Machon Sven

How can you generate the feather coat of a bird using the weights of the skeleton with python?

Graduation work 2021-2022

Digital Arts and Entertainment

Howest.be

CONTENTS

ABSTRACT

An abstract explains the aim of the paper in very brief, (the methods, results, etc.). Maximum length 150 words

INTRODUCTION

In the introduction, you write the background of your topic, explain the purpose of the paper more broadly, and explain the hypothesis, and the research question(s).

RELATED WORK

Computer generated feather coats are a well known and explored challenge in computer graphics. Several approaches have been explored to achieve a satisfactory result. Such as using field lines [1], [2], Bezier Curves [3], constraint surfaces [4] or using the weights of the skeleton[5], [6]. In this chapter we will discuss some of the techniques that have been used to generate feather coats.

1. ANIMATED FEATHER COATS USING FIELD LINES

By creating a vector field and using the field lines within you can generate a feather coat with minimal key orientations. Using the surface orientation vectors of each vertex and the vertex normal, a vector field is created surrounding the base object. Field lines are then created through following the vector field from each vertex. The vector determined at each point along the field line will then be used to estimate the next point.
To make up for the two-dimensional nature of a feather, a second field needs to be created perpendicular to the first. Using both fields you can build a coordinate system with the axes corresponding to feather thickness, length and width.
The use of vectors allows for more control over the feathers. The feathers of a bird are not the same scale everywhere, often the feathers around the head are smaller and more densely packed. By using key vectors that take length into consideration, you can create a feather coat with variation in feather scale.

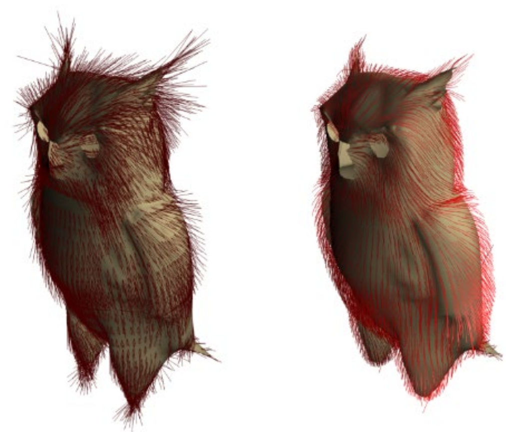

Figure 1: surface orientation field and field lines

After creating the vector field, feathers are placed at every vertex of the base object by duplicating a base feather mesh. All feathers will then be transformed from the root up, following the field lines. [2]

2. ANIMATED FEATHER COATS USING IMPLICIT CONSTRAINT SURFACES

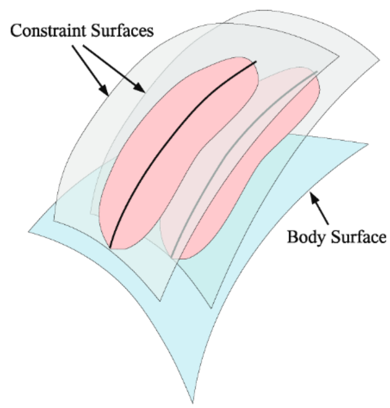

Figure 2: Feather in a constraint surface

A. Weber and G. Gornowicz [4] presented a way to generate feather coats using implicit constraint surfaces. Before all else, dozens of guide feathers are placed on key locations along the object surface. Next all other feather roots are generated. At these roots the feathers will be instanced by interpolating between the different guides.

The primary goal of this technique is to prevent interpenetration between the feathers. To achieve this an implicit constraint surface is defined for each feather that does not intersect with any other constraint surface.
This constraint surface must intersect the bird surface at the feather root position, while not intersecting with all other constraint surfaces.

Every feather has to have root position x_r on the surface of the bird S, x is a point on S and n_x is the surface normal at point x. C is a continuous constraint surface. M(∅) is a monotonically increasing scalar mapping function of scalar potential ∅, from hereon referred to as the loft function.

$$C(x) = x + M\big(\emptyset(\mathrm{x}) - \emptyset(x_r)\big)n_x$$

To ensure that the feathers are places at their root positions a few conditions must be met. The constraint surface must intersect the bird at the root position and the loft function must map zero to zero.

$$C(x_r) = x_r$$

$$M(0) = 0$$

All root positions should have a unique scalar potential value, this can be done by slightly adjusting values of any root positions with similar potential values.
If two constraint surfaces C_1 and C_2 with roots x_{r1} and x_{r2} intersect, they must do so at the same x

$$C_1(x) = x + M\big(\emptyset(x) - \emptyset(x_{r1})\big)n_x$$

$$C_2(x) = x + M\big(\emptyset(x) - \emptyset(x_{r2})\big)n_x$$

Therefor

$$C_1(x) = \ C_2(x)$$

$$M\big(\emptyset(x) - \ \emptyset(x_{r1})\big) = M(\emptyset(x) - \emptyset x_{r2}))$$

Since all root positions have unique scalar values $\emptyset(x_{r1}) \neq \emptyset(x_{r2})$ and constraint surfaces with unique scalar potentials will never intersect.

Once the constraint surfaces have been defined we can begin constructing the individual feathers. A feather will be constructed along the associated constraint surface, but there are a few characteristics of the feather that you would want to preserve, to ensure the fidelity of the result. These characteristics are the direction of the shaft, length of the shaft, angle of the barbs off the shaft and the length of the barbs.
You begin with the shaft direction and then go along the constraint surface for the length of the shaft, before doing the same with the barbs.

3. ANIMATED FEATHER COATS USING THE WEIGHTS OF A SKELETON

CASE STUDY

1. INTRODUCTION

2. MODELLING

2.1. BLOCKOUT

2.2. ZBRUSH

3. TEXTURING

4. SHADING

5. LIGHTING

EXPERIMENTS & RESULTS

REPEAT THE MAIN TOPICS, DISCUSS YOUR MAIN FINDINGS, DISCUSS THE END RESULT.

DISCUSSION

REPEAT THE MAIN TOPICS, DISCUSS YOUR MAIN FINDINGS, DISCUSS THE END RESULT.

CONCLUSION & FUTURE WORK

REPEAT THE MAIN TOPICS, DISCUSS YOUR MAIN FINDINGS, DISCUSS THE END RESULT.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] E. Zhang, K. Mischaikow, and G. Turk, “Vector Field Design on Surfaces.”

[2] Hannah. Slay, Shaun. Bangay, S. N. Spencer, SIGGRAPH., and European Association for Computer Graphics., *Animated feather coats using field lines*. Association for Computing Machinery, 2007.

[3] L. Streit and W. Heidrich, “Generating Feather Coats Using Bezier Curves.”

[4] A. J. Weber and G. Gornowicz, “Collision-free construction of animated feathers using implicit constraint surfaces,” *ACM Transactions on Graphics*, vol. 28, no. 2, Apr. 2009, doi: 10.1145/1516522.1516523.

[5] L. Liu, X. Liu, B. Sheng, Y. Chen, and E. Wu, “Incremental collision-free feathering for animated surfaces,” *Visual Computer*, vol. 33, no. 6–8, pp. 883–890, Jun. 2017, doi: 10.1007/s00371-017-1369-6.

[6] Liu L., Li X., Chen Y., Liu X., Zhang J., and Wu E., “efficient feathering system with collision control,” 2015.

APPENDICES