mathbf{v}_i + \mathbf{T}_j). \quad (2)$$

Note that skinning matrices affect associated vertices uniformly. Therefore, it is not possible to change the position of each vertex individually.

Gravitational force acting on the surface of non-rigid body produces an elastic transformation. The transformation stretches the surface of the body and depends on its internal structure. We would like to compute the stretching factor of the surface of the body caused by this elastic transformation. Solution of such a problem is formidable so we seek to simplify the problem in order to find an approximate solution. Applying small deformations theory and formulas developed for Young's modulus enables us to calculate the change in size of a material under compression. Therefore, we use the following assumption about the body under study to be able to use theory developed for small elastic deformations:

1. Material of the body is homogeneous and isotropic.
2. The body exhibits only small deformations, therefore we can calculate the deformations assuming linear elastic material response.

3. The body has cylindrical shape with length l equal to the length of the body and width w equal to the width of the body.

4. The body is supported from the top and stretched by its own weight.

These assumptions allow us to use Hooke's law and Young's modulus for compression calculations of a cylindrical body under uniform pressure. Using linear elasticity theory we can change the problem from gravitational compression to gravitational stretching. The calculated displacement and stretching factor will have the same value and differ only in sign. Solution to this problem is only a rough approximation of the originally desired result. However, taking into account the properties of animation matrices it is adequate for our purposes.

Now, we wish to reformulate our original problem. Determine the change in width Δw of a cylindrical body supported from the top and stretched by its own weight. A visual representation of the problem is shown in Figure 2. The desired stretching factor is a ratio of Δw of the cylindrical body to its original width w. Gravitational force $\mathbf{F}$ acting on the body has the same direction as negative y axis. The magnitude of the force is $||\mathbf{F}|| = ||m\mathbf{g}||$, where m is the mass of the body and $\mathbf{g} = (0, g, 0)$ is the gravitational acceleration. We can express the components of $\mathbf{F}$ as $F_y = -mg$, $F_x = F_z = 0$.

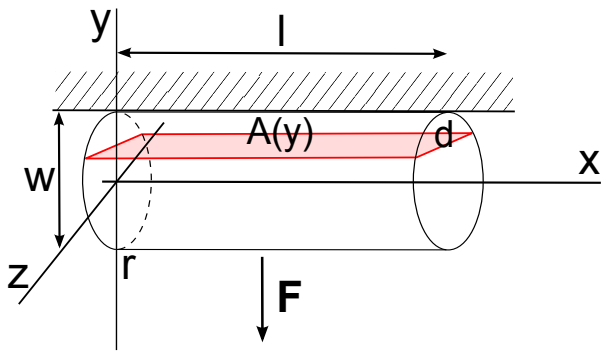


Figure 2: Non-rigid body simplified as a cylindrical body supported from the top and stretched by its own weight. The dimensions of the cylinder are derived from the length l and width w of the body. Gravitational force $\mathbf{F}$ acts on the body in the direction of negative y axis. One cross-section of area $A(y)$ is marked as a red rectangle. The width of the cross-section is given by d.

Using the definition of Young's modulus we can formulate the formula for computation of Δw.

$$\Delta w = \frac{F_y w}{A(y)E}, \tag{3}$$

where F_y is the force acting on the body, w is width of the cylinder, E is Young's modulus and $A(y)$ is the cross-sectional area through which the force is applied. Function $A(y)$ is changing along the body of the cylinder. In Figure 2 we can see a cross section of the cylindrical body. With increasing values of y the width of the cross section $A(y)$ is getting smaller. From the definition of a circle and each value of y we can calculate the width of the cross-section d as $d = 2\sqrt{r^2 - y^2}$, where $r = 0.5w$. Therefore, we can express the cross section area as $A(y) = 2l\sqrt{r^2 - y^2}$.

Since the values of Δw change along the width of the cylinder we can integrate them to calculate the desired value of Δw for the whole cylindrical body. The base of the cylindrical body is axially symmetric so we can calculate the integral for half of the body and multiply the result by 2. From the definition of Young's modulus and calculation of width for each cross-section we get:

$$\Delta w = 2\int_0^r \frac{F_y \mathrm{d}y}{EA(y)} = 2\int_0^r \frac{F_y \mathrm{d}y}{2El\sqrt{r^2 - y^2}}. \tag{4}$$

Which can be reduced to:

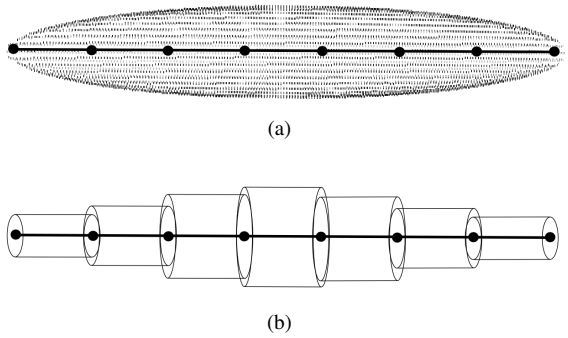
$$\Delta w = \frac{F_y}{El}\int_0^r \frac{\mathrm{d}y}{\sqrt{r^2 - y^2}} = \frac{F_y}{El}\left[\arcsin\left(\frac{y}{r}\right)\right]_0^r. \tag{5}$$

Finally, after evaluating the integral the equation reduces to:

$$\Delta w = \frac{F_y \pi}{El2} = \frac{mg\pi}{El2}. \tag{6}$$

The compression factor can then be simply calculated as the ratio of Δw and the original width of the body w. Having calculated the compression factor of the body under pressure we can use it in a scaling matrix to transform the whole non-rigid body. The body preserves its volume, therefore on one hand the compression factor is used to compress the body in direction of gravitation, on the other hand it is used to stretch the body in direction perpendicular to gravitation and its animation skeleton.

The calculated compression factor is adequate for the center of the body since the width of the body is similar to the width of the cylinder. However, the tips of the body can be considerably thinner. The calculated factor then causes much higher deformation than is desired. In order to correct for the differences in width of the body we calculate the compression factor for each skeletal segment. Computation of new body measurements for each segment is performed as follows. The length of each skeleton segment is measured. Young's modulus is constant for the material. Width of each segment is measured as average value of Shape Diameter Function [Shapira et al. 2008]. Finally, the weight of the cylinder is proportional to the volume of the segment. This way we get the compression factor for each bone. From this compression factor we can reconstruct the corresponding scaling matrix $\mathbf{S}_j$ of each bone. In Figure 3(a) we can see a non-rigid body with its animation skeleton consisting of 7 segments. The body is then approximated with cylinders result is shown in Figure 3(b).



(a)

(b)

Figure 3: Non-rigid body represented as cylinders. (a) a non-rigid body with animation skeleton consisting of 7 segments; (b) cylindrical approximation of the non-rigid body.

During the simulation, the muscles of the non-rigid body are contracting and twisting the body. When the muscles are contracted the body shortens. It is desired to replicate this effect as the body shortens significantly. The stretching factor from twisting is calculated directly from simulation data. Each time the muscles contract and

twist the body, its total curvature along the skeleton increases. Having calculated the rotation between segments of the skeletal structure we can estimate the curvature of the body κ as a sum of all the rotations applied on the skeletal structure:

$$\kappa = \sum \{\measuredangle \mathbf{v}_1\mathbf{v}_2\mathbf{v}_3 | (\mathbf{v}_1,\mathbf{v}_2) \in S, (\mathbf{v}_2,\mathbf{v}_3) \in S\}, \quad (7)$$

where S is a set of skeletal segments, each skeletal segment s is defined as a vertex pair $(\mathbf{v}_i,\mathbf{v}_j)$ such that $\mathbf{v}_i \in \mathbb{R}^3, \mathbf{v}_j \in \mathbb{R}^3$. We then investigate the length of the body in relation to the total curvature κ of its animation skeleton. A function Ψ representing the mapping from the total curvature κ to the length of the body is extracted. During animation we calculate the curvature of the animation skeleton and the length of the body using the pre-calculated function. Finally, the stretching factor is computed as the ratio of the expected length of the body to the original length of the body as:

$$StretchingFactor = \frac{\Psi(\kappa)}{l}. \quad (8)$$

4 Stretching Application

We have used our stretching model in a system for data transfer compression and visualization of particle simulation in OpenWorm project. The input for the system are positions of particles in different time steps during the whole simulation and membrane connectivity of the cuticle particles (see Figure 4). Skinning weights and indices for the cuticle mesh are computed from bind pose in zero time step (Section 4.2). A skeleton is extracted from the cuticle each time step of the simulation (Section 4.1). Transformation changes between skeletons extracted at each time step are computed. The transformations are then post-processed and enhanced by our elasticity inspired stretching model. In the end, the simulation is visualized using skinning animation.

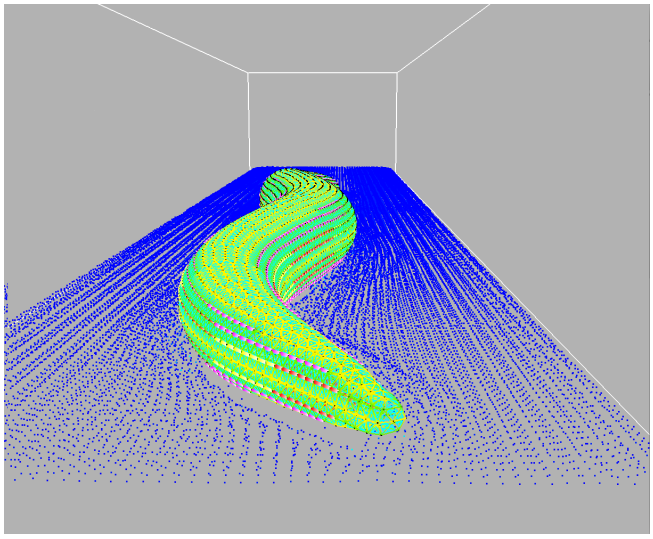


Figure 4: Particle simulation of C. elegans nematode in Sybernetic environment.

4.1 Skeleton Extraction

There are a lot of automatic skeleton extraction methods in the field. For example Au et al. [Au et al. 2008] or Aujay et al. [Aujay et al. 2007] can automatically extract a skeleton from a mesh with arbitrary topological branching. Methods as Cao et al. [Cao et al. 2010] or Sharf et al. [Sharf et al. 2007] can operate even on point clouds. However, all of these methods are computationally too expensive to be performed every time step of the simulation. The method by Shapira et al. [Shapira et al. 2008] extracts the skeleton by shifting model vertices by half of Shape Diameter Function (SDF) values in inward normal direction. Finally, curves are fitted into the shifted point clouds while minimizing least squares error and the final skeleton is constructed.

The cuticle of C. elegans has a very simple topology without branching. Therefore, we have used skeleton extraction algorithm based on SDF [Shapira et al. 2008]. More detailed description of our skeleton extraction algorithm is out of the scope of this paper. However, a different method for skeleton extraction could be used.

4.2 Skinning Data Extraction

Different approaches can be used for computation of skinning data. Existing approaches can be divided into two main groups - approaches computing skinning weights from one input pose and example-based approaches that fit skinning weights to more input poses. From one input mesh only, the skinning weights can be calculated from angles between bones and vertices [Yang and Zhang 2005a] or from Euclidian distances [Yang and Zhang 2006]. More precise results can be obtained from more input poses. In [Mohr and Gleicher 2003b; Kavan et al. 2010; Le and Deng 2012; James and Twigg 2005] authors used least squares optimization to compute skinning weights that minimize a predefined function. These methods are either based on finding rigid regions in the models or require extensive iterative optimization. Therefore, mentioned methods are not suitable for our problem.

Therefore, we use skinning weights based on inverse geodesic distance. We have chosen to use geodesic distance based skinning weights, because they work well even in the extreme case, when the worm is extremely twisted and there are no rigid segments at all. More detailed description of the skinning weights extraction algorithm is out of the scope of this paper.

4.3 Stretching of the Worm

After extracting the skinning data and rotational differences between animation skeletons the rigid motion of C. elegans nematode can be reconstructed. In order to credibly reconstruct the non-rigid elastic transformations we have implemented our model as part of the animation extraction pipeline. Skinning matrices are updated using calculated compression and stretching factors. Final physically inspired matrices are then send to the client-side in place of the original rigid matrices. Thanks to this approach the visualization can run in real-time on the client-side.

In OpenWorm project the simulation point cloud is divided into 30 segments corresponding to the animation bones. For each segment our model is used to calculate the compression factors. For example using Equation 6, physical measures for C. elegans nematode from Table 1, and gravitational acceleration g equal to $9.81 m/s$ we calculated that Δw of the cylindrical body is equal to 1.022×10^{-2} mm. This corresponds to increase of size by 12.3%.

The change in length is depending on the curvature is shown in Figure 5. The rate of curvature to the length can be linearly approximated with function $\Psi(\kappa) = -10.191\kappa + 306.03$, where $\psi(\kappa)$ is

Measurement	Abbreviation	Value
Length	l	$0.88mm$
Width	w	$0.085mm$
Mass	m	$2\mu g$
Young's modulus	E	$3770Pa$

Table 1: Measurements of C. elegans nematode from [McCulloch and Gems 2003] and [Sznitman et al. 2010].

the length of the cuticle and κ is total curvature. This function approximates measured data with correlation coefficient $R^2 = 0.9761$. Knowing the curvature of the worm the stretching factor is calculated using the ratio between worm's expected length and worm's length at rest pose.

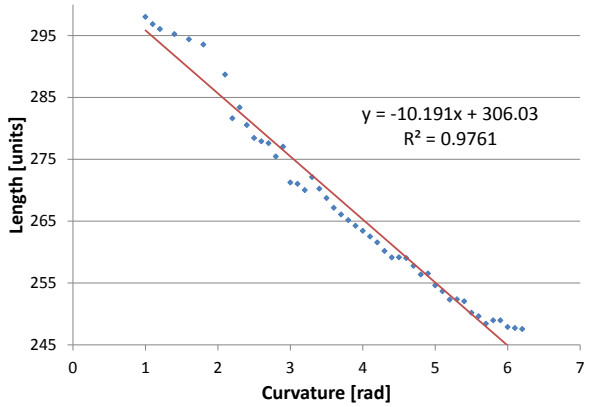


Figure 5: Chart showing worm's mean length for measured curvature of worm's animation skeleton.

5 Results

In this section we present results of our stretching model when applied on a concrete problem. The evaluation of the system concerns the data compression ratio achieved in the application and final skinning animation error evaluation with and without our stretching model.

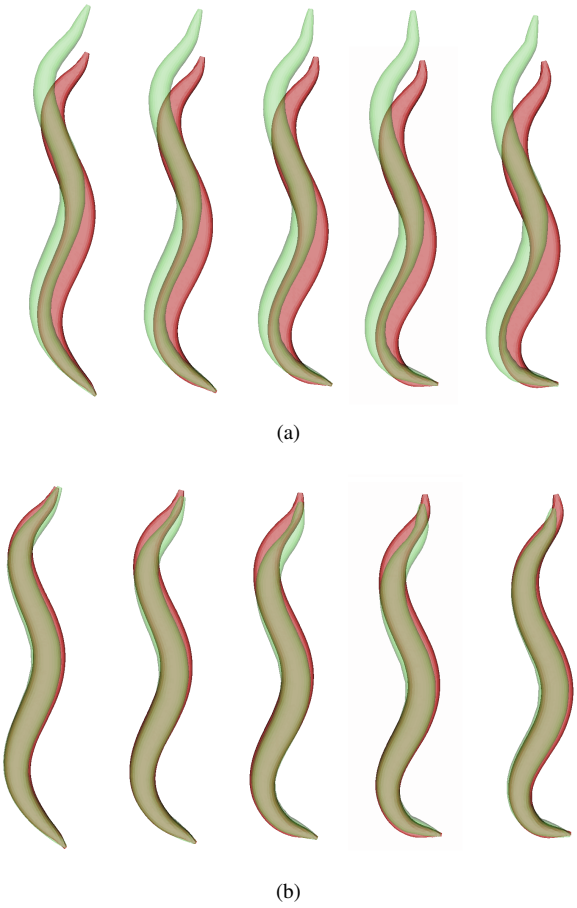
5.1 Skinning Error Evaluation

The quality of lossy compression algorithms is evaluated according to an error measure. Mathematical error measures such as mean square error (MSE) are usually used to evaluate the precision of the algorithm according to a global difference. Meanwhile, the perceptual error measures take human visual system into account and tolerate imperceptible errors. The data from the simulation of C. elegans are processed by our algorithm and sent directly to the end user for visualization. No further processing on the data is applied. Therefore, both kinds of error measure can be used and the choice of error measure depends on the data.

The original non compressed cuticle from the simulation is composed of point cloud and its triangulation. Our algorithm extracts data required for skinning animation of the worm's cuticle. Because of this, the error between original and skinned cuticle should be calculated based on the discrepancy of two animated meshes. One such error measure is STED [Váša and Skala 2011] which measures the spatial and temporal error of an animation sequence. When comparing the original simulation with our skinning simulation we measured STED error of 0.784, which is acceptable error for visualization of animation according to [Váša and Skala 2011].

While perceptual error measures are adequate to measure the similarity of the animation it is still interesting to measure global error measures such as MSE. Low MSE error values would indicate that our system produces models similar to the original simulation. MSE error also better evaluates the differences between classical skinning matrices and our physically inspired skinning matrices. The measured MSE was 24 and 10 for traditional and physically inspired matrices, respectively. The low values for physically inspired matrices indicates that further post processing effects could be implemented directly on the client which would lighten servers computation.

5.2 Skinning Animation Supporting Stretching



(a)

(b)

Figure 6: Results on skinning of the cuticle, red is the cuticle from the simulation, green is cuticle from the skinning. Sequence captures every 200 time steps of the simulation. Comparison of the simulation data with (a) standard rotation based linear blend skinning, (b) our modified skinning with stretching.

We present both results of skinning animation, classical approach and our physically inspired skinning. In classical approach, the bind pose is skinned by rotational matrices only, therefore the skinned

cuticle is longer and thinner than the cuticle from simulation. In our physically inspired skinning with stretching matrices, the cuticle is stretched and the resulting skinned cuticle has almost the same size as the simulated one. Comparison of both skinned meshes can be seen in Figure 6.

5.3 Compression of Data Transfer

The resulting data are transferred to the client where a GPU skinning algorithm is used to deform the cuticle of C. elegans at each time step of the simulation. Transfer of data for one frame of original simulation needs 68340 bytes (5695 x 3 x 4 bytes), one frame for normal skinning needs 496 bytes (31 x 4 x 4 bytes), therefore we have achieved compression ratio of 1:138.

6 Limitations

First limitation of our stretching algorithm is that it is only working on non-rigid bodies with linear animation skeletons. We can work around this limitation by looking at limbs of articulated characters separately. Each limb has a linear animation skeleton. Therefore, our algorithm can create corresponding cylindrical representation and calculate stretching factor for each limb separately.

Second limitation is that our algorithm was developed for invertebrates. Vertebrates, such as snakes or eels, have supporting skeletal structure in their ribcage which inhibits elastic transformations caused by gravitational force. We can work around this limitation by supplying individual values of elastic moduli for each animation skeleton segment, such that regions supported by ribs will exhibit only minimal compression due to gravitational loading when compared to regions not supported by ribs.

The last limitation of our algorithm is that it uses linear blend skinning. While this requirement was necessary in order to achieve real-time performance it is limiting for us, because scaling is applied uniformly to all vertices associated with a bone. Therefore, effects such as collision detection which could reproduce contact of the body with ground are not possible.

7 Conclusion

We have developed a physically inspired model which can reproduce non-rigid body transformations such as compression under gravitational force and stretching of the body caused by muscle contraction. The model can be used for various living organisms, e.g. worms, nematodes, etc. With minor modifications our model can be also used on vertebrates, e.g. snakes, eels, etc. and limbs of articulated models, e.g. humanoids. Our second contribution is the application of our model to a practical problem of point cloud simulation compression. The model is used as a part of our custom automatic pipeline for OpenWorm project which enables us to achieve high compression ratio while maintaining minimal perceivable error.

8 Acknowledgments

We would like to thank Libor Váša to share source code of STED for purpose of compression results evaluation. We would also like to thank Sergey Khayrulin for the simulation data from OpenWorm project. Our project was partly supported from grant No. UK/211/2015.

References

AU, O. K.-C., TAI, C.-L., CHU, H.-K., COHEN-OR, D., AND LEE, T.-Y. 2008. Skeleton extraction by mesh contraction. In *SIGGRAPH '08: ACM SIGGRAPH 2008 papers*, ACM, New York, NY, USA, 1–10.

AUJAY, G., HÉTROY, F., LAZARUS, F., AND DEPRAZ, C. 2007. Harmonic skeleton for realistic character animation. In *SCA '07: Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 151–160.

BRENTZEN, J. A., MISZTAL, M. K., AND WELNICKA, K. 2012. Converting skeletal structures to quad dominant meshes. *Computers & Graphics 36*, 5, 555–561.

CAO, J., TAGLIASACCHI, A., OLSON, M., ZHANG, H., AND SU, Z. 2010. Point cloud skeletons via Laplacian based contraction. In *Proceedings of the 2010 SMI Conf.*, 187–197.

CORMEN, T. H., LEISERSON, C. E., RIVEST, R. L., AND STEIN, C. 1990. *Introduction to Algorithms*, first ed. MIT Press and McGraw-Hill, 558–565.

FORSTMANN, S., OHYA, J., KROHN-GRIMBERGHE, A., AND MCDOUGALL, R. 2007. Deformation styles for spline-based skeletal animation. In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SCA '07, 141–150.

HAHN, F., MARTIN, S., THOMASZEWSKI, B., SUMNER, R., COROS, S., AND GROSS, M. 2012. Rig-space physics. *ACM Trans. Graph. 31*, 4 (July), 72:1–72:8.

JACOBSON, A., AND SORKINE, O. 2011. Stretchable and twistable bones for skeletal shape deformation. *ACM Transactions on Graphics (proceedings of ACM SIGGRAPH ASIA) 30*, 6, 165:1–165:8.

JACOBSON, A., BARAN, I., POPOVIĆ, J., AND SORKINE, O. 2011. Bounded biharmonic weights for real-time deformation. *ACM Trans. Graph. 30*, 4 (July), 78:1–78:8.

JAMES, D. L., AND TWIGG, C. D. 2005. Skinning mesh animations. *ACM Trans. Graph. 24*, 3 (July), 399–407.

KAVAN, L., AND SORKINE, O. 2012. Elasticity-inspired deformers for character articulation. *ACM Transactions on Graphics (proceedings of ACM SIGGRAPH ASIA) 31*, 6, 196:1–196:8.

KAVAN, L., COLLINS, S., ŽÁRA, J., AND O'SULLIVAN, C. 2008. Geometric skinning with approximate dual quaternion blending. *ACM Trans. Graph. 27*, 4 (Nov.), 105:1–105:23.

KAVAN, L., SLOAN, P.-P., AND O'SULLIVAN, C. 2010. Fast and efficient skinning of animated meshes. *Computer Graphics Forum 29*, 2, 327–336.

KIM, Y., AND HAN, J. 2014. Bulging-free dual quaternion skinning. *Computer Animation and Virtual Worlds 25*, 3-4, 321–329.

KIM, J., AND POLLARD, N. S. 2011. Fast simulation of skeleton-driven deformable body characters. *ACM Trans. Graph. 30*, 5 (Oct.), 121:1–121:19.

LE, B. H., AND DENG, Z. 2012. Smooth skinning decomposition with rigid bones. *ACM Trans. Graph. 31*, 6 (Nov.), 199:1–199:10.

MAGNENAT-THALMANN, N., CORDIER, F., SEO, H., AND PAPAGIANAKIS, G. 2004. Modeling of bodies and clothes for virtual environments. In *Cyberworlds, 2004 International Conference on*, 201–208.

MCADAMS, A., ZHU, Y., SELLE, A., EMPEY, M., TAMSTORF, R., TERAN, J., AND SIFAKIS, E. 2011. Efficient elasticity for character skinning with contact and collisions. *ACM Trans. Graph. 30*, 4 (July), 37:1–37:12.

MCCULLOCH, D., AND GEMS, D. 2003. Body size, insulin/igf signaling and aging in the nematode caenorhabditis elegans. *Experimental Gerontology 38*, 12, 129 – 136. Proceedings of the 6th International Symposium on the Neurobiology and Neuroendocrinology of Aging.

MOHR, A., AND GLEICHER, M. 2003. Building efficient, accurate character skins from examples. *ACM Trans. Graph. 22*, 3 (July), 562–568.

MOHR, A., AND GLEICHER, M. 2003. Building efficient, accurate character skins from examples. *ACM Trans. Graph. 22*, 3 (July), 562–568.

NG-THOW-HING, V. 2001. *Anatomically-based Models for Physical and Geometric Reconstruction of Humans and Other Animals*. PhD thesis, Toronto, Ont., Canada, Canada. AAINQ58941.

SHAPIRA, L., SHAMIR, A., AND COHEN-OR, D. 2008. Consistent mesh partitioning and skeletonisation using the shape diameter function. *Vis. Comput. 24* (March), 249–259.

SHARF, A., LEWINER, T., SHAMIR, A., AND KOBBELT, L. 2007. On-the-fly curve-skeleton computation for 3D shapes. *Computer Graphics Forum, (Proceedings Eurographics 2007) 26*, 3, 323–328.

SHI, X., ZHOU, K., TONG, Y., DESBRUN, M., BAO, H., AND GUO, B. 2008. Example-based dynamic skinning in real time. *ACM Trans. Graph. 27*, 3 (Aug.), 29:1–29:8.

Sibernetic. "http://sibernetic.org/".

SZIGETI, B., GLEESON, P., VELLA, M., KHAYRULIN, S., PALYANOV, A., HOKANSON, J., CURRIE, M., CANTARELLI, M., IDILI, G., AND LARSON, S. 2014. Openworm: an open-science approach to modelling caenorhabditis elegans. *Frontiers in Computational Neuroscience 8*, 137.

SZNITMAN, J., PUROHIT, P. K., KRAJACIC, P., LAMITINA, T., AND ARRATIA, P. 2010. Material properties of caenorhabditis elegans swimming at low reynolds number. *Biophysical Journal 98*, 4, 617 – 626.

TAN, J., TURK, G., AND LIU, C. K. 2012. Soft body locomotion. *ACM Trans. Graph. 31*, 4 (July), 26:1–26:11.

Three js. `http://threejs.org/`.

VAILLANT, R., GUENNEBAUD, G., BARTHE, L., WYVILL, B., AND CANI, M.-P. 2014. Robust iso-surface tracking for interactive character skinning. *ACM Trans. Graph. 33*, 6 (Nov.), 189:1–189:11.

VÁŠA, L., AND BRUNNETT, G. 2014. Rate-distortion optimized compression of motion capture data. *Computer Graphics Forum 33*, 2, 283–292.

VÁŠA, L., AND SKALA, V. 2011. A perception correlated comparison method for dynamic meshes. *IEEE Trans. Vis. Comput. Graph.*, 220–230.

WANG, R. Y., PULLI, K., AND POPOVIĆ, J. 2007. Real-time enveloping with rotational regression. *ACM Trans. Graph. 26*, 3 (July).

WANG, X., YANG, W., PENG, H., AND WANG, G. 2013. Shape-aware skeletal deformation for 2d characters. *Vis. Comput. 29*, 6-8 (June), 545–553.

YANG, X. S., AND ZHANG, J. J. 2005. Realistic skeleton driven skin deformation. In *In ICCSA (3*, John Wiley & Sons, Ltd, 1109–1118.

YANG, X., AND ZHANG, J. 2005. Realistic skeleton driven skin deformation. In *Computational Science and Its Applications ICCSA 2005*, O. Gervasi, M. Gavrilova, V. Kumar, A. Lagan, H. Lee, Y. Mun, D. Taniar, and C. Tan, Eds., vol. 3482 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1109–1118.

YANG, X., AND ZHANG, J. J. 2006. Stretch it - realistic smooth skinning. *2013 10th International Conference Computer Graphics, Imaging and Visualization 0*, 323–328.