Simulation-Based Cartoon Hair Animation

Eiji Sugisaki
Waseda University
3-4-1 Okubo, Shinjuku-ku,
Tokyo Japan, 169-8555
eijil11@toki.waseda.jp

YizhouYu
University of Illinois
201 N. Goodwin,
Urbana, IL U.S.A 61801
yys@cs.uiuc.edu

Ken Anjyo
OLM Digital Inc.
1-8-8 Wakabayashi,
Setagaya-ku, Tokyo Japan
anjyo@olm.co.jp

Shigeo Morishima
Waseda University
3-4-1 Okubo, Shinjuku-ku,
Tokyo Japan, 169-8555
shigeo@waseda.jp

ABSTRACT

This paper describes a new hybrid technique for cartoon hair animation, one that allows the animators to create attractive and controllable hair animations without having to draw everything by hand except a sparse set of key frames. We demonstrate how to give a cel animation character accentuated hair motion. The novelty of this approach is that we neither simply interpolate the key frames nor generate the movement of the hair only using physical simulations. From a small number of rough sketches we prepare key frames that are used as indicators of hair motion. The hair movements are created based on a hair motion database built from physical simulations custom-designed by the animator. Hair animations with constraints from the key frames can be generated in two stages: a matching process to search for the desired motion sequences from the database and then smoothly connect them; the discrepancies between the database sequences and the key frames are interpolated throughoutthe animation using transition function.

Keyword :

Cel Animation, Cartoon Animation, Hair Dynamics, 3D Animation

1 INTRODUCTION

Hair movement in cel character animation is sometimes inconsistent. For example, there may be inconsistencies in the number of strands or locks of hair, as can easily be seen in the images in Fig. 1. The representation of hair is thus a specialized work in cel animation. In fact, Cartoon hair representation is very difficult to achieve in computer graphics. This is because all of the hair attributes may not be consistent between camera positions. Although physics equations can be used to obtain physically correct movement, it is not always the movement for which the animator is looking. Since the movement of hair in cartoon animation sometimes carries meaning, the animator may be looking for something that exists only in his or her imagination. This means that the characteristics of the hair (modeled shape, number, etc.) between key frames may actually have not to agree. Even though the frames are physically inconsistent, the results can be quite convincing. This is the most difficult part of creating cartoons using computer graphics and is the reason why cartoon hair animation has been done by hand. This requires a lot of human labor. In the full-cel animated movie "Princess Mononoke", [Dvd 01a] for example, it takes a month to complete five minutes animation in windy scenes.

The representation of hair motion is very important for computer graphics regardless of whether the graphic is simulation or cartoon animation. Moreover, while cartoon-like hair description plays a crucial role in making cartoon character animation impressive, hair simulation for cartoon animation is a challenging task because the human eyes discern the subtleties of hair motion and readily notices anything unnatural.

Although attractive cartoon hair animation is often seen on TV and in movies, there are few animators who can achieve impressive hair motion in cel animation. The hair motion as the camera angle changes is particularly hard to draw by hand. It requires the instincts of an expert animator and is very time consuming. While the work of a very skilled animator is very demanding, there has been little research directed at solving such time-consuming problems as the cel animation of hair motion. [Pno04a]

Figure 1. Example of cel images

1.1 Overview

We develop a method for creating cartoon hair animation in computer graphics that easily retains the "anime-like" aspect of animation. Our goal is to produce an interesting cartoon hair animation that matches the hair designs shown in hand-drawn rough sketches. Our approach is a hybrid one taking advantages from both keyframe interpolation and physical simulation. The input is a sparse set of hand-drawn hair sketches for a cartoon character in the key frames. These sketches illustrate the target cartoon features.

A crucial step in our approach is building a motion database in advance. Building a database with an exhaustive list of motions is infeasible given the sheer number of hair strands and the number of vertices on each strand, so we build an "animator-directed motion database" with a chosen set of force impulses that can be used to generate sequences that potentially match the hair sketches. That is, we build a custom-designed database for each hair animation. It takes about 30 minutes to build such a database.

One way to reduce an animator`s workload is to consider hair geometry in three dimensions. Although cel animation uses a two-dimensional structure, we use a three-dimensional structure so that hair motion can be created more easily. It is difficult, however, to use three-dimensional structures to fully express in two dimensions the inconsistencies that are peculiar to animation. We therefore propose a method for hair animation that matches data between the three-dimensional hair models and the key frames of rough sketches made by animators or directors that indicate hair motion. More specifically, we use the three-dimensional hair models and the impulse force to generate interactive and attractive motions, which we use to construct a hair motion sequence database. We then try to match the rough sketches to the three-dimensional hair motion sequence database data by projecting the sequence data onto the rough sketches. Once we find a match, we create a cartoon hair model to interpolate between the rough sketches.

2 Related Work

This overview of related work is limited to previous work on hair dynamics, focusing on explicit hair models. These models consider the shape and dynamics of each strand. While they are especially suitable for the dynamics of long hair, they do not consider cartoon simulations. Anjyo et al. [Ken92a] used a simplified cantilever beam to model hair and used one-dimensional projective differential equations of angular momentum to animate strands. Rosenblum et al. [Rer91a] and Daldegan et al. used sparse characteristic hair to reduce computation time. Kim and Neumann [Tae02a] presented an artful method for creating hairstyles that uses Multi-resolution Hair Modeling (MHM) system, which is based on

the observed tendency of adjacent hair strands to form clusters at multiple scales. Yu et al. [Yyu01a] also presented a method for creating hairstyles. These advances greatly improved hair expression in computer graphics.

Several researchers have proposed novel approaches to hair-hair interaction. Hadap and Magnenat-Thalmann [Had00a] proposed modeling dense dynamic hair as a continuum by using a fluid model for lateral hair movement. Hair-hair collision is approximated by the pressure term in fluid mechanics while friction is approximated by viscosity. Hair-air interaction is approximated by integrating hairs with an additional fluid system for the air. Chang et al. [Jtc02a] modeled a single strand as a multibody open chain expressed in generalized coordinates. Dynamic hair-to-hair collision is solved with the help of auxiliary triangle strips among nearby strands. The input to their simulation algorithm is an initial sparse hair model with a few hundred strands generated from their previous hair modeling method. Plante et al. [Epl01a] proposed a "wisps model" for simulating interactions in long hair. Bando et al. [Yba03a] proposed a method in which they model unordered particles that have only loose connections to nearby control points. By freeing particles from some constraints, they are able to animate hair including hair-hair interactions at a reasonable computational cost. Considering hair-hair interaction is also making significant contribution to hair expression in computer graphics.

In terms of cartoon expression, Lasseter [Jon87a] is very likely the first paper to describe the basic principles of traditional two-dimensional hand-drawn animation and their application to three-dimensional computer animation. In this paper, he clearly describes cartoon animation and what it requires of an animator. Paul Noble and Wen Tang [Pno04a] achieved cartoon hair modeling and animation by using NURBS Surfaces to model the primary shape and motion of cartoon character hair.

Rademacher proposed the method that a three-dimensional structure is used in cel animation.[Pau99a] The reference hand-drawn image of the object or character often contains various view-dependent distortions that cannot be described with conventional 3D models. Therefore, to prepare view-dependent models, which consist of a base model, a set of key deformations created by the base model, a set of corresponding key viewpoints, and a given discretional viewpoint, they interpolate the key deformations that are specific to the new viewpoint. They thus capture the view-dependent inconsistencies of the reference drawing.

3 Constructions of Hair Data
from Original Input and 3D Geometry

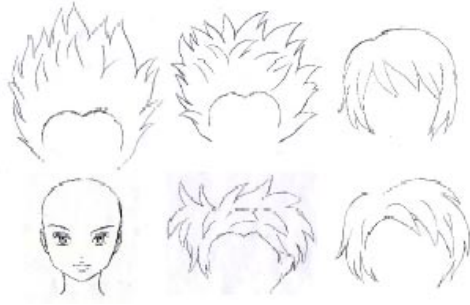Our method requires the preparation of various types of hair data before starting the simulation. We first need

Figure 2 Rough sketches hand-drawn by an animator

to prepare a high-quality two-dimensional cel image of the animated character (like those shown in Fig. 1) and roughly sketched images of the character's hairstyles (Fig. 2). These images are hand-drawn by a skilled animator and are used as original input. This is the only step in which a skilled animator draws images. We also need to prepare a three-dimensional model of the character's head. This model is based on a wire frame model. We obtain the three-dimensional head model by texture mapping. This step is performed by users (animators).

3.1 Making Hair Strands

We also need to obtain position coordinates manually from high-quality two-dimensional cel images drawn by the animator. By obtaining the position coordinates and adjusting them to extract a hair model that is well adapted for the character animation, we create a three-dimensional sparse-hair model. This hair model has boundary lines to form the hair shapes and a centerline to control hair motion (see Fig. 3). These lines are expressed using Catmull-Rom splines. The centerline and boundary lines are connected by weak springs.

3.2 Constructing Initial 3D Hair Model and Converting The Rough sketch Images to data

To construct an initial three-dimensional hair model that matches the character, we use a tool that allows the head model and hairs to be displayed simultaneously in three dimensions. A commercial tool can be used for this step. We also convert the rough-sketch hair images into quantitative data and match this data to hair data in the hair motion database (Fig. 4). Points on the sketched hair strands are plotted interactively. The plotted data is initially two-dimensional. Users perform all of these steps.

4 Creating Hair Motion

In this section, we describe how we create hair motion from a database and explain the construction of our "designed hair motion database". We also explain the matching process, which uses the similarities between the angles derived from the rough sketches and the data obtained by projecting the three-dimensional hair data from the database sequences onto the image planes
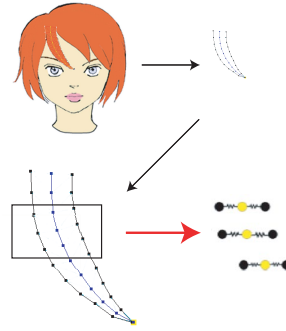

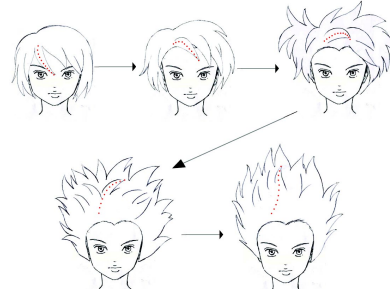**Figure 3 Example of making hair strund steps**

Figure 4 Example of plotting the rough sketch images

defined for the rough sketches. We describe how we obtain the attractive hair shape by applying a deformation function to the differences between the angles. In addition, we describe how we interpolate between the database sequences to maintain smooth transitions between them.

4.1 Designing Hair Motion Database

We apply forces such as those due to wind and head move to our hair model. After designing the forces, we simulate the hair dynamics using an implementation of Featherstone's algorithm for *multi-body dynamic chains.*[Mul01a] [Pli94a] Only the centerline is controlled by this dynamics. Every sequence generates using this method specifies three-dimensional points with velocity vectors.

Compiling a database [Luc02a] [Jpl00a] [Dou03a] has become a common way of handling a corpus or scattered data. We also deal with the hair control point data as a database. A strong point of our method is that this database is custom-designed by an animator for a specific target animation. One unit of database sequence is about five frames long. All movements of hair strands in this database can be designed by an animator so as to reduce database size and enable target hair motions to be extracted. The animator can even define the specific forces needed to obtain a motion that does not conform to general physical laws. The size of the database depends on the trade-off between speed and quality. The database we use in our simulation is not so large, usually less than 5MB. (It depends on how many hairs a character has.) The force we apply is an impulse
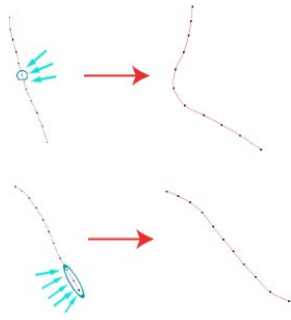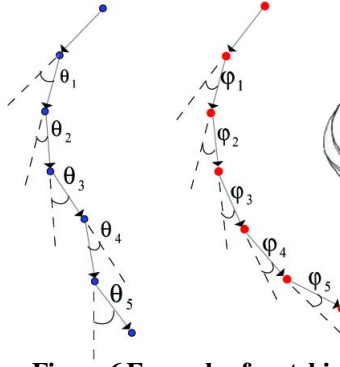
Figure 5 Example of
designing hair motion

Figure 6 Example of matching aspect

Figure 7 Example of applying
RBFs to matching aspect

function. Since the animator needs to design hair that is close to the sketched hair in the key frames, we consider the rough sketches as the indicator of hair motion. For instance, to bend a hair strand dramatically, the animator should only apply a force to the middle part of the hair strand. Moreover, to make a motion in which only the hair tip moves, the animator should define forces applied only to the tip (see Fig. 5).

4.2 Matching Process:

Finding Similar Hair Strands in Database

Our method requires animators to use their intuition to decide the camera positions for the rough hair sketches. Three-dimensional position points from the hair database sequence are then projected onto the image plane of he rough sketch. The data thereby become two-dimensional. Then we carry out matching in the 2D image plane with an x- and y-axes, comparing the angle between hair segments (Fig. 6). More specifically, we measure the difference in the angles between the data made from the rough hair sketches and the data of the hair sequences from the motion database projected onto the image plane. Then we compare this data using Eq. 1 from the hair root to the edge. We select the hair motion sequence that has the smallest error at the key frames.

$$d_{\min} = \frac{\sum_{i=1}^{} \| \theta_i - \phi_i \|}{k} \qquad \| \theta_i - \phi_i \| < th \qquad \textbf{(1)}$$

where θ is the angle made from the rough sketch data, ϕ is the angle made from the data projected onto the rough sketch surface, and i is the angle number counting from the hair strand root. In this equation, the result of each subtraction must be lower than a certain threshold, *th*. If the subtraction result is higher than *th*, we do not choose the database sequence with the minimum error. We thereby obtain from the rough sketch the most similar hair form in the motion database.

4.3 Deformation Function for Angles

Research on scattered interpolation has generally used radial basis functions [Jpl 00a](RBFs), which can be basically expressed as

$$\hat{d}(X) = \sum_{\psi}^{N} W_{\psi} \phi(\| X - X_{\psi} \|) \qquad \textbf{(2)}$$

This interpolation is a linear combination of nonlinear functions of a distance from the data points. It uses the database sequence chosen in the matching process. To transform hair shapes to make them more similar to the hair sketches than to the database sequences obtained in the matching process, we use RBFs. Normally, impressive and exaggerated hairstyles cannot be obtained by using only physical equations. Initially, we used the RBFs to interpolate the discrepancies in the projected hair vertex positions on the image plane of the rough sketch. However, the hairs became awkwardly long (Fig. 7, center) and seemed unnatural. We therefore apply RBFs to the discrepancies in the angles in order to get the target shape. This is done using Eq. 3.

$$d(\phi) = \sum_{i=1}^{N} W_i \exp(-\frac{\| \phi_{ik} - \theta_i \|}{2r^2}) \qquad \textbf{(3)}$$

where i is the number of links in a hair strand, N is the total number of links, k is the number of database sequences, and W_i is calculated by subtracting θ from ϕ. (The answer is an absolute value.) This calculation is carried out for the database sequence chosen in the previous step. We then repeat this step from the next hair root link to one point before the hair tip link, and adjust the database sequence. (Fig. 8)

To implement this interpolation, we can preserve the length of the hair segments. Once the angle differences at the first and last frames of a matching database sequence are computed, this interpolation is performed for all the frames within the sequence. It is repeated for all hair control points, thereby improving the strands, as shown in Fig. 7 (right). A second strong point of this process is that it preserves hair length and reduces the dimensionality of the calculations from two dimensions (for x and y) to one (the angle)

4.4 Interpolation Between Hair Motions

We interpolate between two consecutive hair sequences in order to connect the hair motion smoothly. Since we use impulse forces to construct the hair motion
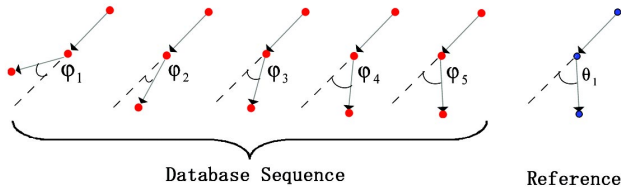
Figure 8. An image of how to implement deformation to the database sequence

Figure 9 An image of the transition between DB sequences

database, the animation result may not be smooth without interpolation. We use Eq. (4) for the interpolation.

$$ S = \frac{1}{2}\left(1 + \frac{(1+c)\times(t\times 2 - 1)}{\| t\times 2 - 1 \| + c} \right) \qquad (4)$$

Where c is a variable parameter. If c is large, this interpolation becomes close to linear interpolation. If it is small, the interpolation becomes close to a step function. t represents the time interval; the smaller t, the smoother the interpolation. The advantage of this equation is that animators can control c. They can decide whether the interpolation should change dramatically or smoothly. Too much smoothness and meticulousness, however, is inconsistent with cartoon-like animation because it makes animation results more realistic. The animator thus must decide on the best parameters to use. Figure 9 shows the image of this interpolation.

5 Results

Camera Position Interpolation

The camera position for each rough sketch is set to a key position, and we carry out linear interpolation between these positions. The number of segments depends on the number of frames obtained between the rough sketch images.

Creating hair thickness

Our hair model does not consider thickness because we obtain this data from the two-dimensional images. To do the shading, we create the thickness of the hair. To create the thickness, we use the hair boundary lines. We obtain the center position from the boundary points and then calculate the vector from the center to right-boundary control point. To get the direction from the hair root to the end, we use an average vector made from both boundary points. We then calculate the outer product of these vectors. By controlling the calculated vector's magnitude, we can create and control hair thickness.

Cartoon shading

We have implemented cartoon shading to obtain cartoon-like aspect character. [Adv02a]

Using our method, we have successfully animated our hair model in two animations using two kinds of inputs. For each animation, we have also used the 'on-twos' animation method, which is commonly used for cartoons and i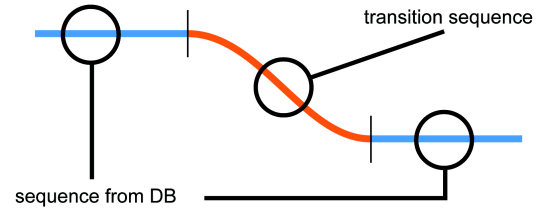s a means of getting more cartoon-like animation by using the same image for two frames. All animations were made at 24 frames per second, the standard rate for cartoon animation. Figure 10 shows the results using input that requires camera motion. It clearly shows the effectiveness of our method, since there are few animators who can achieve animation by hand. Figure 11 shows the results using input that does not require camera motion. This also shows that our method is clearly working for matching. The computations for these example results were executed on a computer with a 2.4 GHz Pentium 4 processor and were completed within five minutes for 120-frames animation.

6 Discussion

Using physical parameters obtained from a database of hair motions designed by a skilled animator, our method enables animators to use computer graphics to interactively design and generate cartoon hair animations with quality close to that obtained by hand. We easily produced target hair motions by using a simple hair model designed physically. Using our method we can achieve hair motions visualized from roughly sketched indicator images.

However, since the target cartoon expression is in the animator`s mind, the animation results could be considered to be the right expression by one animator but not by another animator. An extreme way of saying this is that a cartoon character has a life in all scenes. What the character does must have the meaning the animator wants to express. One can argue that this is why various methods of cartoon expression have been developed.

Allowing this inconsistency is a huge advantage of cartoon expression and the most difficult thing to express in computer graphics. Even if the appearance of a cartoon characters` hair is physically impossible, the animation can still be fantastic and express the animator's intention. This is an advantage of drawing by hand. Since our model uses a three-dimensional hair structure to render hair, it cannot deal with such inconsistencies. The question of how to address these inconsistencies is future work. Automating the pre-computing steps, sections 3.1 and 3.2 is also left for future work. Another future project is consideration of how shadowing, rendering, etc. should be handled to enable discrepancy in cartoon expression.

Acknowledgment

Reference

[Ken92a] Ken. Anjyo, Y. Usami, and T.Kurihara. A simple method for extracting the natural beauty of hair. In *Proc. of SIGGRAPH 92*, pp. 111 - 120, 1992.

[Dal93a] Daldegan, N.M.Thalmann, Kurihara, and D. Thalmann. An integrated system for modeling animationandhair renderin *Computer Graphics Forum (Eurographics 93)*, 12(3): pp. 211 - 221, 1993.

[Had00a] Hadap and N. MagnenatThalmann. Interactive hair styler based on fluid flow. In *Computer Animation and Simulation 2000. Proceedings of the Eleventh Eurographics Workshop*, 2000.

[Had 01a] Hadap and N. Magnenat-Thalmann. Modeling dynamic hair as continuum. In *Eurographics Proceedings. Computer Graphics Forum, Vol.20,No.3*, 2001.

[Yyu01a] Y.Yu. Modeling realistic virtual hairstyles. In *Proceedings of Pacific Graphics*, pp. 295-304, 2001.

[Joh02a] Johnny Chang, Jingyi Jin, and Yizhou Yu, A Practical Model for Hair Mutual Interactions, ACM SIGGRAPH Symposium on Computer Animation, San Antonio, July 2002, pp.73 - 80.

[Rer91a] R.E. Rosenblum, W.E. Carlson, and E. Tripp. Simulating the structure and dynamics of human hair: Modeling, rendering and animation. *The Journal of Visualization and Computer Animation*, pp.141-148, 1991.

[Rfe87a] R. Featherstone. *Robot Dynamics Algorithms*. Kluwer Academic Publishers, 1987.

[Mul01a] Multibod Dynamics (Package software) http://www.kuffner.org/james/software/index.html

[PLi94a] P. Litwinowicz, L. Williams. Animating images with drawings. SIGGRAPH 94, Orlando, FL, pp. 409-412,1994

[Nbu76a] N. Burtnyk, M. Wein: Interactive Skeleton Techniques for Enhancing Motion Dynamics in Key Frame Animation. SIGGRAPH 76 , Orlando 564 - 569.

[Pau99a] Paul Rademacher, "View-Dependent Geometry" In *Proceedings of SIGGRAPH 99*, L.A pp. 439-446

[Luc02a] Lucas Kovar, Michael Gleicher , and Fred Pighin, Motion Graph, In Proceedings of SIGGRAPH 2002 San Antonio. pp 473-482.

[Jpl00a] J. P. Lewis, Matt Cordner, Nickson Fong,"Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation" In Proceedings of SIGGRAPH 2000 New Orleans. pp 165 - 172.

[Adv02a] Advanced RenderMan, A.A.Apodaca and L. Gritz. Morgan Kaufmann 2002

[Dvd01a] DVD, Making process of "Princess of Mononoke" produced by Studio GHIBLI.

[Pno04a] P. Noble and W. Tang: Modelling and Animating Cartoon Hair with NURBS Surfaces, Proc. CG International 2004, pp. 60 - 67.

[Yba03a] Y. Bando, B.-Y. Chen and T. Nishita, Animating Hair with Loosely Connected Particles, Computer Graphics Forum, Vol. 22, No. 3, 2003.

[Kwa03a] K. Ward and M. Lin, "Adaptive Grouping and Subdivision for Simulating Hair Dynamics", Proceedings of Pacific Graphics, 2003, pp.234-243.

[Fbr03a] F. Bertails, T.-Y. Kim, M.-P. Cani, and U. Neumann, " Adaptive Wisp Tree - a multiresolution control structure for simulating dynamic clustering in hair motion", ACM Symposium on Computer Animation, 2003.

[Tae02a] Tae-Yong Kim and Ulrich Neumann "Interactive Multiresolution Hair Modeling and Editing", In Proceedings of SIGGRAPH 2002 San Antonio pp 620 - 629.

[Epl01a] E. Plante, M.-P. Cani, and P. Poulin. A layered wisps model for simulating interactions inside long hair. In *Proceedings of Eurographics Computer Animation and Simulation*, 2001.

[Jtc02a] J. T. Chang, J. Jin, and Y. Yu."A practical model for hair mutual interactions" *Proceedings of ACM SIGGRAPH Symposium on Computer Animation 2002*, 73–80, 2002.

[Jon87a] John Lasseter "Principle of Traditional Animation Applied to 3D Computer Animation" Proc Siggraph 1987, pp 35-44.

[Dou03a] Doug L. James and Kayvon Fatahalian, Precomputing Interactive Dynamic Deformable Scenes, In Proceedings of ACM SIGGRAPH, 2003.
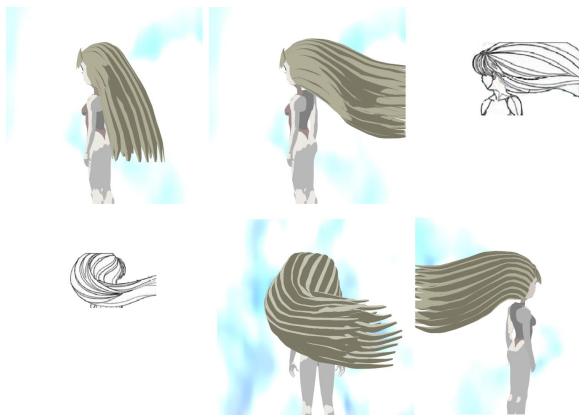
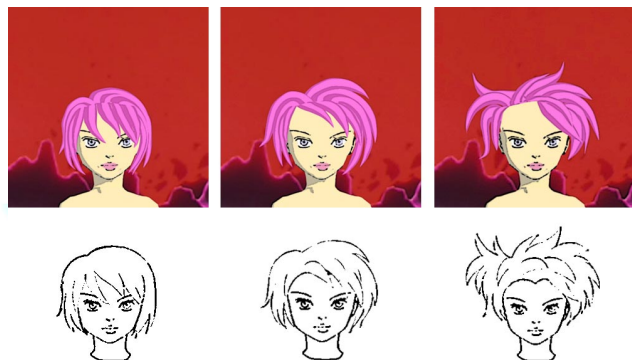**Figure 10 Animation sequence that requires camera motion**

Figure 11 Matching sample of another character from animation sequence