Felt-Based Rendering

Peter O'Donovan[*]
University of Saskatchewan

David Mould[†]
University of Saskatchewan

Abstract

Felt is mankind's oldest and simplest textile, composed of a pressed mass of fibers. Images can be formed directly in the fabric by arranging the fibers to represent the image before pressure is applied. We describe a computational method for transforming input images into objects which look as if they were produced by a felting process. The synthesis method places three dimensional line segments one by one, analogous to individual fibers being placed. Individual layers of fibers are drawn according to image structure and a probabilistic framework. A fuzzy three dimensional felt object is created by compositing layers of fibers; rendering uses a deep shadow map for correct self-shadowing of the matted felt.

CR Categories: I.3.0 [Computer Graphics]: General;

Keywords: felt, textile rendering

1 Introduction

One of the tasks of nonphotorealistic rendering is to produce images which have the same appearance as traditional media, such as oil paints [Meier 1996] or pen-and-ink [Winkenbach and Salesin 1994]. Although cloth has been used by artists through the ages, and modeling and photorealistic rendering of cloth art have been common in computer graphics, nonphotorealistic treatments of cloth have been lacking. In this paper, we present a method for transforming an input image into an image done in a felted style.

Felt is mankind's oldest textile, and felt art, such as rugs and tapestries, dates back thousands of years [Gordon 1980; Evers 1987]. While plain felt can be made by heating and pressing a mass of woolen fibers, art objects can be made by properly arranging different types of fibers before pressing them. The felt style is characterized by the underlying tangle of differentiated fibers, which gives rise to a fine-scale fuzzy surface texture and to a blending of colors across boundaries in a felted image.

In this paper, we are inspired by real-world feltmaking processes to produce images which look as if they are made of felt. Plain, flat felt can be hand-constructed using the following process, where we follow the description given by Gordon [Gordon 1980]. First, the fibers of wool are separated by being combed, or carded to spread the fibers and orient them into a single direction. Next, pieces of wool are placed in layers on top of some sort of substrate. Typically, each layer is rotated in the plane so that the average fiber direction is perpendicular to that of the previous layer. The wool is secured in place and kneaded, hammered, or pounded until the individual fibers have hardened into a single cohesive sheet of felt.

The above abbreviated description gives us the basics of the felt-making process, which – done with raw wool, or wool of a single color – can produce textures, but not meaningful images. Images can be placed in the felt by directly printing on or dyeing the finished felt. However, we are interested in so-called "inlay" of colors, where an image is constructed by arranging fibers or pieces of differently colored wool. Arranging wool fibers in a structured image and then pressing the resulting arrangement produces a piece of felt containing the desired image.

Felt artists have tremendous leeway in arranging the image. Unlike woven fabrics, felt places no constraints on the orientation of the constituent fibers. While the felt process often involves placing fibers in alternating perpendicular layers, this aspect of the process is done to ensure strength and durability in the final felt, and hence may not be compulsory in the case of a piece of felted art. As we shall show in later examples of real felt images, the directionality of the fibers is often lost during the matting process, leaving a tangle of connected fibers which results in the fuzzy appearance of the felt piece. In our algorithm, we automate the process of placing wool, guided by an input image.

The paper is organized as follows. First, we discuss previous work. Second, we describe our algorithm for creating synthetic felt images. Third, we show our results, chiefly consisting of images synthesized by our method. Finally, we close and give pointers ahead to future research directions.

2 Previous Work

Photorealistic renditions of cloth have been undertaken by numerous authors, including Sattler et al. [Sattler et al. 2003], Xu et al. [Xu et al. 2001], and many others. However, to our knowledge, little work has been done on NPR in the context of textiles. The batik modeling by Wyvill et al. [Wyvill et al. 2004] is an exception, treating the appearance of cracks in the dyeing process of batik painting.

Animation and modeling of cloth has had a long history in computer graphics, continuing to recent work such as that of Baraff, Witkin, and Kass [Baraff et al. 2003]. However, the patterns and images appearing on the cloth are outside the scope of such models; in contrast, this paper focuses on synthesizing patterns that appear on the model.

Self-shadowing is necessary for realistic rendering of semi-transparent objects like clouds, fur, or hair. Shadow maps are one technique used to properly cast self-shadows for these volumetric objects. Lokovic and Veach [Lokovic and Veach 2000] extended traditional shadow maps to deep shadow maps whereby each pixel stores a transmittance or visibility function, rather than a single depth value. This function represents the amount of light penetrating to each depth and allows proper self-shadowing of semi-transparent objects.

Bertails et al. [Bertails et al. 2005] further extended this approach to create a self-shadowing algorithm for interactive hair animation.

[*] e-mail: peo499@mail.usask.ca

[†] e-mail:mould@cs.usask.ca

They proposed a 3D light-orientated shadow map composed of a uniform cubic voxel grid with both density and transmittance values for each voxel. Each vertex is projected into the light's view direction and increments the density value of the containing voxel or cell. Once all the density values have been computed, light rays are cast from the light through the voxel grid to find the transmittance or visibility of each cell. Light rays are assumed to be parallel, so that calculating the transmittance value simply involves iterating through a row of the voxel grid. The equation for the transmittance of cell (i, j, k) is given by:

$$T_{i,j,k} = \sum_{m=i_{min}}^{i} exp(-d_{m,j,k} f ds) \tag{1}$$

where $d_{i,j,k}$ is the density of cell (i, j, k), f is a scaling factor, ds is the cell width and i_{min} is the index of the map slice closest to the light. Their algorithm allows an efficient computation of self-shadowing for a large number of transparent fibers and is thus well suited for realistic rendering of felt objects.

In our work, we concentrate on creating realistic three dimensional models of felt inlaid with colours from an input image. How can we use an image to create a model that looks as if it had been constructed of felt? For clues about how to answer this question, we looked to both the real-world felting process and to previous work in NPR. Although felting has not been previously treated in computer graphics, our solution has much in common with painterly rendering, which has seen a great deal of attention. Early painterly rendering systems such as those of Meier [Meier 1996] and Hertzmann [Hertzmann 1998] placed individual strokes of paint on an initially empty canvas; this process is analogous to the process described earlier of laying down fibers of wool one by one to build up a piece of felt. In fact, felt artists sometimes refer to the process as "felt painting" [Gordon 1980].

One of the differences between our automated felt painting and other painterly rendering work is in the number of strokes used. Our felt paintings require potentially dozens of strokes per image pixel. In many automated painterly systems, this ratio is reversed. A large number of fibers are required in order for our algorithm to produce a detailed texture resembling felt.

3 Algorithm

The fuzzy, textured surface of felt offers an intriguing modeling challenge, similar to those experienced in the areas of fur and hair simulation. To simulate this texture, our computational process imitates the real-world felting process. Fibers are drawn in three dimensions according to the image structure and random variations.

The basic primitive for our felt model is a strand of wool. Individual strands of wool are quite thin, and like other fibers, are partly transparent. Strands are modeled in this work using short transparent piecewise linear curves. While strands maintain a displacement vector, denoted by $D_{x,y,z}$, that controls their overall direction, segments of the fiber (the individual linear curves) can vary probabilistically from this direction. The direction of strand segment i, denoted by $D^i_{x,y,z}$, is given by a normal random variable with a mean of the overall strand direction and a variance parameter σ.

$$D^i_{x,y,z} = (N(D_x, \sigma), N(D_y, \sigma), N(D_z, \sigma)) \tag{2}$$

where $D^i_{x,y,z}$ is the direction of strand segment i, D_x, D_y, and D_z are the x, y, z components of the strand's overall direction, and $N(\mu, \sigma)$ is a normal random variable with mean μ and standard deviation σ. Therefore, the segments of the strand display an overall orientation but also a user-controlled probabilistic deviation. As shown in Section 4, with an increased variance the image loses definition and becomes fuzzier. The variance parameter also varies the segment lengths, a desirable property since real-world fibers vary in length.

Strands are combined together to form layers of fibers. A layer is created by drawing a single strand from each pixel of the image. Layers are composited by placing each layer a small distance above the previous one. Strands are colored by their originating pixel and remain the same over all the strand segments. No intersection tests are performed on the strands, nor are density tests to prevent clumps of fibers; each strand is drawn independently of all other strands.

In the simplest case, the overall directions for the strands are selected from a uniform random variable and normalized. We control the direction of the strands by weighting the z component less than the x and y components:

$$D_{x,y,z} = (\alpha U, \alpha U, \beta U) \tag{3}$$

where α is the weighting component for the x and y dimension, β is the weighting component in the z dimension, and U is a uniform random variable between -1 and 1.

The weighting allows us to orient the strands approximately along a plane. In the real world felting process, hunks of wool are carded before they are placed on the felt object, so while each layer of wool is three dimensional, it is relatively thin and nearly planar. Fig 1 shows the effect of using these undirected layers. On top, we have a single layer of fibers drawn from the common mandrill test image. After compositing a number of undirected layers, the result is the image shown in the bottom of Fig 1.

Homogenous fuzziness does not characterize all felt work. Felt artists have the ability to orient strands of wool and place them in sufficient density, either by layering or using larger, half-spun pieces of wool, to produce areas of sharper definition where fibres of varying colour have been used. For example, in Fig 12 the purple wool in the top third of the piece has not blended significantly with the surrounding pinkish regions leaving a distinctive border. However, the agitation stage of the felting process causes fibers to shift; even distinctive borders will often show some visible color spill when examined closely.

To model this phenomenon we introduce a directed layer of felt. A directed layer is identical to a regular felt layer except for its overall strand direction. Edge structure in the image is used to influence the direction of strands. First, the edge magnitude and direction are measured for each pixel in the image using the Sobel operator. The strand direction is the edge direction rotated 90^o to lie perpendicular to the gradient, multiplied by the edge magnitude and added to a scalable uniform random variable. The equation is given by

$$D_{x,y,z} = \quad (G_x|G| + \alpha U, G_y|G| + \alpha U, \beta U) \tag{4}$$

where (G_x, G_y) is the 90^o rotation of the x and y gradients, $|G|$ is the gradient magnitude, (α, β) are weighting components, and U a uniform random variable between -1 and 1.

The strand direction is now controlled by the edge magnitude. If the magnitude is low, the random variable will dominate and the resulting strand will be undirected. As the edge magnitude increases, the strand will orient itself along the edge. Finally, since edge information is only given in two dimensions, a uniform random variable gives a small displacement in the z direction. This displacement vector is normalized so the strand length is independent of the edge magnitude. Fig 2 shows a directed layer from the mandrill test image. Section 4 contains numerous examples of images composited using directed layers. Note that the strands of both Fig 1 and 2

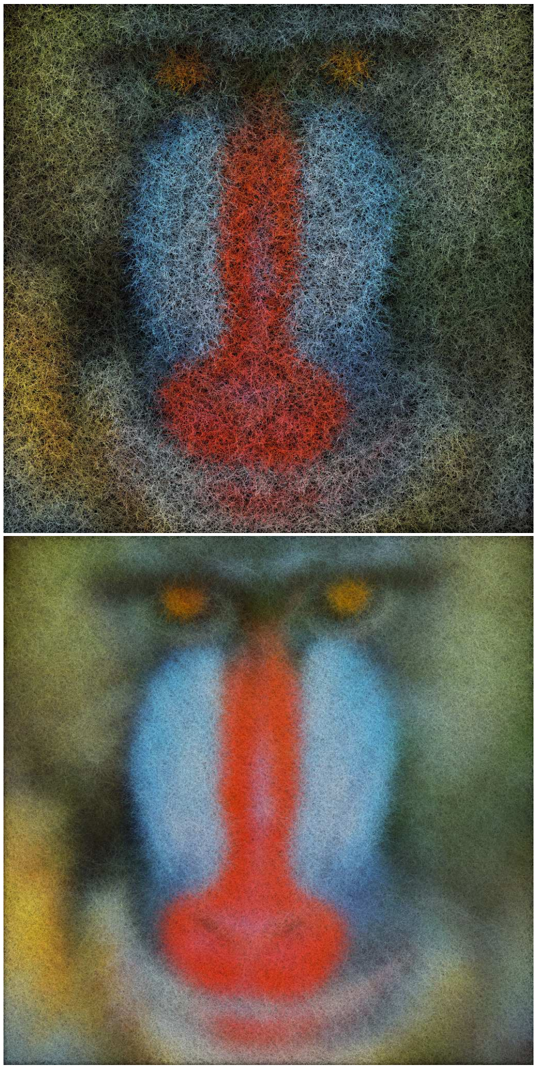

Figure 1: Undirected mandrill felt images. Top, a single undirected layer of felt; bottom, nine layers composited.

have a higher transparency (0.5) than the images with multiple layers (where the transparency is 0.1). The higher transparency is necessary to view a single layer with its low number of strands. Color spill, as shown in Fig 8, still occurs with directed layers when longer strands from inside a region, and therefore with no directional bias, pass over an edge. However, this spilling is not critical as the orientated strands emphasize the edge and a little spill is both acceptable and similar to real felt borders.

While input images can be any size, the algorithm can also upscale the resulting felt image. A random offset is added to the start position of each strand to fill in empty areas. The algorithm combines multiple layers with different characteristics to create the final image. Lower layers of the felt object are undirected, creating a fuzzy base which covers the image plane. A few upper directed layers are drawn to enhance edge information. For our work, nine undirected layers were drawn followed by three directed layers; input images were 256×256 and the resulting felt images were 1024×1024. In Fig 3 we also show the felt object from different angles to better view its dimensions.

The illumination of felt poses a challenge. The high number of thin, translucent fibers produce a self-shadowing effect that is vital to reproduce for realistic rendering. As mentioned earlier, the efficiency of Bertails et al.'s algorithm [Bertails et al. 2005] for rendering hair makes it ideal for calculating the self-shadowing of the numerous fibers of a felt model. Initially, a voxel grid is created to encompass the felt model. The voxel grid is orientated to lie along the light's view direction, allowing an efficient calculation of the transmittance function. Tracing a light ray from the light source involves simply iterating through a row of the grid.

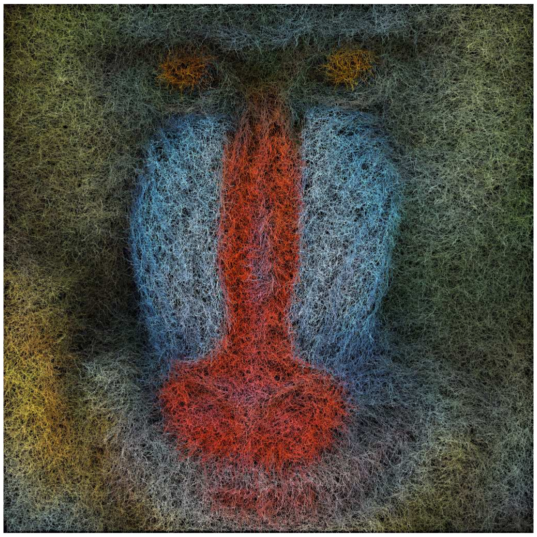

Figure 2: A directed layer of felt

The first step of the illumination algorithm fills the shadow map with fiber densities. Each fiber is projected into the light's view direction and the associated cell density incremented. Secondly, the transmittance values are calculated by tracing light rays through the grid along the light direction. The transmittance for each cell is calculated according to Equation 1. Lastly, transmittance values are filtered using trilinear interpolation to remove any patterns aligned with the shadow map.

Once the transmittance values are calculated, we apply a lighting calcuation to each vertex during rendering. Unlike other fiber objects like hair, specular highlights are not common in felt work. The common fiber in felt, wool, has a matte appearance. Therefore, the lighting model for felt is composed simply of an ambient and diffuse component. The strand color is sampled from the underlying image near the strand's originating pixel and maintains the same color for all strand vertices. The color ϕ_P of vertex P is given by:

$$\phi_P = \phi_{strand} \times \phi_{Ambient} + \phi_{strand} \times Trans(P) \times \phi_{Diffuse} \quad (5)$$

where ϕ_{strand} is the strand color, $\phi_{Ambient}$ and $\phi_{Diffuse}$ are the colors of the light sources, and $Trans(P)$ is the transmittance from the light source to vertex P. However, real felt objects can be composed of fibers, or blends of fibers, other than wool. Future work may include a better analysis of fiber properties to allow realistic rendering of a variety of fiber types such as silk or mohair.

In Fig 14 we compare the algorithm with self-shadowing lighting disabled and enabled. We feel the deep shadow map produces a more cohesive, realistic looking surface texture than the simple texture produced without self-shadowing. The surface texture without self-shadowing is signficantly more blurred and lacks the uniform fibrous surface texture apparent on the self-shadowed image.

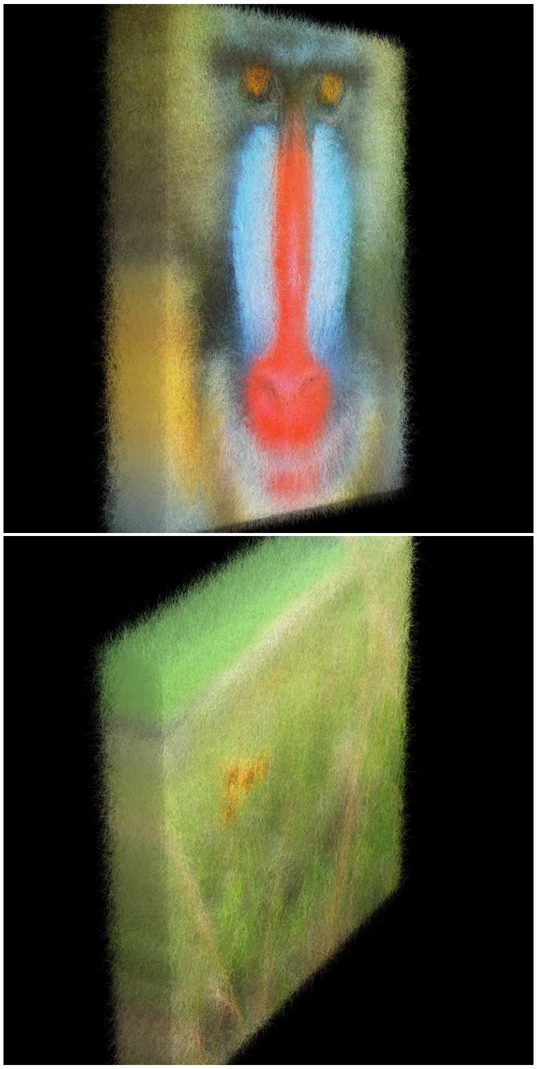

Figure 3: Different views of the mandrill and field flower felt pieces

4 Results and Discussion

Here we show a collection of images generated by our algorithm, as well as the effects of changing various parameters. We also compare our synthetic images with work produced by professional felt artists. For all images presented in this section, 12 layers were used to produce the felt model. The lower nine layers are undirected to allow a base that covers the image plane. The upper three layers are directed to enhance edge information. Since a strand is drawn for each pixel of the image, twelve layers for a 256×256 image will produce 786432 strands to display.

We begin by showing the effect of modifying the strand length on the resulting image. We control the strand length by changing the number of segments for each strand. Fig 4 shows the original fence image along with felted images with segment counts of 10 and 30. The longer strand length results in a less defined image. By using a small variance parameter with a longer strand, the primitives become straighter and increasingly apparent.

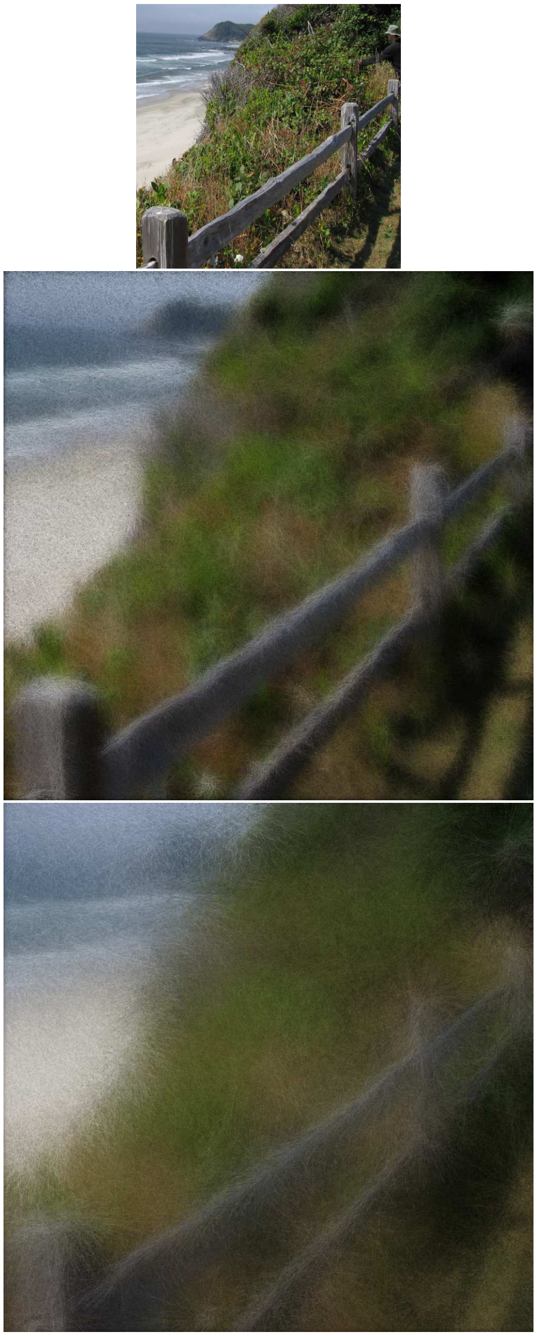

Figure 4: The effect of changing the strand segment count. Top, the original image; middle, segment count of 10; bottom, segment count of 30.

As we mentioned earlier, the variance parameter can be used to control the fuzziness of the resulting felt object. A larger variance will create more random movement of strand segments, blurring the boundaries. Fig 5 shows the effect of changing the variance parameter on the mandrill test image. We must also modify the transparency as the higher variance creates longer strands, resulting in a brighter image. To control for this effect, we have reduced the transparency as the variance is increased, allowing a better comparison. As the variance increases, the resulting images become fuzzier. The final image bears a strong resemblance to the undirected composite of Fig 1, which is not surprising as edge direction becomes increasing irrelevant with a higher variance. The importance of the variance parameter lies in the length of the fiber strands. If longer strands are used with a low variance, as in Fig 4, these long straight

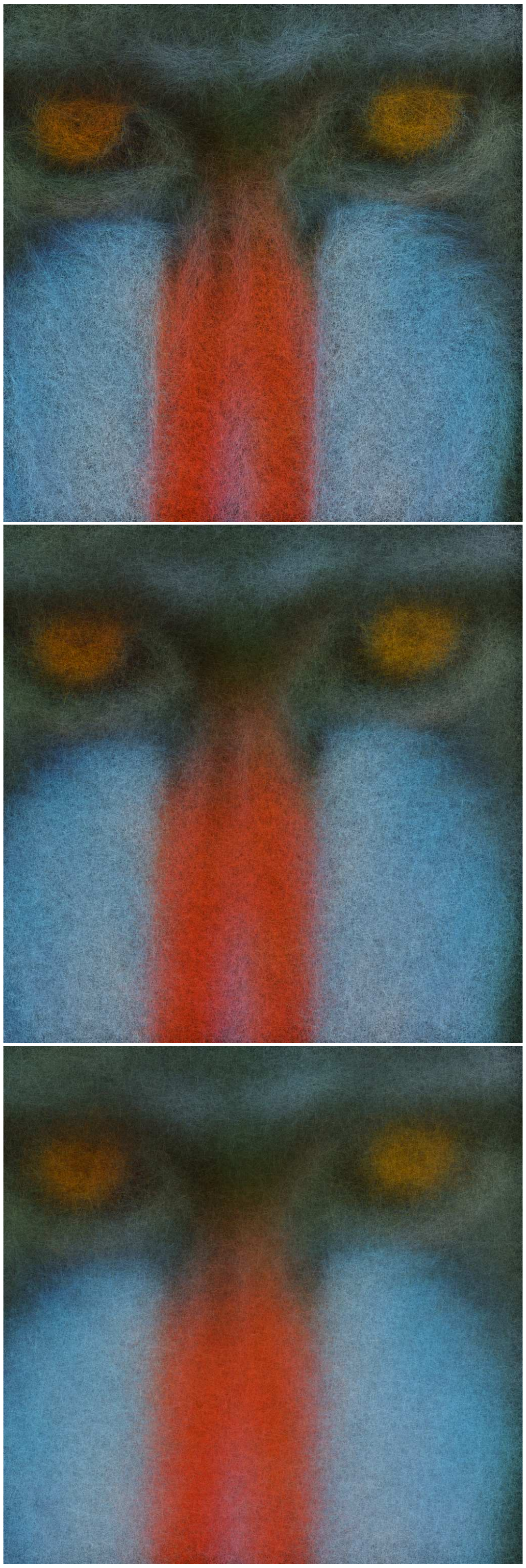

Figure 5: The effect of changing the variance and transparency parameters. Top, variance = 0,transparency = 0.1; middle, variance = 1, transparency = 0.07; bottom, variance = 2, transparency = 0.05.

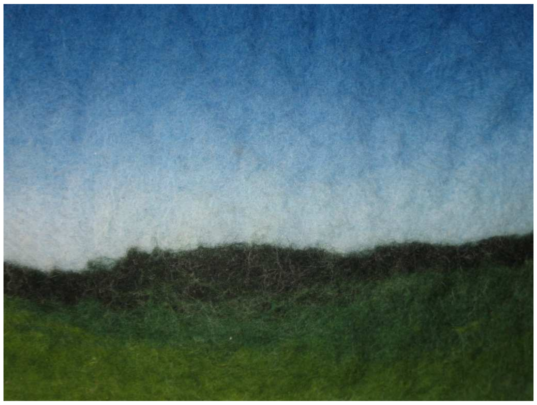

Figure 6: Felt piece by artist Pat Adams

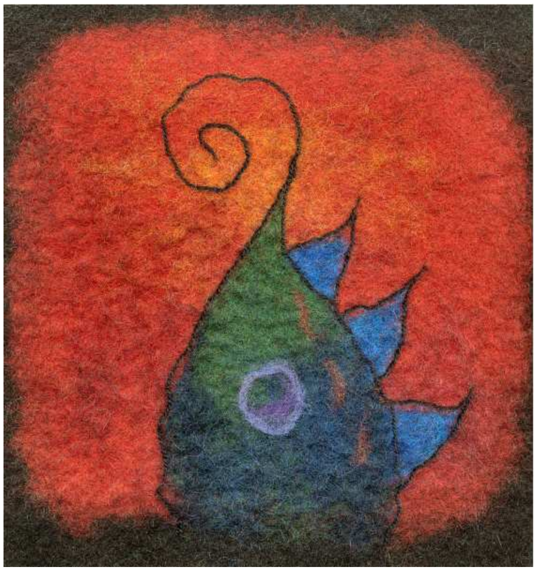

Figure 7: Felt piece by artist Karoliina Arvilommi

strands become increasingly noticeable. If a longer strand is desired, the variance parameter is necessary for visually pleasing results. For the remaining images, a small variance of 0.2 is used to provide a small deviation from the overall strand direction.

Figs. 6, 7, 12 and 13 show felt pieces created by professional felt artists. Fig. 12 by Myrna Harris [1] suggests some of the strengths of the felt medium. The relatively featureless sky can be enriched with colourful felted regions. Note that the lower half of the image is created by embroidered fabric rather than felting. Carefully chosen details such as the tree in the foreground were sewn in after the felt was made. In Fig 6 by Pat Adams, the artist has used large regions of gradually blended colors to depict a landscape. The randomized, fuzzy texture of felt increases the detail of these regions and creates a more attractive image. While the abstract regions of Fig 7 by Karoliina Arvilommi [2] are primarily homogenous, the artist has added wool of various colors to increase complexity in the felt texture. While the underlying textures remain similar, longer surface strands are more visible in Fig 7 than in the other felt pieces. These differences are due by different wools or techniques used by the different artists. The black outline is yarn attached to the felt

[1] http://myrhar.sasktelwebsite.net - used by permission

[2] http://www.4felts.com - used by permission

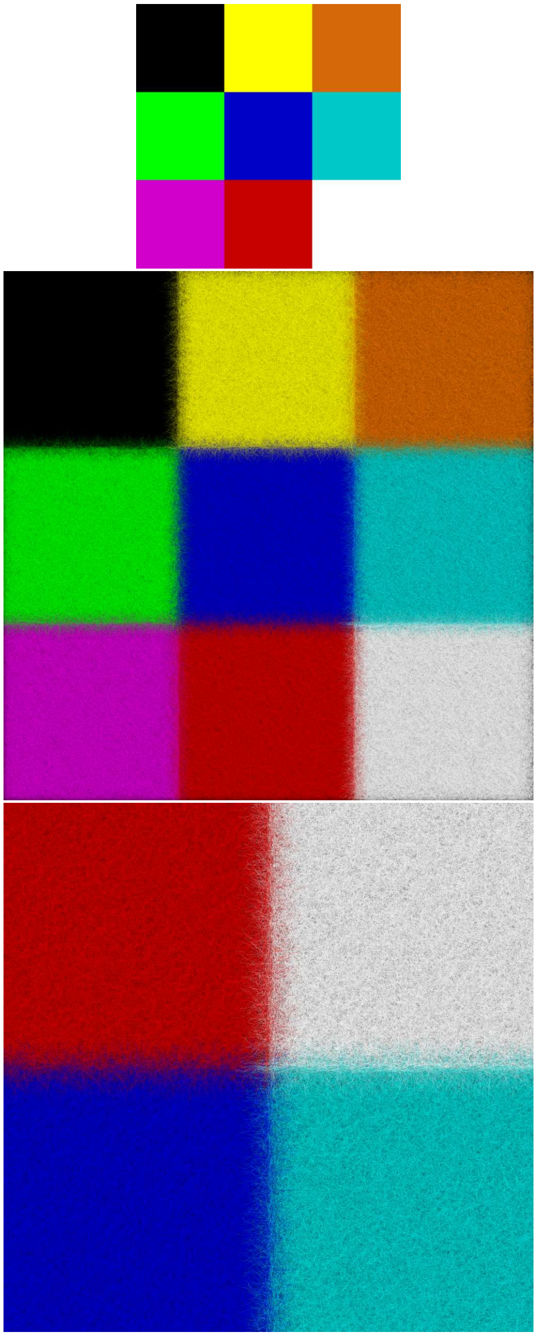

Figure 8: Top, original checkerboard; middle, the felted checkerboard; bottom, closeup of lower right corner.

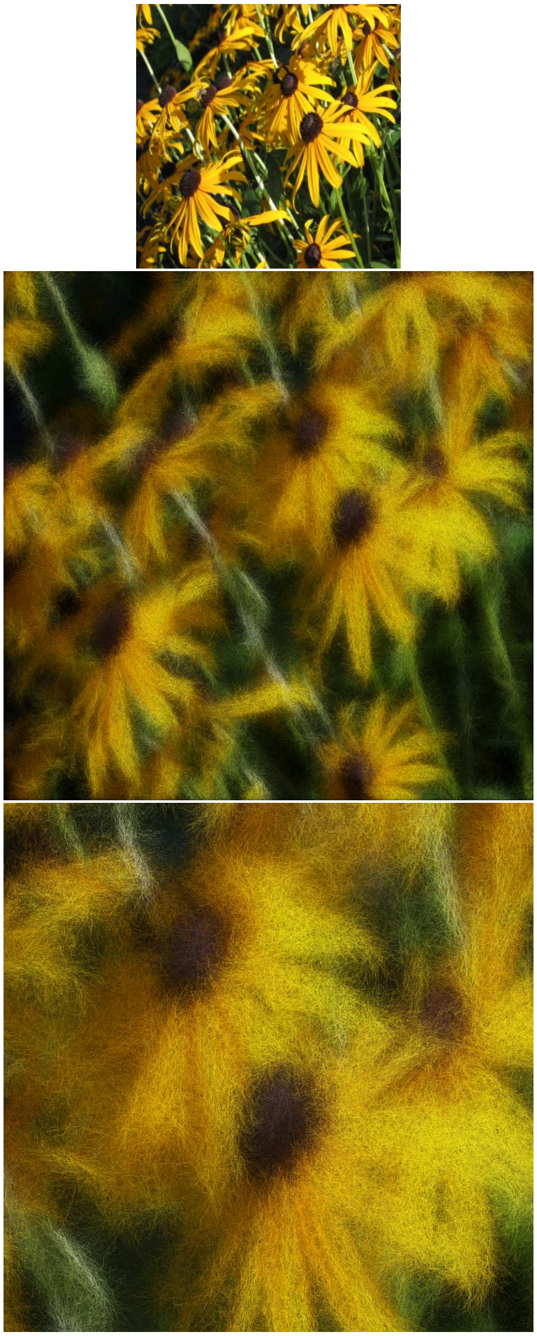

Figure 9: Top, the original orange flowers image; middle, the felted image; bottom, closeup detail of petals from the orange image.

piece. While such post felting processes add detail, they are not part to the felting process. We have chosen not to simulate such processes as they suit a more interactive felt creation program than our data driven approach.

Our synthetic felt process produces a textured image with several important characteristics. First, individual strands are not noticeable unless in high contrast areas or the image is scaled to a higher resolution. The thin and transparent individual strands are less important than the combination of strands. It is the tangle, the interwoven structure, that is perceptible and produces a fibrous, randomized surface texture. This fuzzy surface texture is the most important characteristic of felt and a primary goal for our algorithm. Fig. 13 shows a comparison between the details of real felt and our synthetic felt model. While there remains room for improvement in the algorithm, the resulting synthetic texture does resemble the undirected, chaotic texture of real felt work. However, the texture exists at quite a fine scale. As in real felt work, as the viewing distance increases or the image is scaled to a lower resolution, the fine texture of the strands is lost and the image becomes blurred.

We next examine the effect of applying the algorithm to images with a variety of region sizes and homogeneities. In Fig 8, we test the algorithm on a synthetic checkerboard of various colors. The

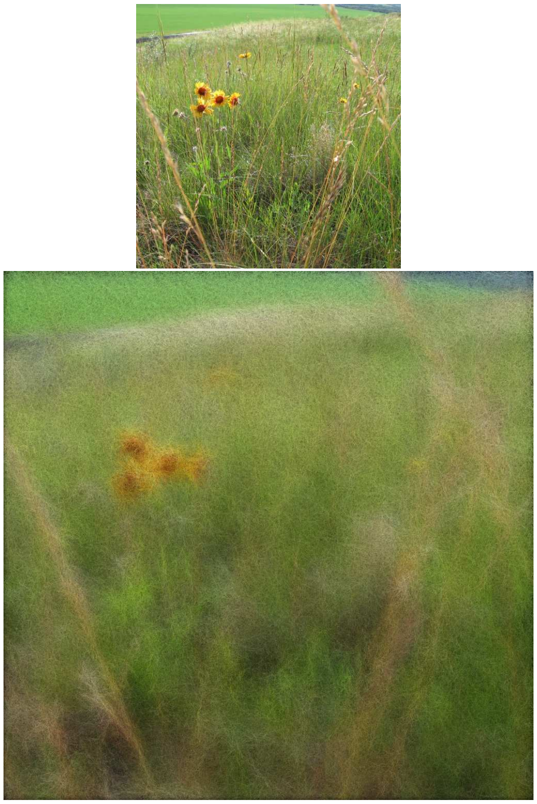

Figure 10: Top, the field flower image; bottom, the felted image.

synthetic image gives a good test of the algorithm's effect on homogenous regions, allowing us to view the felt texture independent of the visual complexities of high frequency colors. Fig 8 shows that the algorithm produces the fibrous felt texture without relying on high frequency content from the image. Also, while the borders between colors contain orientated strands, there is a small amount of crossover that produces a color spill.

In Fig 9 we apply the algorithm to an image with smaller, relatively homogenous regions. While the boundaries of the petals and stems are preserved, some lines do deviate from the boundary and produce a fuzzy appearance. Within the petals, the randomized felt texture is apparent.

In Fig 10 we see the effect of the algorithm on a natural image with a great deal of high-frequency detail in the stalks of plants and blades of grass. In the resulting felt image, most of the fine local detail has been lost. However, we should not expect that details of this kind should be preserved; the felting medium does not well lend itself to representing arbitrary fine-scale structure. The resulting image is also significantly flatter and more cohesive; even though the regions are seperate, the impression is of a single surface with changing colours. This cohesiveness is another important characteristic of felt work. By having a single, fine-scale texture over the entire image, the regions are related, creating the appearance of a single surface with an inlaid image. In this image the regions of the image have blended together to form a homogenous surface texture that maintains the overall appearance of the image in a rather impressionist style. In Fig 15 we show a closeup of the field flower image to better demonstrate the felt texture.

The felting process appears to remove some of the subtle depth cues of images. The distant field in the upper part of Fig 10 is unrecognizable as such in the felted image. This flattening is caused by the removal of the high frequency details from the image. Whereas in the real image, the distinction between the high frequency of the foreground and the low frequency of the background is noticeable, in the felt image, the two regions have a similar texture.

Another issue is the complexity of color in felt pieces. Due to the transparency of wool strands and the effects of color dithering, felt artists are able to produce a vast array of colors by layering with even a small number of colors. Currently our algorithm repeatedly samples from the underlying image to determine strand colors for different layers. The effect is that homogenous regions like in Fig 8 produce a fuzzy felt texture without great color variation. While felters can and do produce regions of flat color, the color complexity possible in felt is one of its attractive features. This complexity is approached in our algorithm when the image contains a number of different colors in a small region as in Fig. 5 or 10. With a higher frequency of color, dithering and blending over the layers increase the color variation and similarity to real felt work while also maintaining the same fibrous texture as in homogenous regions. Future work may include less reliance on the image for color complexity by modifying the strand colors for different layers to increase the complexity of homogenous regions.

Our felting algorithm does contain several limitations. The algorithm does not currently allow a tremendous variation in the resulting image. There is little effect if we add numerous more strands; by varying parameters, we control only strand length and fuzziness. These parameter changes are only variations of a single, uniform felting style; they do not allow the wide variety of textures possible with felt. The styles of felt work can be radically different; everything from Fig 6 to Fig 12, to dense felt hats, to long, loosely matted decorative pieces are possible. The artist has tremendous control in choosing, orientating, and pressing the wool fibers. In contrast, our approach is mostly data-driven with little input from the user aside from modifying initial conditions. Unlike real felt working, there is no control for modifying parameters for different regions. Lastly, artists are free to manipulate the felt object once the felting process is complete. They can embroider the work, add other fabrics, re-dye, or perform other post-felting operations. Our process does not consider any of these further steps.

4.1 Timing

The felting process is moderately expensive, owing chiefly to the rendering time for the large number of strands in a felt object. The felting process takes approximately 40s to run on a 2.4 GHz P4 with input images of size 256×256 and resulting felt images of size 1024×1024. Of this, modelling is approximately 10s, calculating the deep shadow maps another 10s and rendering the remaining 20s. This expense could be improved upon by modifying the algorithm to draw the strands away from dense felt regions, resulting in a smaller number of strands needed to cover the image plane.

5 Conclusions

We have presented an automated method for transforming a given image into a felted version of the image. Rather than synthesizing two-dimensional images, as has been commonplace in NPR, our work creates a three dimensional felt object. Strands of wool are drawn in three dimensions and organized into layers of fibers. We employ image structure and a user-controlled variance parameter to control the fuzziness of our felt objects. We also use deep shadow maps to account for the complex self-shadowing of the large number of transparent wool strands.

The resulting objects resemble real felted artworks, having visually similar surface detail and depicting the artistic composition of the input image. We suggest that the method might be employed by real felt artists as a way of previewing a project before actually starting the crafting. There is also scope for future refinements of the procedure on this topic, some possibilities for which we describe below.

5.1 Future Work

Future work may include exploring the effect of using fibers for other NPR styles. In Fig 11 we show the effect of producing longer strands which grow up out of the image. The resulting texture bears a resemblance to moss or fur.

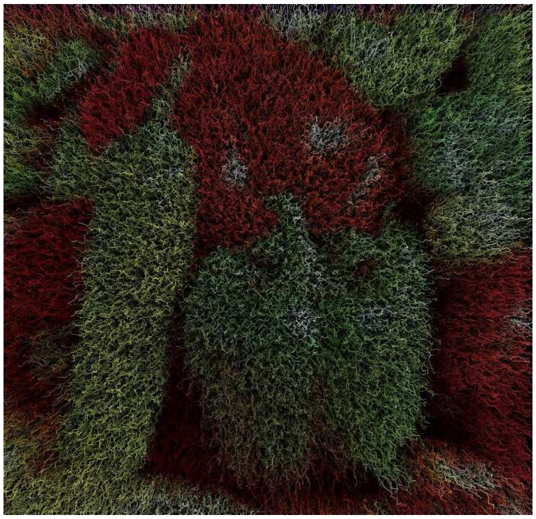

Figure 11: Mossy image of the standard peppers test image

More complex models are needed to better synthesize felt objects. Modelling larger pieces of wool may result in more realistic felt objects. Fiber density and intersections are also important aspects of the interwoven nature of textiles. Adding in further interwoven structure may also alleviate computational demands as fewer strands will be necessary to cover the image plane. Some artists also use felt to create 3D felt sculptutures. An intriguing modelling challenge would be to create a felt sculpture from an abitrary model. A more complex self-shadowing algorithm may also be implemented to allow more sophisticated lighting effects such as spotlights. Deviation from the colors of the underlying image for different layers should allow a more attractive blend of color where the homogenity of the image is not desired in the resulting felt object.

As mentioned earlier, another avenue of further research may explore the effects of using various fibers. Some felt artists use fibers, or blends of fibers, other than wool to create pieces with different surface properties. One of the limitations real felt artists must consider is the matting abilities of certain fibers. Silk, for instance, does not lend itself well to felt pieces. In simulation however, these properties are irrelevant and complex rendering exploring these visual fiber properties may allow additional effects such as shinier felt objects. Artists may be able to explore the visualization of felting diverse fibers to inspire novel real-world felting techniques.

References

Baraff, D., Witkin, A., and Kass, M. 2003. Untangling cloth. *ACM Trans. Graph. 22*, 3, 862–870.

Bertails, F., Ménier, C., and Cani, M.-P. 2005. A practical self-shadowing algorithm for interactive hair animations. In *Graphics Interface*, 71–78.

Christoudias, C., Georgescu, B., and Meer, P. 2002. Synergism in low-level vision. *Internat'l Conference on Pattern Recognition 4*, 16 (August), 150–155.

Comanicu, D., and Meer, P. 2002. Mean shift: a robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Machine Intell. 24*, 4 (May), 603–619.

DeCarlo, D., and Santella, A. 2002. Stylization and abstraction of photographs. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, ACM Press, 769–776.

Evers, I. 1987. *Feltmaking: Techniques and Projects*. Lark Books, Asheville, NC, USA.

Gordon, B. 1980. *Feltmaking*. Watson-Guptill Publications, New York, NY, USA.

Hertzmann, A. 1998. Painterly rendering with curved brush strokes of multiple sizes. *Proceedings of SIGGRAPH 98* (July), 453–460.

Lokovic, T., and Veach, E. 2000. Deep shadow maps. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 385–392.

Meer, P., and Georgescu, B. 2001. Edge detection with embedded confidence. *IEEE Trans. Pattern Anal. Machine Intell. 23*, 12 (December), 1351–1365.

Meier, B. J. 1996. Painterly rendering for animation. In *SIGGRAPH 96 Conference Proceedings*, ACM Press, 477–484.

Sattler, M., Sarlette, R., and Klein, R. 2003. Efficient and realistic visualization of cloth. In *EGRW '03: Proceedings of the 14th Eurographics workshop on Rendering*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 167–177.

Sonka, M., Hlavac, V., and Boyle, R. 1999. *Image Processing, Analysis, and Machine Vision*. Brooks/Cole Publishing, Pacific Grove, USA.

Winkenbach, G., and Salesin, D. 1994. Computer-generated pen-and-ink illustration. In *Computer Graphics (SIGGRAPH '94 Proceedings)*, vol. 28, 163–170.

Wyvill, B., van Overveld, K., and Carpendale, S. 2004. Rendering cracks in batik. In *Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering*, ACM Press, 61–69.

Xu, Y.-Q., Chen, Y., Lin, S., Zhong, H., Wu, E., Guo, B., and Shum, H.-Y. 2001. Photorealistic rendering of knitwear using the lumislice. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, 391–398.

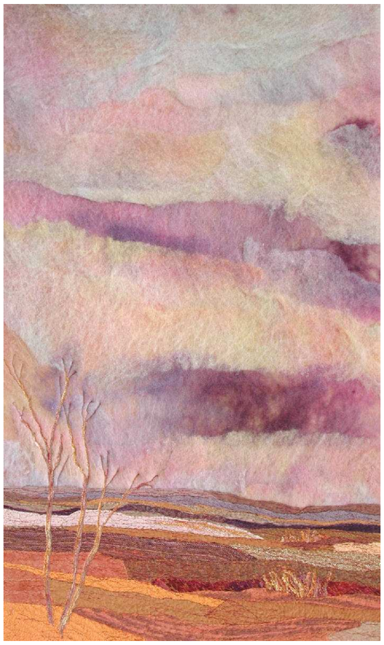

Figure 12: *Sunrise II* by artist Myrna Harris

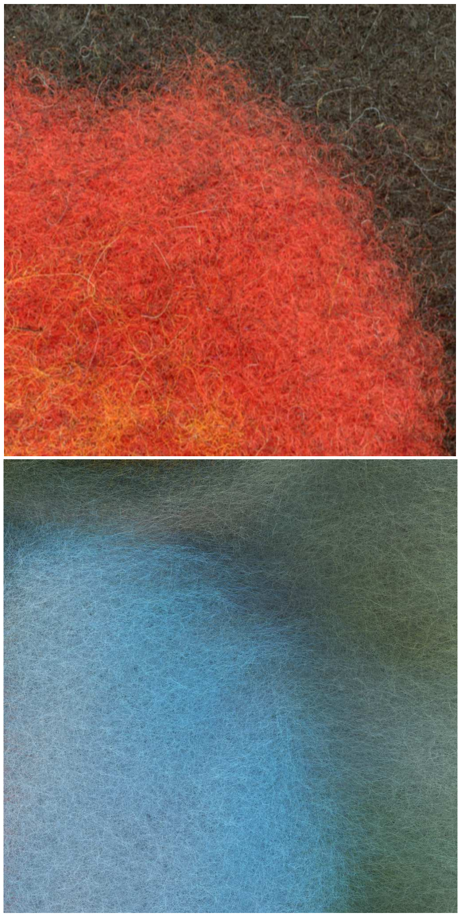

Figure 13: Top, detail from Fig 7; bottom, detail from the synthetic felt mandrill

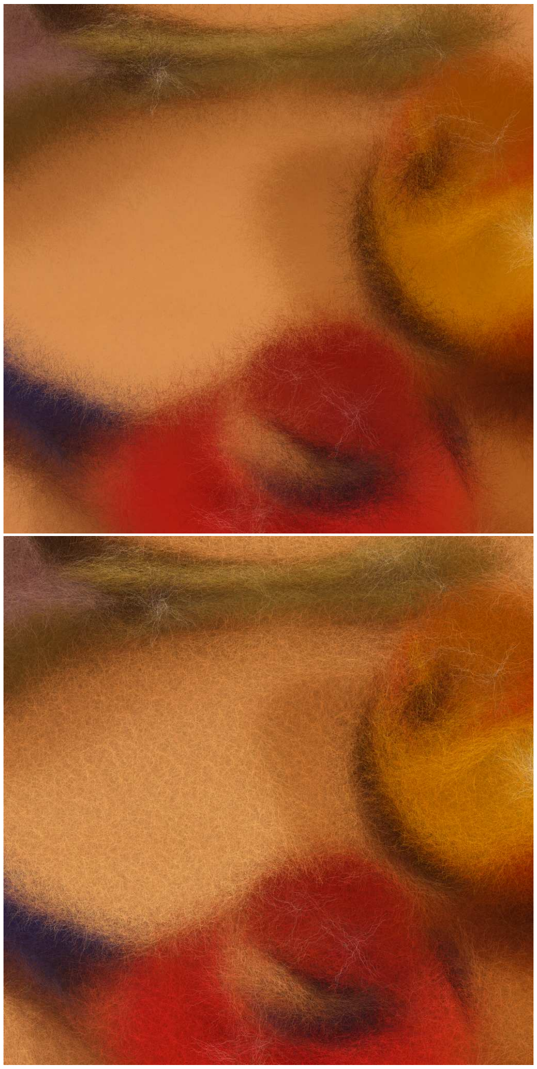

Figure 14: Glass bird images. Top, closeup without self-shadowed lighting; bottom, closeup with self-shadowed lighting using deep shadow maps.

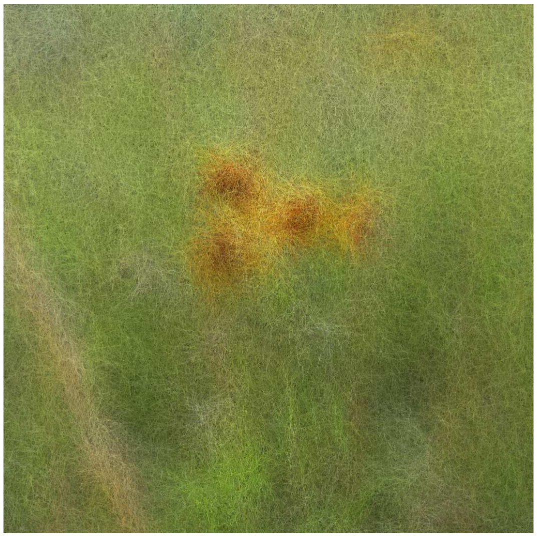

Figure 15: Closeup of field flower image