DIRECT TEXTURE SYNTHESIS OF FEATHER PIGMENTATION PATTERNS

C. G. Franco and M. Walter
Masters in Applied Computing, Unisinos
Centro de Informatica, Universidade Federal de Pernambuco, Brazil

Keywords: Natural Phenomena, Texture Synthesis, Feather Pattern Modeling, Feather Rendering.

Abstract: Feathers present exquisite shapes and visual patterns. Just recently feathers have been addressed in computer graphics with realistic-looking results. Current solutions, however, make use of texture mapping for adding the visual patterns seen on feathers. In this paper we present an integrated model for direct texture synthesis of feather pigmentation patterns. The model uses Bézier curves to model the feather shape and procedural MCLONE patterns for the synthesis of feather pigmentation patterns. We show that the Clonal Mosaic model (MCLONE) is a viable option for direct texture synthesis of the many pigmentation feather patterns found in Nature.

1 INTRODUCTION

The modelling of natural phenomena is one of the most challenging fields in computer graphics, due to the intrinsic complexity of natural phenomena and the familiarity we have with them. In the Animal Kingdom, feather formation is one of the most fascinating and intriguing process, which has attracted little attention in computer graphics and it is still a not clearly understood process in Biology (Prum and Williamson, 2002). In addition, feather pigmentation patterns exhibit a large range of visual diversity.

In the work presented so far on this subject, this visual diversity is achieved through texture mapping real world images. Although powerful, this method lacks flexibility and we are restricted to real images. Not to mention the usual problems of texture mapping such as distortion and the mapping itself. The main investigation problem in this work are visual patterns of feathers. We propose a procedural model based on the MCLONE model (Walter et al., 1998) which is able to synthesize feather pigmentation patterns directly on the feather shape, avoiding the difficulties associated with texture mapping. Besides, MCLONE patterns capture the large range of visual diversity found in feathers, as illustrated in Figure 1. We initially present related work on the topics explored in this paper, followed by a background section on the biology of feathers and the MCLONE model. Our solution is presented in Section 4. Finally, we present a few results with our approach and the conclusions.

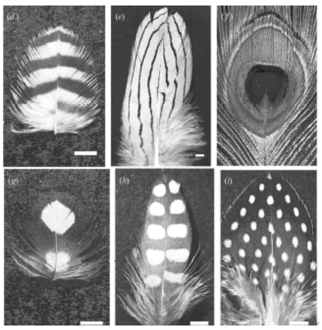


Figure 1: Example of pattern and shape diversity of real feathers. From (Prum and Williamson, 2002).

2 PREVIOUS WORK

As our main goal is the direct texture synthesis of feather pigmentation patterns, we present related work on direct texture synthesis and on feather modelling.

G. Franco C. and Walter M. (2007).
DIRECT TEXTURE SYNTHESIS OF FEATHER PIGMENTATION PATTERNS.
In *Proceedings of the Second International Conference on Computer Graphics Theory and Applications - GM/R*, pages 277-283
DOI: 10.5220/0002076602770283

2.1 Procedural Methods

In 1991 Turk (Turk, 1991) generated animal skin patterns with Reaction-Diffusion (RD) mechanisms, introduced by Turing (Turing, 1952). Instead of mapping the generated pattern onto a polyhedral or parametric model, Turk's approach simulated the RD system on the surface of the model, without the intermediate mapping from texture space to object space. Fleischer *et al.* also presented an approach (Fleischer et al., 1995) for texturing with direct simulation on the surface of an object. The surface of the object is covered with cells that are constrained to remain on an iso-surface computed from the original model. The whole approach is general, and can in principle generate many interesting organic-like textures, including RD ones. Their results show an organic quality to the generated textures, but they did not present any results simulating real-life patterns.

2.2 Direct Texture Synthesis From Samples

The following four approaches are variations on the basic idea of texture synthesis from samples on arbitrary surfaces, and therefore extensions of (Wei and Levoy, 2000).

Turk presented an approach where the texture synthesis on the surface is achieved through a hierarchy of points on the surface (Turk, 2001). A user-defined vector field indicates the orientation of the texture. The mesh vertices are then sorted in such a way that visiting the points in order will follow the vector field and will sweep across the surface from one end to the other. The color of a particular point is defined by examining the color of neighboring points and finding the best match to a similar neighborhood in the given texture sample. Wei and Levoy (Wei and Levoy, 2001) also presented an approach based on searching strategy to solve the texture synthesis problem on arbitrary surfaces. The difference between this approach and Turk's one is that in this work there is no need to specify a vector field, since it is obtained on the fly.

In 2002, Soler *et al.* (Soler et al., 2002) presented an approach that, instead of searching the whole surface point by point, it progressively covers the texture surface with texture patches, of several sizes, selected from a single input image. One of the advantages of this approach is that there is no need of generation of a intermediary geometry, and the initial geometry is preserved. In another work in 2002, Tong *et al.* (Tong et al., 2002) presented a solution that enables the synthesis of BTFs (Bidirectional Texture Functions) (Dana et al., 1999), on arbitrary surfaces. This solution performs the synthesis of the BTF's samples directly on the surface, avoiding distortions and discontinuities. The results show that this approach achieved the main goal of maintaining details of the mesostructure, represented by the BTF, in all viewing and lighting directions.

2.3 Feather Modelling

The first published model targeted specifically for feathers was presented in (Dai et al., 1995). The feathers were modelled as line segments branching from a main structure. The textures were computed from simulations of dynamical systems.

The work in (Chen et al., 2002) uses parametric L-systems for the modelling of the feathers and texture mapping and customized BTF for the rendering. In (Streit and Heidrich, 2002), the feathers are modeled as a collection of Bézier curves. The overall shape of the individual feathers is achieved by the user specifying key barbs from which the other barbs are derived by interpolation. The rendering uses texture mapping to add color to the barbs. Recently in Biology (Prum and Williamson, 2002) proposed an approach addressing pattern formation for feathers. They used Reaction-Diffusion (RD) mechanisms to build the pattern of the feathers. The results are interesting, but since their work is not focused in computer graphics, the visual aspects were not privileged. Furthermore, there is no universal RD system capable of generating all patterns, and therefore distinct RD systems are used or even sometimes a new one must be developed.

Our work advances the idea of direct texture synthesis, using a procedural model targeted to visual patterns of feathers. In this sense we make contributions on procedural texture generation, feather modeling, and direct texture synthesis.

3 BACKGROUND

Since our model derives its inspiration from real feathers, we briefly review in this section basic information on the biology of feathers.

Feathers are a type of branching structure, flexible and yet strong (Freethy, 1982). They present a main rigid structure called *calamus* at the base (with no branching structures) and the *rachis* where the main body of the feather develops (see Figure 2). From the rachis a variable number of *barbs* are originated. The collection of barbs at each side of the feather's body is known as *vanes*. Each barb is built from two sets

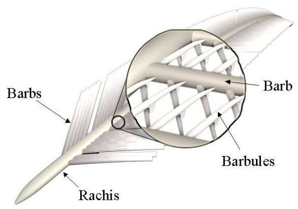


Figure 2: Structure of a typical feather.

of interconnected *barbules*: the anterior and the posterior barbules. For some feathers each barbule has in turn microscopic *barbicles* (not labeled in the figure), structures with small hooks that connect the anterior barbule of one barb to the posterior barbule of the next barb. This connection helps maintaining the feather overall shape.

There are 5 types of feathers. The most familiar type is the *contour* feather. The semiplume has a structure between the contour and the plume. The plume type is soft and the length of the barbs is typically longer than the length of the rachis itself. Finally, the filoplume and the bristles are very small specialized feathers.

Most recent theories (Prum, 1999) present the feather as a tubular structure that grows from a folliculus. Barbs grow around this folliculus, and as they approach the anterior part of that, they are joined to form the rachis, creating the branching structure (Prum, 2001). Melanin, one of the key elements for producing the colors, come from melanocytes, that migrate through this structure during feather development. Currently, there is no universal theory on how a specific pigmentation pattern is defined. An accepted concept today is that within-feather pigment patterns are determined by differential pigmentation of keratinocytes within independent barb ridges during feather development (Prum and Williamson, 2002).

3.1 The Clonal Mosaic Model

The Clonal Mosaic theory (Walter et al., 1998), proposes that the typical yellow-black stripped and spotted patterns occurring in several species of mammals, reflect a spatial arrangement – a mosaic – of epithelial cells which derive from a single progenitor, i.e., they are clones. Hence the name Clonal Mosaic (MCLONE). The basic idea of the Clonal Mosaic model is that the pattern on mammals is obtained from simulation of the interaction between cells with different properties, such as color, division rate, adhesion, and others (Walter et al., 1998). One of the main advantages of the MCLONE model is its generality, since in the last years it has been explored in different domains, ranging from pattern modelling in butterflies (Lied and Walter, 2002) up to simulation of growth of tumors (dos Reis et al., 2003). Although the MCLONE has been initially designed for simulation of mammalian patterns, it is general enough to handle also visual patterns of feathers.

4 THE PROPOSED MODEL

The approach introduced here is an extension of (Franco and Walter, 2002). For completeness, we start this section with a brief review of that work.

In order to model a single feather, the user initially defines a cubic Bézier, which represents the rachis, and two Bézier curves that define the boundaries of the overall feather structure (the vanes). From the rachis the technique generates a variable number of barbs, controlled by a set of parameters. Each barb is itself a Bézier segment with 4 control points. Even though this scheme uses cubic segments, the resulting Bézier in practice has enough flexibility to represent a wide range of possible barb shapes. The model has 7 main parameters which allow real time generation of many feather structures, from contour feathers to semiplumes, plumes, and even filoplumes. To get a detailed description of the meaning and playing role of each parameter we suggest the reader to consult (Franco and Walter, 2002).

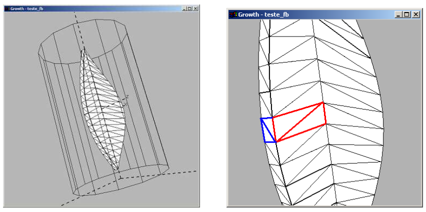
4.1 Modelling the Pattern

This step consists in synthesizing the texture representing a determined feather pattern, using the MCLONE. At this point, we have the feather represented only by its barbs, that are a collection of Bézier curves. In order to simulate the MCLONE model we used a process analogous to the one described in (Walter et al., 2001). This process has two main steps described next.

4.1.1 Generation of the Mesh Representing the Feather Structure

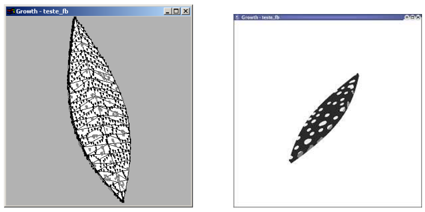
The MCLONE model is applied directly onto the object's surface. Since we only have curves in our model, in this step we derive a triangular mesh from the original structure of the feather modelled by the user, as we can see in Figure 3.

This derivation process is quite simple. The surface represented by the rachis and the vanes forms the



(a) Generated Mesh (b) Algorithm's detail

Figure 3: Visualization of the mesh generation algorithm.



(a) Cells (b) Final Pattern

Figure 4: Graphical representation of cells.

superior part of the volume. Then, this surface is triangulated, using a simple interpolation mechanism, with a level of detail defined by the user. As we can see in Figure 3, the rachis and the vanes are divided into n sections, where n is the user-defined level of detail. From there we build a quadrilateral covering this section, and this is divided into two triangles (marked red in the figure). In order to build the posterior part of the feather, we project the faces using the inverted normal vectors as guidelines. The user specifies the thickness of the feather. After this step, the two surfaces are connected by the borders, which will be triangulated using the same process (represented in blue in the figure). At the end, we have a closed mesh representing the overall shape of the feather, able to be submitted to the simulation by the MCLONE. This structure is only required for MCLONE simulation, so this 3D model will not be used for the rendering of the feather.

4.1.2 Simulating the Pattern

In this step we run a MCLONE simulation onto the 3D feather generated by the previous step, in order to create the desired visual pattern. We run the simulation, and as output we have the desired pattern described in a file that contains all the cells of the pattern, with respective positions and types. The graphical representation of these cells is illustrated in Figure 4(a), while the final pattern is presented in Figure 4(b). This pattern is obtained by computing the Voronoi diagram for all cells in Figure 4(a). The use of the Voronoi Diagram to represent the pattern and the number and meaning of parameters is fully explained in (Walter et al., 1998).

4.2 Rendering

Once the structure for one feather is built, we can (i) render it in a non-realistic way; (ii) render it with texture that can be acquired from feather pictures (direct texture mapping), or (iii) we can generate textures procedurally using the MCLONE model (transferring the pattern to the barbs). The last 2 approaches use BTF to improve the realism.

For each feather, we maintain a set of control points defining the rachis and the barbs in each vane (left and right). Sampling on Bézier curves is user driven and allow us to generate feathers in multiple levels of resolution, from coarse (few sampling points) to fine (many sampling points). This feature could be used, for instance, to adjust rendering according to distance from camera to feather. Results in this paper were generated using 30 sampling points. The way the feather is constructed allows us to easily compute texture map coordinates, since the Bézier curves (rachis and barbs) are already parameterized in the $[0,1]$ interval.

4.2.1 Transferring the Generated Pattern to the Barbs

In case we are using the pattern generated by the MCLONE, we have a pattern simulated directly on the surface derived from the structure of the feather built by the user. In this context, the rendering step needs to transfer this pattern onto the barbs of the feather. During the rendering of the segments describing the barbs, we need to find out the color indicated by the pattern. In a intuitive way, we must find the nearest cell in the pattern for that segment, obtaining type and consequently color, finally rendering the segment using this color. We solve this problem using a *kd*-tree. Figure 5 illustrates the problem.

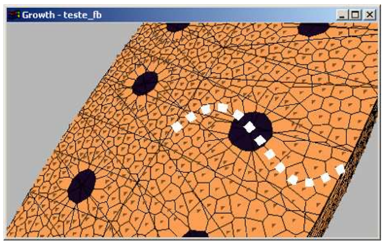


Figure 5: Searching for the nearest cell, when transferring the pattern to the barbs. Dotted line in the picture represents the barb.

4.2.2 Generating and Mapping a Btf

Bidirectional Texture Functions (BTFs) are used to capture the mesostructure's details of an object and represent those details in the texture, improving the realism. In the case of feathers, the interconnection between barbs-barbules plays a very important role in the final appearance of a feather. In order to represent these details in our model, we based our solution in an approach proposed in (Chen et al., 2002). We built the geometric structure of the interconnection between barbs-barbules, an rendered it from several different combinations of viewing and lighting conditions, using POV-Ray. As a result, we have an image of the structure for each different combination sampled. Since this is an off-line step, we can use complex geometries and sophisticated lighting models.

During rendering, we map the BTF texture to simulate the barb-barbules interconnections. Each barb B is described as a polyline L_B with vertices x_0, ..., x_n. When the barb is rendered, instead of rendering only a Bézier curve, a quadrilateral is built for each segment of the polyline L_B . For each of these segments, we obtain lighting and viewing directions. With these directions, we can extract the 1D texture of the image built off-line, map it onto the quadrilateral representing the segment, and merge it with the RGBA color that was obtained from the texture of the MCLONE pattern, resulting in the final pixel's color. In order to reduce aliasing and the size of the texture, we apply supersampling when generating the BTF, filtering it with a gaussian kernel.

5 RESULTS

In this section, we show three results of our approach. All results are realtime, considering that the computation of the MCLONE pattern and BTF are done offline. A detailed explanation about the MCLONE parameters can be found in (Walter et al., 1998). We list the parameters used for these results in Tables 1 and 2.

In Figure 6, we can see that our approach simulates in a realistic way the eyespots found in peacock feathers. In addition, using BTF improves the overall visual appearance of the result, when compared with the rendering using only curves. In other approaches that use Reaction-Diffusion (Prum and Williamson, 2002) such results could not be achieved. We can notice also that for these results we have two set of values for the MCLONE parameters. This is because in order to synthesize the eyespot, we used MCLONE *cascade simulations*, where the result of one simulation is used as input for the second.

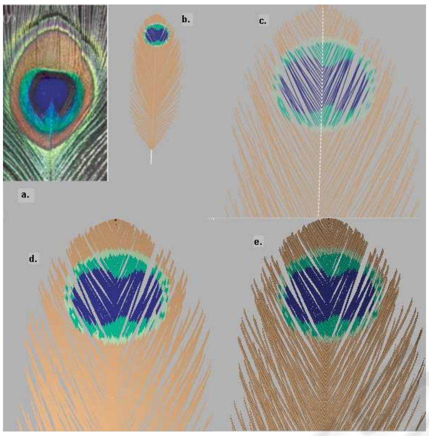


Figure 6: Central Spot *(Pavo cristatus, Phasianidae)*. a) Image of real feather. b) Feather rendered with Bézier. c)Bézier curves's details. d) Polyline rendering. e) Rendering with polylines and BTF.

With the MCLONE we are able to generate several different visual feather patterns. Together with the set of parameters proposed to model the structure of feathers, we have a robust model that allows us to generate procedurally all types of feather structures and many of the visual patterns as well. One of the main advantages of using a procedural mechanism for pattern generation is the flexibility. If we have a texture sample extracted from a picture, for instance, all the models where we map it would have exactly the same appearance, that came from the picture. If using a procedural mechanism, on the other hand, we can generate many slightly different patterns, which is much more similar to the way Nature works. As examples we show Figures 7 and 8. Both illustrate typical spots and stripes found in many feathers. For Figure 8, we can see that we have two different values for each parameter for modelling the structure, and the parameter I_s is set. This means that we divided the feather into two different segments, each one using a set of parameters.

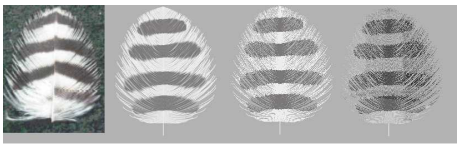


Figure 7: Stripes *(Phloeceastes guatemalensis)*. First picture is from a real feather, second result generated only with Bézier curves, third with *polylines.* and last with *polylines* and BTF.

Although not very extensive, these results confirm the potential of an integrated approach for modelling feathers as a whole, shape and visual together. More

Table 1: Values of the parameters for the peacock feather (fig. 6). $\rho = 18$, $w_r = 3.0$ e $time = 40$, same in both simulations.

Model	N_b	F_{pv}	F_b	S_v	U_f	U_c	I_s	G_d/I_c			
MCLONE	w_d	mit F	mit B	mit M	mit I	αFF	αBB	αMM	αII	mut FM	*no. cells*
Figure 6	180	0.045	0.2	T	T	F	-	1.0/0.25	-	-	-
	0.15	5	150	7	0	0.6	0.4	0.95	0.0	0.3	45/5000/82/0
	0.1	160	160	160	5	0.2	0.2	0.3	0.95	-	45/5000/81/821

Table 2: Values of the parameters for results shown in Figures 7 and 8.

Model	N_b	F_{pv}	F_b	S_v	U_f	U_c	I_s	G_d/I_c	
MCLONE	ρ	w_r	*time*	w_d	mit F	mit B	αFF	αBB	*no. cells*
Figure 7	250	0.135	0.2	T	T	F	-	2.5/0.1	-
	18	2.4	30	0.1	10	150	0.9	0.6	2029/4664
Figure 8	10	0.200	0.35	T	T	F	0.2	2.0/0.1	-
	60	0.035	0.2	T	T	F	-	-	-
	18	3.0	55	0.1	6	150	0.9	0.6	506/4999

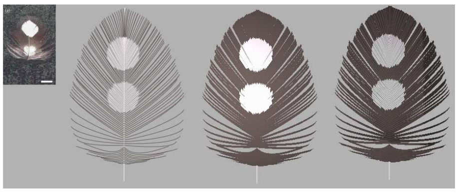


Figure 8: Central and circular Spots (*Chrysocolaptes lucidus*). First picture is from a real feather, second result generated only with Bézier curves, third with *polylines.* and last with *polylines* and BTF.

sophisticated visual patterns could be achieved with elaborated MCLONE simulations.

6 CONCLUSIONS

This work introduced an approach to compute integrated generation of shape and visual pattern in feathers. The approach presented here includes both the geometric model of a single feather and the visual pattern of the same. We have also extended the Clonal Mosaic Model, applying it for procedural generation of feather visual patterns. In comparison with other approaches (Dai et al., 1995), the set of parameters that defines the model showed greater flexibility, where we can simulate several different types of feathers. Besides that, our approach includes a model to generate the patterns procedurally, in opposition to other approaches where the textures are extracted from pictures of real feathers (Chen et al., 2002; Streit and Heidrich, 2002). The use of a procedural model brings flexibility as needed to generate several similar textures, with only small variations.

REFERENCES

Chen, Y., Xu, Y., Guo, B., and Shum, H.-Y. (2002). Modeling and rendering of realistic feathers. *ACM Transactions on Graphics*, 21(3):630–636.

Dai, W.-K., Shih, Z.-C., and Chang, R.-C. (1995). Synthesizing feather textures in galliformes. *Computer Graphics Forum*, 14(3):407–420.

Dana, K., Ginneken, B., Nayar, S., and Koenderink, J. (1999). Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics*, 18(1):1–34.

dos Reis, A. N., Mombach, J. C., Walter, M., and de Avila, L. F. (2003). The interplay between cell adhesion and environment rigidity in the morphology of tumors. *Physica A*, 322:546–554.

Fleischer, K., Laidlaw, D., Currin, B., and Barr, A. (1995). Cellular texture generation. In Cook, R., editor, *Computer Graphics (SIGGRAPH 95 Conference Proceedings)*, pages 239–248. Addison-Wesley. ISBN 0-201-84776-0.

Franco, C. and Walter, M. (2002). Modeling and rendering of individual feathers. In Goncalves, L. and Musse, S., editors, *Proceedings of SIBGRAPI 2002 - XV Brazilian Symp on Comp Graph and Image Processing*, pages 293–299, Los Alamitos. IEEE Press.

Freethy, R. (1982). *How Birds Work*. Blandford Press, USA, 1 edition.

Lied, T. P. and Walter, M. (2002). Procedural butterfly wing patterns. In Goncalves, L. and Musse, S., editors, *Proceedings of SIBGRAPI 2002*, page 424, Los Alamitos. IEEE Press.

Prum, R. O. (1999). Development and evolutionary origin of feathers. *Journal of Experimental Zoology (Mol. Dev. Evol.)*, 285:291–306.

Prum, R. O. (2001). A theory of the growth and evolution of feather shape. *Journal of Experimental Zoology (Mol. Dev. Evol.)*, 291:30–57.

Prum, R. O. and Williamson, S. (2002). Reaction-diffusion models of within-feather pigmentation patterning. *The Royal Society*, 269:781–792.

Soler, C., Cani, M.-P., and Angelidis, A. (2002). Hierarchical pattern mapping. *ACM Transactions on Graphics*, 21(3):673–680.

Streit, L. and Heidrich, W. (2002). A biologically-parameterized feather model. *Computer Graphics Forum*, 21(3):565–574.

Tong, X., Zhang, J., Liu, L., Wang, X., Guo, B., and Shum, H.-Y. (2002). Synthesis of bidirectional texture functions on arbitrary surfaces. *ACM Transactions on Graphics*, 21(3):665–672.

Turing, A. (1952). The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society B*, 237:37–72.

Turk, G. (1991). Generating textures for arbitrary surfaces using reaction-diffusion. *Computer Graphics (Proceedings of SIGGRAPH 91)*, 25(4):289–298.

Turk, G. (2001). Texture synthesis on surfaces. In *Proceedings of ACM SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 347–354.

Walter, M., Fournier, A., and Menevaux, D. (2001). Integrating shape and pattern in mammalian models. In *Proceedings of SIGGRAPH 2001*, pages 317–326, Los Angeles. Addison-Wesley. ISBN 1-58113-292-1.

Walter, M., Fournier, A., and Reimers, M. (1998). Clonal mosaic model for the synthesis of mammalian coat patterns. In *Proceedings of Graphics Interface 1998*, pages 82–91.

Wei, L.-Y. and Levoy, M. (2000). Fast texture synthesis using tree-structured vector quantization. In *Proceedings of ACM SIGGRAPH 2000*, Computer Graphics Proceedings, Annual Conference Series, pages 479–488.

Wei, L.-Y. and Levoy, M. (2001). Texture synthesis over arbitrary manifold surfaces. In *Proceedings of ACM SIGGRAPH 2001*, Computer Graphics Proceedings, Annual Conference Series, pages 355–360.