

# Procedural Generation of Hair

Vinícius Jurinic Cassol  
Fernando Pinho Marson  
Soraia Raupp Musse

Graduate Programme in Computer Science  
PUCRS - Av. Ipiranga, 6681, Porto Alegre, RS, Brazil



**Figure 1:** *Different types of hair generated through variation of parameters.*

## Abstract

This paper presents a hair generation technique for cartoon and Anime characters. The main goal of this work is to provide a procedural generation technique, robust enough in order to create different types of hairs. We propose a parametric model that can produce diversity of obtained results through parameters variation. The model is organized in three steps: firstly, the strands generation, followed by positioning them in the head of the characters. Finally, the third step is responsible for providing cartoon rendering, including common characteristics in cartoon shading, as the black edges and solid colors. Visual inspection of obtained results indicate that generated hairs are coherent with expected in terms of visual aspects.

**Keywords::** Hair Generation, Cartoon Hair, Procedural Modeling.

## Author's Contact:

vinicius.cassol@acad.pucrs.br  
{fernando.marson, soraia.musse}@pucrs.br

## 1 Introduction

Hair modeling and rendering in cartoon characters can be applied in animation movies and computer games. Hair effects can be used to distinguish among different cartoon characters, especially in Anime [Shin et al. 2006] characters. When characters faces are similar, they can be distinguished as a function of their hair geometry and color. Indeed, hair can be considered as an identifier or a factor who compose the characters identity. We can observe some of these factors in human beings, e.g. hair color, length and type: straight or curly. Such parameters compose the individual identity and, in our computational model are described in hair generation and rendering process.

In terms of appearance, computer graphics techniques have been used to provide different types of hair representation. Indeed, such techniques can provide photorealistic and non-photorealistic rendering. The photorealistic rendering techniques represent characteristics from the real world, i.e. they imitate real effects. On the other hand, the non-photorealistic rendering techniques are used to represent other visual languages, e.g. cartoon appearance.

Our work presents a procedural hair generation technique focused

on cartoon and Anime characters. In order to procedural generate hairs, we propose a parametric model which can produce visually different results.

## 1.1 Procedural Generation

Procedural techniques can be defined as segments of code or algorithms that specify some model feature or a computer generated effect [Ebert et al. 2002]. Such techniques have been used since the beginning of Computer Graphics, in order to generate geometrical models, create textures and animation of objects and characters.

The main characteristic of procedural methods is the abstraction. Instead storing all details of a particular shape or animation, the main concept of procedural technique aims to provide details in-game (or during the simulation). Consequently, the algorithms should be robust enough to allow coherent re-creation, for instance each time a specific model should be re-created, e.g. a tree, it should be made through a computational procedure, and not loading files. Yet, procedural methods should be able to create different trees, while keeping the expected global picture of a tree.

The description of a robust and coherent procedural model allows the generation of shapes/animations, without required extra files. Indeed, procedural methods should generate different characteristics of obtained results. That is why it uses parameters which are associated with generated characteristics. Consequently, generated distinct objects or animations can share common features. For instance, one can mention an algorithm to codify the general concept of chair [Morkel and Bangay 2006]. There are many combinations of different features that can be found in chairs: height, material, type of seat and backrest, among others. However, instead of storing pre-computed models, the procedural methods aims to generate them. A strong point is the flexibility to generate diversity into obtained results. The model proposed in this paper aims to generate cartoon hair models and appearance through a procedural method.

The paper is organized as follows: in Section 2, we present an overview of published work in hair modeling area, while in Section 3, we present our procedural generation model. The prototype and obtained results are presented and discussed in Section 4, and finally, in Section 5, we conclude the paper with the possibilities for future work.

## 2 Related Work

Procedural methods of hair generation can produce characters for different types of applications. According to the application, the required level of realism can vary from photorealistic to non-photorealistic images. Both of representations can be used in different applications for computer games and animation movies.

Indeed, realistic hair rendering is still considered a challenge, since each hair, in each strand, should be individually modeled. Please, notice that a common hair volume has about 100.000 strands [Ward 2005]. The individual representation of any strand is necessary when we need to represent the perfection of the hair shape and motion to look like a human hair. In last years, many methods have been developed with the main goal of producing realistic hair models [Volino and Magnenat-Thalmann 2004], [Bertails et al. 2006], [Ward et al. 2007b].

Focused on photorealistic rendering, Magnenat-Thalmann et al. [Magnenat-Thalmann et al. 2000] presents a survey paper in hair generation methods where authors analyze the main presented challenges. Two are important requirements: firstly, the shape modeling considering the perfection of the human hair, and secondly, the computational time of proposed methods. Recently, another survey was produced by Ward [Ward et al. 2007a] and again presents the state-of-the-art in hair generation in terms of style generation, simulation and quality of rendering.

On the other hand, non-photorealistic rendering process do not represent the object or character as it is in real world. This kind of representation provides stylized representation, usually applicable in cartoon characters. Due to the visual aspect of cartoons, it is not necessary to model and generate each hair strand. Considering this perspective, it is possible to model specific shapes and rendering often presented in cartoons characters. One example of hair modeling and cartoon rendering is presented by Shin [Shin et al. 2006], who develop a model based on particle system, using GPU and billboards in order to produce rendering highlights. To any generated strand, the silhouette is highlighted as expected in cartoons.

Sugisaki [Sugisaki et al. 2004] presents a hair animation technique based on Physical Simulation. The process begins with the extraction of hair motion from a existing cel cartoon animation. With this data, hair motion database is built, which is physics-based. In this method, an human intervention is required in order to produce visually good results. Yet, this technique provides hair motion through deformation functions applied in strands angles. Another work from the same authors [Sugisaki et al. 2006] describes a new model to animate hair motion where existing cel are used. The sequences are extracted from a database and parameters are applied in order to control the hair stiffness. Physically-based equations are used to produce hair shape and motion.

Noble [Noble and Tang 2004] shows an approach to generate and animate cartoons hair. Such technique uses NURBS surfaces [Piegl and Tiller 1997] to compose a geometric mesh which produces different hair shapes and motion. An animated NURBS volume define a primary shape and motion. By this way, along the surface are generated clumps of hair geometry. With the technique developed by Noble, artists have a flexible tool to provide hair generation and animation in real-time.

Yet, considering non-photorealistic rendering techniques, different hair characteristics can be simulated. Moreover, it is defined by the type of character where we will apply the hair. We can use a cartoon rendering to produce these characteristics in a draw cel character, e.g. Decaudin [Decaudin 1996] can be considered as a cartoon rendering forerunner. In such technique the author consider some typical cartoon features as to use uniform colors in the draw fill and always use the same edge black thickness in the character silhouette. In this way, other points can be considered in the rendering process like shadows and lights.

## 3 Procedural Hair Generation

In this section, we describe the method for hair generation, the proposed steps as well as the overall architecture. The procedural gen-

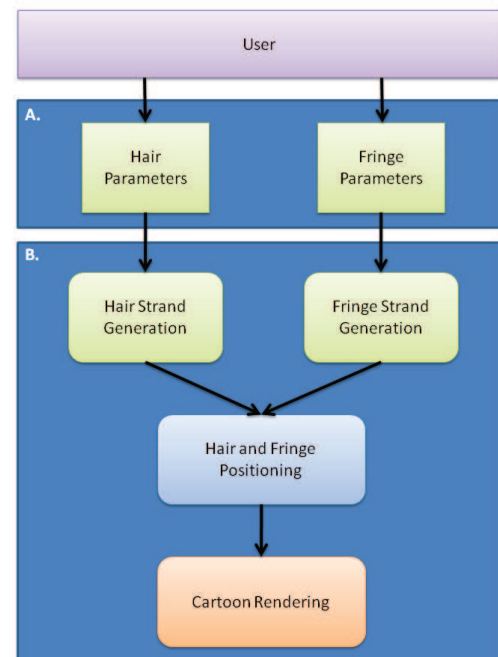


**Figure 2:** The pipeline of hair generation process.

eration of hair consists in three main steps:

- **Parametrization and generation of strands:** The user must set the desired features for the hair and the fringe. These settings are made through a graphical interface. This phase ends with the generation of each strand based on parameters selection.
- **Strands positioning in the head:** After the generation, the strands and fringe must be correctly positioned and aligned with the character head.
- **Cartoon Rendering:** In this step, we apply cartoon rendering characteristics in the generated geometry.

The pipeline of proposed model is presented in the Figure 2. Box A represents the user inputs that are used to define the hair features and box B shows the main steps involved in the hair generation process. In the next Section, the steps will be more detailed.

### 3.1 Strands Generation

As it was presented before, our technique is based on procedural generation. By this way, the model need some parameters definition to produce different hair models. To allow the user to easy and fast parameters definition and variation, we developed a prototype wit and a graphic interface. In this prototype the parameters can be defined or changed and the user can see and validate the results in real time. This prototype will be detailed in the Section 4.

Our model can generate three different styles of hair: straight, wavy and curly. The modeling of each wick of hair is given initially by setting a guideline. This guideline will be used as a basis to generate the final shape of strand. The amount of vertex in the guideline allows to vary the obtained result in a wavy or curly hair. The other parameters are the desired style of hair, the width, the initial thickness and the length of each match. All these parameters are defined in the interface.

Every vertex in the guideline is used to create other four points: two points to specify the width in the  $x$  axis and two points for the thickness in the  $z$  axis. In Figure 3, we can see the guideline vertex (red) and the four auxiliaries points (blue). The distance between the guideline points (in  $y$  axis) is given by a step times the length. This step is obtained by

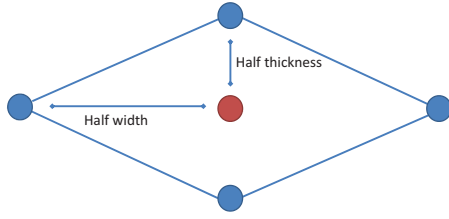


**Figure 3:** Generation of auxiliary points (blue points) based on a guideline vertex (red point).

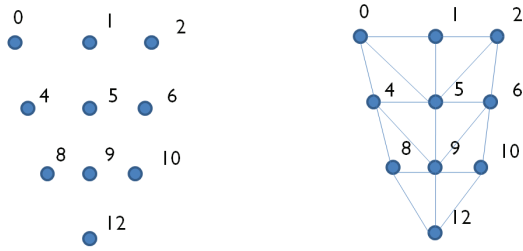


**Figure 4:** Vertex distribution in the strand (left) and the strand triangularization (right).

$$step = (1/\text{number of vertex}). \quad (1)$$

Based on the number of vertex in the guideline, we can calculate how many vertex and faces will compose the surface strand's:

$$\text{surface vertex} = (\text{number of vertex} \cdot 4) + 1, \quad (2)$$

$$\text{surface faces} = ((\text{number of vertex} - 1) \cdot 8) + 4. \quad (3)$$

After calculating the number of surfaces vertex, they are organized in a triangle shape where the larger size is given by the user and the others are linearly distributed, as illustrated in Figure 4 (left). Based on the vertex distribution, we can do the strand triangularization, generating the mesh illustrated in the Figure 4 (right). In this example, the vertex number 3, 7 and 11 located at the back face of strand generate other triangles; e.g. 0, 1, 3 and 4, 5, 7.

In the straight and wavy hair style, the entire width and thickness defined by the user, are considered from the beginning of the hair (from the top of head). For each point at the guideline from the beginning to the end of the hair, the width is reduced, causing an effect into the strand as seen in Figure 5. If the process is interrupted by the user before the end, we can create an additional effect: the strand seems to be cut. The distinction between straight and wavy hair is obtained applying a variation wavy factor, positive and negative alternately, in the  $x$  coordinate, that is given by

$$\text{wavy factor} = (1 - \text{step}_n). \quad (4)$$

A peculiar feature of the wavy hair is the location where the wavy effect should start. By this way, we can define the point of the strand where the frizz should begin. With this possibility is not necessary to produce always the same strand wavy.

For the curly hair style, the process is a little bit different. Firstly, it is defined the angle( $\theta$ ) that the curly hair should have to provide the curly effect, e.g. 20 degrees. The  $y$  value is calculate according Equation 1, but the value of  $x$  and  $z$  are given, respectively by

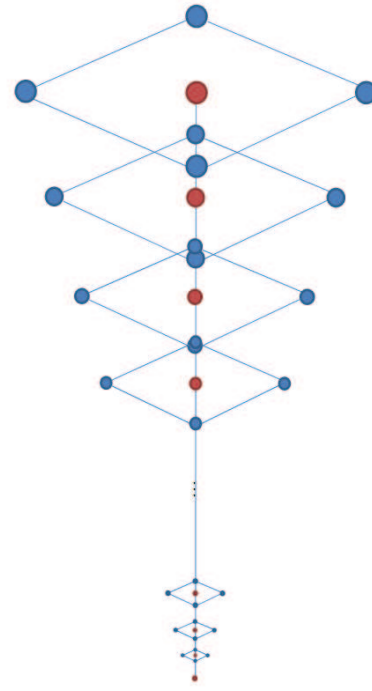


**Figure 5:** Generation of a straight strand considering the position of the auxiliary points (blue points) with the points of the guideline (red points).

$$x = \sin(\theta) \times \left(\frac{\text{width}}{4}\right), \quad (5)$$

$$z = \cos(\theta) \times \left(\frac{\text{width}}{4}\right). \quad (6)$$

After each step, it is added 20 degrees into the angle. Different values can generate distinct results. The resulting points from this process are used to building a polygonal mesh. The Figure 6 shows the three styles of strands generated by our technique.

The next Section presents the next step in the pipeline to place artificial hair in characters.

### 3.2 Positioning of Strands in the Head

After the generation of each strand, it is necessary to place them correctly positioned and aligned with the head of the character. This

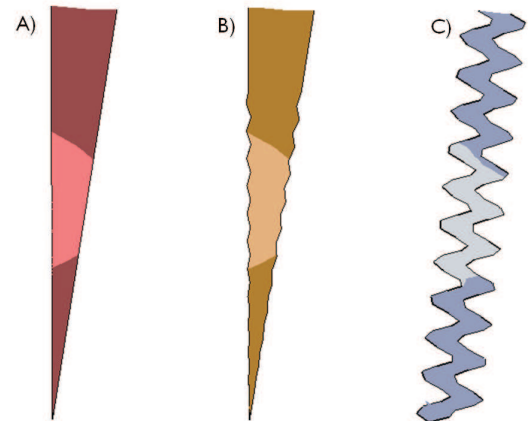


**Figure 6:** Styles of hair generated with our technique: a) straight strand, b) wavy strand with the definition of a point to start the frizz, c) curly hair.

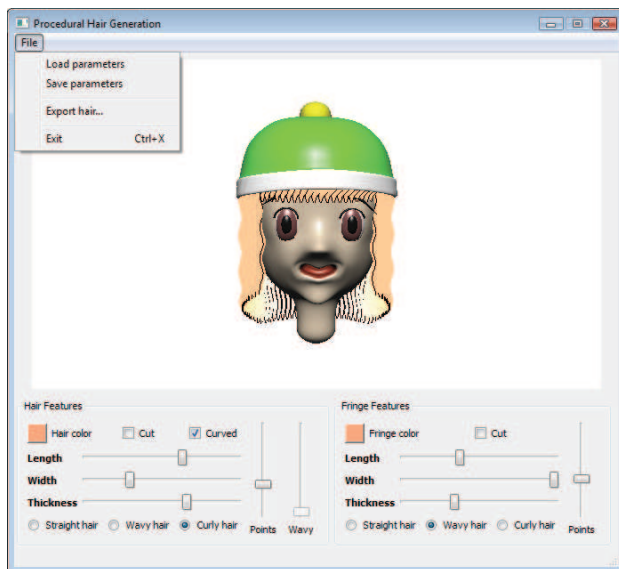


Figure 7: The interface of the prototype.

process is composed by four steps. At first, the initial point of the guideline is translated to a specific  $y$  value in the head where strand should start. After, it is necessary to calculate the angle between the normal vector of the strand ( $\vec{u}$ ) and the normal vector of each triangle that will receive the strand ( $\vec{v}$ ). The strands are generated at position and orientation.

### 3.3 Cartoon Rendering in This Work

With a cartoon rendering technique, it is possible to apply a cartoon style in objects or characters, like an animated movie or represent comics draws. There are some peculiar features about cartoon. In our results, we can identify some features of this technique as used in the Decaudin model [Decaudin 1996]:

- Uniform colors in the draw fill: In the traditional cartoons draws we can see the use of solid colors in the draw fill, as we used to produce our hair. Otherwise, in a sophisticated rendering process, it is possible to use different levels of color intensity. This levels can provide some different results in the character rendering.
- Use the same edge black thickness to define the character silhouette: This is an important point to allow to identify different characters in the same scene. By this way, we can avoid the overlap among the characters or objects rendered. This point is verified in the strands generation process. In this process, we identify and stored the external edge to allow to render this edges in black color.

## 4 Prototype and Results

Based on the proposed model, a prototype was developed to validate the technique and evaluate obtained results. It was coded using C++. The QT framework was used to create the interface. A head was modeled in order to test the correct positioning of strands. Figure 7 presents the interface of the prototype.

The needed parameters are separated into two different sets. First one allows to control just the fringe (frontal hair). The other set, controls the rest of hair. To the both sets, the user can choose the color, the length, the width, the thickness of hair or the fringe. The style of hair (straight, wavy or curly) also can be applied individually. For curly or wavy hair, it is possible to modify the number of sampled points. This feature allows to create more defined strands. Just for the wavy hair style, is feasible specify where the frizz point should start, as we can see in the Figure 8. Also it is possible to create a cut effect, i.e. the strand seems to be cut. Yet, in the Figure 7, we can see the last parameter: the curved hair. With this parameter it is possible creates short strands on the sides and longer at

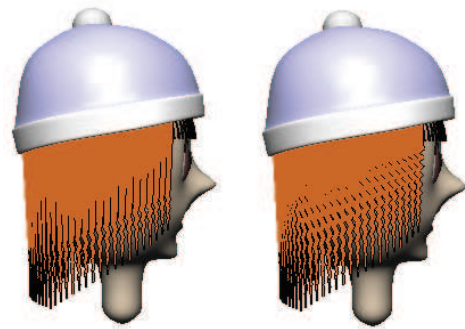


Figure 8: Different results obtained by the variation of the wavy start point.



Figure 9: Different results by using curved and cut parameters.

the nape of neck. The Figure 9 shows results obtained by applying curved and cut parameters.

The prototype allows to save the used parameters into a configuration file in order to use them in a next time. Also it is possible to export the generated polygonal mesh to the OBJ<sup>1</sup> format, thus allowing its use in games and other graphical applications.

With the parameters variation, we can present different results in the Figure 10. An interesting feature of our model is the real time hair generation. This is possible because our model has low computer cost. The time for hair generation is basically the time for interface manipulation, since the model is generated in interactive frame-rates.

## 5 Conclusion and Future Work

This paper presents a procedural technique to generate different kinds of hair that can be applied to cartoon characters. The main feature of our method is the capability to simulate, analyze and validate visually the results in real-time. This is provided by a parametric model that can be changed through the specification of parameters provided by the user. Another important feature is the simplicity of our method. It can be easily reproduced and modified to add new hair styles.

As a future work we want to improve our model by handling other parameters generating a varied range of results. Currently, the strands are not generated from the top of hair. This feature will

<sup>1</sup>A geometry definition file format first developed by Wavefront Technologies



**Figure 10:** Results produced by our model.

be implemented soon. Other question is the possibility of using different head models. To do this, it is necessary to create a tool that allow to specify the faces where the hair will grow up. In this sense, a metalanguage can be used to import and export geometry.

## References

- BERTAITS, F., AUDOLY, B., CANI, M.-P., QUERLEUX, B., LEROY, F., AND LÉVÊQUE, J.-L. 2006. Super-helices for predicting the dynamics of natural hair. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*, ACM, New York, NY, USA, 1180–1187.
- DECAUDIN, P. 1996. Cartoon looking rendering of 3D scenes. Research Report 2919, INRIA, June.
- EBERT, D. S., MUSGRAVE, F. K., PEACHEY, D., PERLIN, K., AND WORLEY, S. 2002. *Texturing and Modeling: A Procedural Approach*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- MAGNENAT-THALMANN, N., HADAP, S., AND KALRA, P. 2000. State of the art in hair simulation. In *International Workshop on Human Modeling and Animation*, Computer Graphics Society, 3–9.
- MORKEL, C., AND BANGAY, S. 2006. Procedural modeling facilities for hierarchical object generation. In *Afrigraph '06: Proceedings of the 4th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa*, ACM, New York, NY, USA, 145–154.
- NOBLE, P., AND TANG, W. 2004. Modelling and animating cartoon hair with nurbs surfaces. In *CGI '04: Proceedings of the Computer Graphics International*, IEEE Computer Society, Washington, DC, USA, 60–67.
- PIEGL, L., AND TILLER, W. 1997. *The NURBS book (2nd ed.)*. Springer-Verlag New York, Inc., New York, NY, USA.
- SHIN, J., HALLER, M., AND MUKUNDAN, R. 2006. A stylized cartoon hair renderer. In *ACE '06: Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology*, ACM, New York, NY, USA, 64.
- SUGISAKI, E., YU, Y., ANJYO, K., AND MORISHIMA, S. 2004. Cartoon hair animation based on physical simulation. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Posters*, ACM, New York, NY, USA, 27.
- SUGISAKI, E., KAZAMA, Y., MORISHIMA, S., TANAKA, N., AND SATO, A. 2006. Anime hair motion design from animation database. In *CyberGames '06: Proceedings of the 2006 international conference on Game research and development*, Murdoch University, Murdoch University, Australia, Australia, 33–40.
- VOLINO, P., AND MAGNENAT-THALMANN, N. 2004. Animating complex hairstyles in real-time. In *VRST '04: Proceedings of the ACM symposium on Virtual reality software and technology*, ACM, New York, NY, USA, 41–48.
- WARD, K., BERTAITS, F., KIM, T.-Y., MARSCHNER, S. R., AND CANI, M.-P. 2007. A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics* 13, 2, 213–234. Member-Lin, Ming C.
- WARD, K., GALOPPO, N., AND LIN, M. 2007. Interactive virtual hair salon. *Presence: Teleoper. Virtual Environ.* 16, 3, 237–251.
- WARD, K. A. 2005. *Modeling hair using levels of detail*. PhD thesis, Chapel Hill, NC, USA. Adviser-Lin, Ming C.