Report
on the
DeMaMech EU-Japan Exchange Program
2004/2005

Tina Wilczynski

1. Personal Data and University Information

1.1 Personal Data

Name: Tina Wilczynski
E-Mail: tinawilczynski@yahoo.de

1.2 Home Institute Information

Faculty: Mechanical Engineering and Transport
Department: Aeronautics and Astronautics
Institute: Institut für Luft - und Raumfahrt
Marchstraße 12-14
D-10587 Berlin

Supervisor: Prof. L. Blessing

1.3 Host Institute Information

Faculty: Science and Technology
Department: System Design Engineering
Lab: Lab. for Digital Design and Digital Manufacturing
Hideki Aoyama Laboratori

Institute: Keio University
Faculty of Science and Technology
Department of System Design Engineering
Hideki Aoyama Laboratory
3-14-1, Hiyoshi, Kohoku-Ku
Yokohama 223-8522, Kanagawa
Japan

Supervisor: Prof. Hideki Aoyama

2. Executive summary

In the year 2004 an exchange program between Europe and Japan is started. By this program I was exchange from the Technical University of Berlin (Germany) to Keio University in Japan, Yokohama. I spend 5 month in the laboratory of Prof. Hideki Aoyama.

The exchange program started with a two-week preparation course in September at the Technical University of Delft in Netherlands. During this weeks we heard presentation about history, culture and industry of the Netherlands and Japan and each morning we had Japanese language lecture. This lecture was for me not so useful, because I was going to Japan a half year later and I forgot the most vocabularies in this time.

The topic of my research was " A Study on Aesthetic Design of Wrinkle Form of Industrial Product Surfaces". I created two digital programs.

I also took 2 classes for one semester. During these classes, I came in contact with international students from Italia, Malaysia, Sweden and Indonesia, who participate in a 2 year master course at Keio.

During the research work I had time to visit some places in Japan. But it was not time enough to see it all. I went to Tokyo very often, because Yokohama is located in the near of the capital town. I also visited the cities Nagoya, Kyoto, Nara, Hakone, Kamakura and the area around Mt. Fuji.

The daily life in Japan is a little bit differently to Germany. Especially in the first two weeks after my arrival. The most people outside the university can`t speak English and it was sometimes very difficult with the people to communicate. It takes a lot of time for example in the office or supermarket to explain what do you want. It was a very good experience.

3. Travel Schedule

3.1 Trip to Tokyo

Departure from Berlin 07.04.2005

Flight BA	Berlin TXL	→ London
Flight BA	London	→ Narita (Tokyo)

Arrival in Yokohama 08.04.2005

3.2 Trip to Berlin

Departure from Tokyo 19.09.2005

Flight BA	Narita (Tokyo)	→ London
Flight BA	London	→ Berlin TXL

Arrival in Berlin 19.09.2005

3.3 Trip to Mt. Takao (03.05.2005 – 03.05.2005)

Mt. Takao is to find in the northwest of Tokyo. The highest point is 633 m and you can see from the top of the mountain Tokyo.

3.4 Trip to the Mt. Fuji Lake and Hakone (21.05.2005 – 22.05.2005)

Hakone and the lakes around the Mt. Fuji is a very beautiful area. It is very easy to travel around in this area by bus, boat rope way or by car. You can see a fantastical green nature and in the summer time is not so hot there.

3.5 Trip to Nagoya – Aichi (Expo) (20.06.2005 – 21.06.2005)

The Expo 2005 this year would be organized from Japan. The Japanese name is “Aichi”, because you can find it in the prefecture Aichi. The biggest city in the near of the Expo 2005 is Nagoya.

3.6 Trip to Kyoto (10.08.2005 – 12.08.2005)

Kyoto is a city with a lot of beautiful temples and shrines. Every time is it very crowded there, because many tourist want to see the typical Japanese architecture.

4. Technical Report

4.1 Introduction

Computer generated shaded images have reached an impressive degree of realism with the current state of the art. They are not so realistic, however that they would fool many people into believing they are real. One problem is that the surfaces tend to look artificial due to their extreme smoothness. What is needed is a means of simulating the surface irregularities that are on real surfaces. The idea is it to using parameter values of parametrically defined surfaces to index into a texture definition function which scales the intensity of the reflected light. By tying the texture pattern to the parameter values, the texture is guaranteed to rotate and move with the object. This is good for showing patterns painted on the surface, but attempts to simulate rough surfaces in this way are unconvincing.

Recent work in computer graphics has been devoted to the development of algorithms for making pictures of objects modeled by other than the conventional polygonal facet technique. In particular, several algorithms have been devised for making images of parametric surface patches. Such surfaces are defined by the values of three derivative functions:

$$X = X(u,v)$$
$$Y = Y(uv)$$
$$Z = Z(u,v)$$

as the parameters vary between 0 and 1. Such algorithms basically consist of techniques for inverting the X and Y functions. That is, given the X and Y of a picture element, the corresponding u and v parameter values are found. This parameter pair is then used to find the Z coordinate of the surface to perform depth comparisons with other objects.

To calculate the surface normal we first examine the derive actives of the surface definition functions:

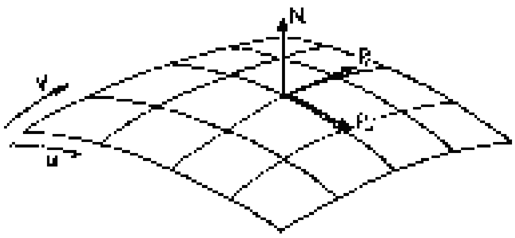
$$P= (X,Y,Z)$$

Two new vectors:

$$Pu = (Xu,Yu,Zu)$$
$$Pv = (Xv,Yv,Zv)$$

These two vectors define a plane tangent to the surface at that point.

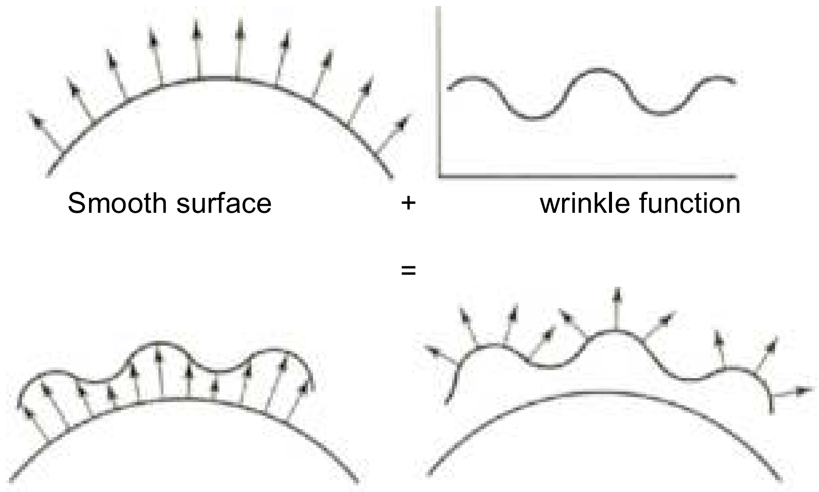
$$N = Pu \times Pv$$



Definition of Normal Vector

Images of smooth surfaces made directly from the patch description do not have the usual artifacts associated with polygonal facets, they do indeed look smooth. In fact they sometimes look too smooth. To make them look less artificial it is necessary to simulate some of the surface irregularities of real surfaces.
In order to simulate bumpy or wrinkly surfaces one might use, as the defining texture pattern, a digitized photograph of a bumpy or wrinkly 286 surface. The images usually looked like

smooth surfaces with photographs of wrinkles glued on. The main reason for this is that the light source direction when making the texture photograph was rarely the same as that used when synthesizing the image. In fact, if the surface (and thus the mapped texture pattern) is curved, the angle of the light source vector with the surface is not even the same at different locations on the patch.



Wrinkle surface

The normal vector to the new surfaces is derived by talking the cross product of its partial derivatives.

$$N' = P_u' \times P_v'$$

The partial derivatives involved are evaluated by the chain rule.

$$\begin{aligned} P_u' &= d/du\, P' = d/du(P + F\, N/|N|) \\ &= P_u + F_u\, N/|N| + F\, (N/|N|)_u \\ P_v' &= d/dv\, P' = d/dv(P + F\, N/|N|) \\ &= P_v + F_v\, N/|N| + F\, (N/|N|)_v \end{aligned}$$

The formulation of the normal to the wrinkled surface is now in terms of the original surface definition functions, their derivatives, and the bump function. This is reasonable for the types of surface irregularities for which this process is intended where the height of the wrinkles in a surface is small compared to the extent of the surface.

In summary, we can now calculate the perturbed normal vector, N', at any desired u and v parameter value. This vector must still be scaled to a length of 1 by dividing by its length. The result is then passed to the intensity calculation routines in place of the actual normal N.

4.2 Wrinkle Surface

4.2.1 Fractal in Manufacturing

Fractals are mathematical (strictly geometrical) creatures. They can have non-integer.

The dimensions, it is the main difference between them and classic geometrical objects. They usually posses self-similar structure: the same or similar outline with various magnifications. A fractal is an object or quantity that displays self-similarity, in a somewhat technical sense, on all scales. The object need not exhibit *exactly* the same structure at all scales, but the same "type" of structures must appear on all scales. For classical self-similar curves the reduction coefficient can be facultative, for example 0.13, 1/27. However these coefficients are strictly determined for fractals; they depend on concrete shape. Dependence between reduction coefficient (so-called scale coefficient) and the number of reduced fragments “a’ is the common feature of self-similar objects where *d* – self-similarity dimension.

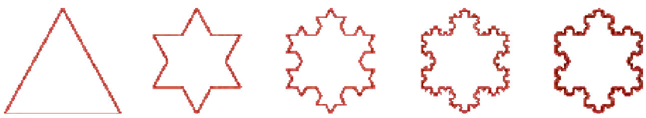
$$A = \frac{1}{S^d} \qquad d = \frac{\log a}{\log \frac{1}{S}}$$

It is one of the forms of fractal dimensions. Of course there are other forms, for example compass or box dimensions. Box dimension is the best for topography of engineering surfaces after manufacturing processes modeling (these surfaces have both self-similar and non self-similar structures). For example box dimension was used for estimation of fractal dimension of anisotropic surfaces.

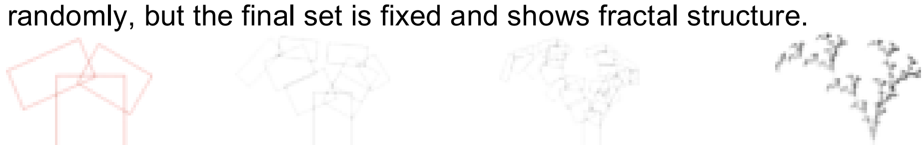
4.2.2 Fractal types

There are many different fractal types which are not covered when explaining coloring algorithms because they are either not common in fractal explorations or because they are not present in the popular fractal software packages.
But basically, we can classify fractal types in six main groups:

a) Fractals derived from standard geometry by using iterative transformations on an initial common figure like a straight line (the Cantor dust or the von Koch curve), a triangle (the Sierpinski triangle), or a cube (the Menger sponge). The first fractal figures invented near the end of the 19th and early 20th centuries belong to this group.

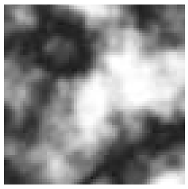


b) IFS (Iterated Function Systems). This is a type of fractal introduced by Michael Barnsley. The structure of these fractals is described by a set of affine (linear) functions that compute the transformations undergone by each point by homothetic, translation, and rotation. The functions introduced into the system are selected randomly, but the final set is fixed and shows fractal structure.

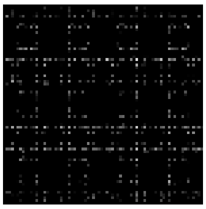


c) Strange attractors. These sets can be considered as the representation of a chaotic movement (in no place and no time identical). These attractors are very complex and composed by a line of infinite length drawing tightly intertwined loops that never crosses its own trajectory.

d) Plasma fractals. Created with techniques like the fractional Brownian motion (fBm) or the midpoint displacement algorithm, these fractal type produce beautiful textures with fractal structure, like clouds, fire, stone, wood, etc. widely used in CAD programs. Skilled fractal artists love plasma to create textures or backgrounds in their images.



e) L-Systems, also called Lindenmayer systems, were not invented to create fractals but to model cellular growth and interactions. A L-System is a formal grammar that recursively applies its rules to an initial set. As a result, sometimes, a fractal structure is created.



4.2.3 Idea of fractal iteration

In the majority of research works in these field fractal iterations are done till fractal dimension is determined. In model proposed here it can be possible but it is not necessary. It is a result of the choice of similarity criterion between the model and real surfaces. One can say that some kind of fractal interpolation is used here. The sections (at various angular positions) were analyzed using bearing length and area ratio curves. The analysis of computer generated surfaces allows us to avoid ambiguity during interpretation of results of roughness measurement. This ambiguity is the result of variety of profilometer used in industry (various productions date, various companies, not well known numerical procedures, different verification criteria). Model presented here was developed by simultaneous execution of fractal sequences along x and y axes (see Fig. 1). It assured stationary of surfaces (stationary of random area according to nomenclature of random signals.

4.2.4 Koch Snowflake

One of the simply fractal shapes is “Koch snowflake”. The method of creating this shape is to repeatedly replace each line segment with the following 4 line segment.

It is built by starting with an equilateral triangle, removing the inner third of each side, building another equilateral triangle at the location where the side was removed, and then repeating the process indefinitely. The Koch snowflake can be simply encoded as a Lindenmayer System with initial string “F - - F - - F”, string rewriting (can be used to generate fractals with dimension between 1 and 2) rule “F→ F + F - - F + F” and angle 60°. Zero through third iterations of the construction are shown above. The fractal can also be constructed using a base curve and motif, illustrated below.

The capacity dimension is then

$$d_{cap} = -\lim_{X\to\infty}\frac{\ln N_X}{\ln L_X}$$

$$d_{cap} = -\lim_{X\to\infty}\frac{\ln(3*4^X)_X}{\ln 3^{-X}}$$

$$d_{cap} = \lim_{X\to\infty}\frac{\ln 3 + n\ln 4}{n\ln 3}$$

$$d_{cap} = \frac{2\ln 2}{\ln 3}$$

$$d_{cap} = 1,2618...$$

4.2.5 Mandelbrot

Mandelbrot fractal use complex numbers, and the only difference relates to the use of the constant that is added on each time.

First we will choose a point to test: for example 0.2 + 0.5i. Next we define two complex variables, Z and C, such that

Z = 0 +0i

and C = value of the test point,

C = 0.2 + 0.5i.

The following algorithm is iterated:

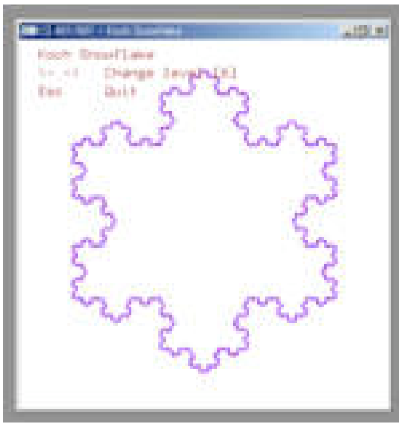
1. Calculate value of Z^2 + C
2. place the result in Z
3. → Z has new value → find a point on the complex plane that corresponds to Z
4. check the distance between Z and origin 2

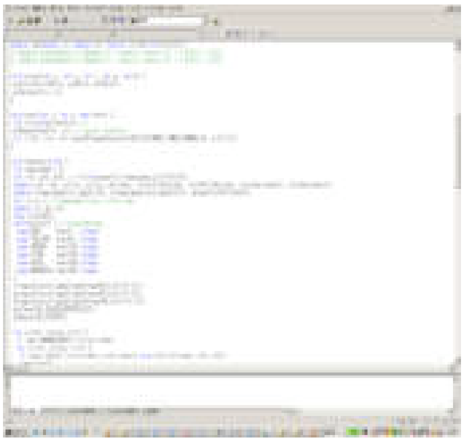
   → Z > 2 → Z is outside of the circle → you have to stop the value for Z
   → Z < 2 → moving back to the top of the loop and calculate a new value for Z and check it

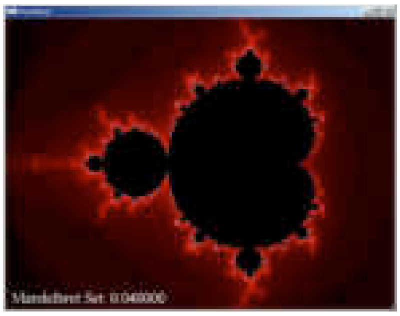
Example:
Z = 0 + 0i C = 0.2 + 0.5i
1. Iteration Z = 0.2 + 0.5i
2. Iteration Z = -0.01 +0.7i
3. Iteration Z = -0.2899 + 0.486i
....

We have to set a limit of maximum iterations. The maximum iteration of value is 256. That means, that if the program goes through the iterated loop 256 times and Z has never strayed outside the bailout circle. The maximum number of iterations has very important effect of the rendered image. Setting it too low will result in a bloated, “rounded off” fractal, where the fine details are missing. Setting it too high will just cause the rendering to slow down, especially in the black interior regions, but otherwise have no other effects. When zooming in, however, it is important to set the max iterations to higher value, or else the image will become more bloated and rounded as the scaling factor gets larger. There are some instances however, where setting the max iterations to a lower value may bring out more detail in the image – this depends on the color gradients in use and also on the particular region of image that is being viewed.

4.3 Program and Conclusion







- Texture with artistic effects of texture processing decorate articles of all kind of materials with help defines.
- There are no materials completely void to texture processing.
- Outstanding economy, stability, process ability.
- Outstanding decorativeness increases product value.
- Enhances commercial value, and prevents problems due to scratches and blemishes.
- Product textures are determined from the dual perspectives of functionality and creative design after accurately identifying product needs.

5. Life in Japan

5.1 Social Life

The first week in Yokohama was hard working for an international student. I had a lot of things to do. (To go alone to the office and so on.) It was sometimes very difficult. For example, to find a right way, to ask people "how much is it or can you explain me, how can I find, to fill out a lot of letters (written in Japanese) in the immigration office, to get a bank account or a mobile-phone… and you need this all here.

I was so lucky, then I get in touch with many students in the university → in the lab. They were very hopefully for me, because my Japanese was very badly or is it.
I met a lot of international people in the dormitory and in my two classes (from Italia, Sweden, Indonesia, Malaysia) We had a lot of fun together in the lecture and outside the university. We went to Tokyo or sojourn us in Yokohama. But the most time did we spend together in lunch B. This is a place, where we could cook together, watch TV or we had a party.

In Tokyo you can find a lot of different places, where you can go out. Roppongi is a very with many pub`s, discotheques and international restaurants. Very often we went to Shibuya and had a drink there. Here you can find many young and crazy people. The streets were so crowded there, that we had sometimes a problem to find a coffee shop with free places.

The best in all Japan is that you have overall convenient stores, which have open around-the-clock and every day.

5.2 University Life

The most time in the university I spend in the laboratory. Two times in the week I had lectures. There we had a lot of presentations and to design a game, which was very funny to play on the end of the course. The situation between university and dormitory was very good arranged. My Professor (H. Aoyama), my students and the international centre were very helpfully in each situation. I learned a lot of Japanese culture and new technical information in my lab. I visited with my Professor a manufacture industry. That very interesting and useful for my research.

6. Summary

The 5 month, did I spend in Japan in Prof. H. Aoyama Lab, was the best time in my study life. In this half year I learned a lot of design and system engineering and to write a program to get by on technical systems. I learned a lot of teamwork, because we don`t have it in Germany.

I hope I can go back to Japan for a short time – maybe for working or as a PhD student.

Mata ne