Understandings about neuromechanical simulations and anime interpolations

Haina, Wang

April 13, 2023

Contents

Meanings of our work

Animal behaviors are intrigued by interactions between neural network dynamics, musculoskeletal properties and the physical environment.

Accessing and understanding them requires integrative and morphologically realistic neuromechanical simulations.

Illustration

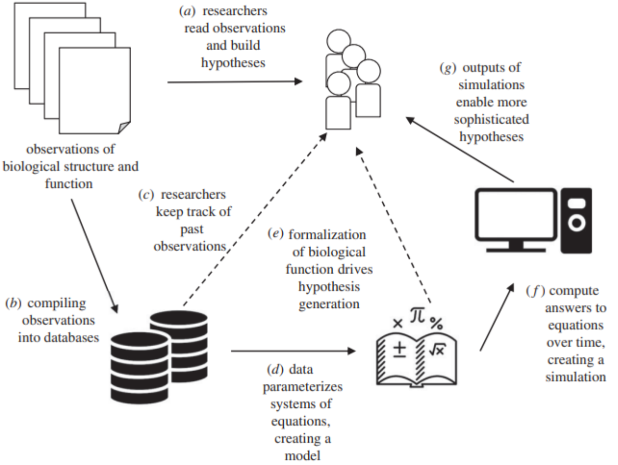


Figure 1: A schematic view of scientific value of modelling and simulation.

Observation, database, equation systems, simulation, every step relates to the researchers. They depict the development of the modeling in biology.

Our task stands at the last step: simulation. Within our help, the researchers may get a much more clear view when coming up with hypotheses.

A physically-based simulation of Caenorhabditis elegans

The force involved in the movement of C.elegans include Muscle Activation, internal forces and environment forces.

The internal forces are mainly pressure generated by internal liquids, which is simulated by a force shield; and the Environmental forces are simulated with a friction model abide by Signorini's law and Coulomb's friction law; And the key part is the muscle activations, which requires proper modeling of the worm.

The paper models the C. elegans as biphasic springs, sets their spring constant as a periodic function.

This way, the spring force may not just make the spring tend to stabilize, but to contract and relax as muscles, thus the worm may go forward.

Core formulas

Standard linear spring has the following spring force and the damper force:

$\vec{f_s} = -k_s(\vec{L_c} - \vec{L_r})(\frac{\vec{p_2} - \vec{p_1}}{||\vec{p_2} - \vec{p_1}||})$ $\vec{f_d} = -k_d(\vec{v_2} - \vec{v_1})(\frac{\vec{p_2} - \vec{p_1}}{||\vec{p_2} - \vec{p_1}||})(\frac{\vec{p_2} - \vec{p_1}}{||\vec{p_2} - \vec{p_1}||})$

A parameter θ can be added to the formula of spring force in order to turn the spring to a biphasic one : $\vec{f_m} = \theta k_s(\vec{L_c} - \vec{L_r})(\frac{\vec{p_2} - \vec{p_1}}{||\vec{p_2} - \vec{p_1}||})$

The problem is to set values of θ correctly real-time.

Core formula(Cont.)

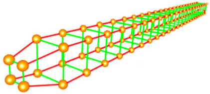


Figure 2: Structure of virtual C.elegans in essay

C.elegan is a kind of Annelida, so its body can be structured as connected 24 hexahedras, all with edges of biphasic strings. Currently, the strings have the following θ:

$$\theta = Csin(\frac{2\pi t}{T} - \frac{3\pi i}{N} - \frac{\pi}{2})$$

Where C is constant, i is the index of the ring, T is the period of activation and $N = 24$ is the total number of muscles.

Parameters & Details

The spring structure is not indentical to the muscle system of the worm, values for spring constants cannot simply taken from biological data, and the author get k_s, k_d, θ and μ by experience.

I'm also unable to tell why our simulation may result in vibrations of the worm, But force analysis during debugging may be the key way to fix.

Thoughts

The difficulties of the paper's reproduction are as follows:

1) The author's saying " muscles in the left must relax when the muscles in the right" is ambiguous, and there are other parameters(e.g. weight of the C.elegan)which should be set but not mentioned.

2) Force analysis inside the spring is tough.

Introduction

I have set up our project "Machine-learning based hand-drawn anime inbetweening" since 2022.4.

figure 3: hand-drawn anime sample.


"JUJUTSU KAISEN" EPISODE 13 BY STUDIO MAPPA
0:13
-0:01
1x

Main challenges

Technical challenges:

1) No color information

2) Non-linear, exaggerated and non-trivial motions

3) Limited state-of-the-art

Other challenges

Lack of datasets: Trade secrets.

Technical roadmap

Our current method:

1) Use other's style-painting model and some common DIP method to transform colored images into sketched images.

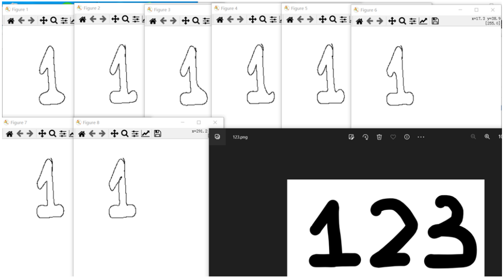
Technical roadmapCont.

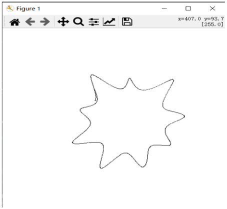
2) Use *cv2.drawcontours*() to fetch piecewise contours(about 4000-5000 per image) for every sketched images.

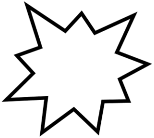
3) Use Newton-Raphson iteration method to convert all the contours with length over 40px into Bsplines.

4) Set equation of motion for Bsplines(long contours) and dot-sets(short contours) and noises separately.

sample







Model the contours

We are checking if the motion of the contour can be perform perfectly by the motions of the control points.

Let the track of the contours' control points be be $\{x(t), y(t)\}$, and make $\{x(t), y(t)\}$ be polynomials related to t. Then we can describe most kinds of motions easily only with order < 3.

Training requires the result has the least distance with mid-frame. By fixing some coefficients, uniform motions can be fit.

Details

The function is not necessarily a convex function, which will lead to divergence if the initial value of the Newton iteration method is not well selected. I'm trying to divide the contour into convexes and select the turning point as initial value.

How would I contribute :Experience

Drawing anime inbetweening frame has many difference with simulating mysterious creatures.

However, using physics, I may assume the motion of the creatures in the animation. And for some special scene such as the characters are fixed, modeling do work.

How would I contribute :Experience

Besides, though topic different, the process of exploring methods is similar

Our method has walked into dead end several times.

We've tried a motion-detection model but end up finding it can only tell what kind of motion it is with a poor accuracy;

We've tried to just treat the image as huge point clouds and deal it with Transformers, but traditional score-running tasks are too different from ours.

Only when we realize we can simplify the contours into Bsplines instead of points do we get progress. The kind of experience coincides with the development of OpenWorm.

How would I contribute: technical stacks

Languages:

C/C++: ACM/ICPC experience

python: SRTP & quantitive investment internship

Tools:

maya, houdini