

Simulation of a particular reflexion group: The kaleidoscope

DEMEULENAERE Paul

University of Teesside Tees Valley, Middlesbrough, TS1 3BA UK – 2007

1. Abstract

The reflection groups can result really interesting effects. The kaleidoscopes are a good sample of reflection group. The most often considered like a toy, this system could be very useful in computer graphics.

We represent the result of the kaleidoscope as a two dimensional plane with a particular shape and particular computed texture coordinates to display multiple reflection of the real object.

2. Introduction

In the antiquity, the kaleidoscope was already known by the ancient Greek but the system has been reinvented by a Scot, Sir David Brewster in 1816 while he was studying the light polarization. The first kaleidoscope was a pair of mirror in a tube. Now day, it is more a toy than an object of study but his based on interesting physical concepts.

The kaleidoscope can be represented thanks to reflection group; indeed, it is a group of action in a finite space which is generated by reflections.

This research paper will give a simple model of the real interaction into a kaleidoscope to present a rendering strategy in real time.

3. Model of real system

The kaleidoscope is a set of several mirrors; the following part will be not a thorough description of the physical concept but more a way of representation of this effect.

First, we will admit a simple and perfect plane mirror. The surface of the mirror reflects the light. The result is an inverted image of real objects.

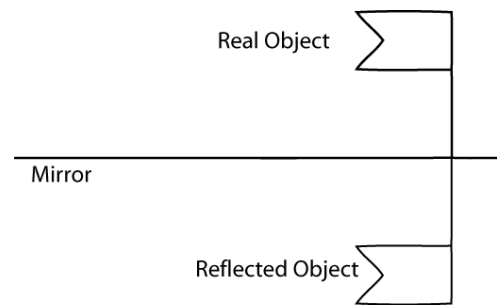


Figure 1: Simple system with one mirror

If we add a mirror on an edge of this mirror, we have almost a kaleidoscope; the reflected object is copy by the second mirror and the previous effect is repeated several times depending of the angle between the two mirrors.

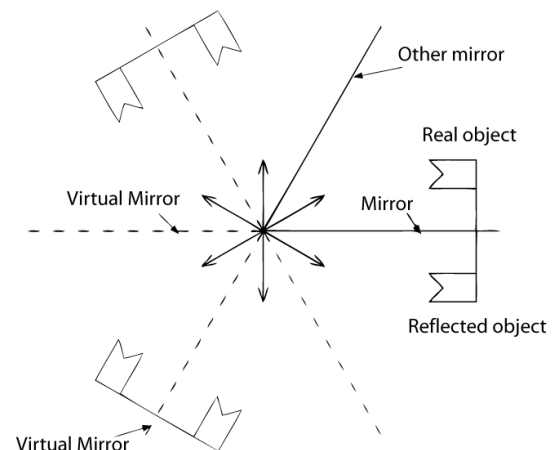


Figure 2: Kaleidoscope with two mirrors at 60°

It is possible to represent a kaleidoscope as a set of tree mirror forming an equilateral triangle. In this configuration, the system gives an infinite image.

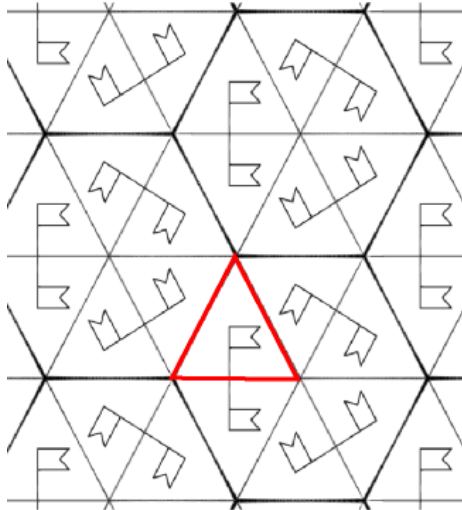


Figure 3: Result with tree mirrors (in red, the real object)

Studying these particular transformations, it is possible to identify a constant repetition of twelve triangles. This element is visible in the following figure.

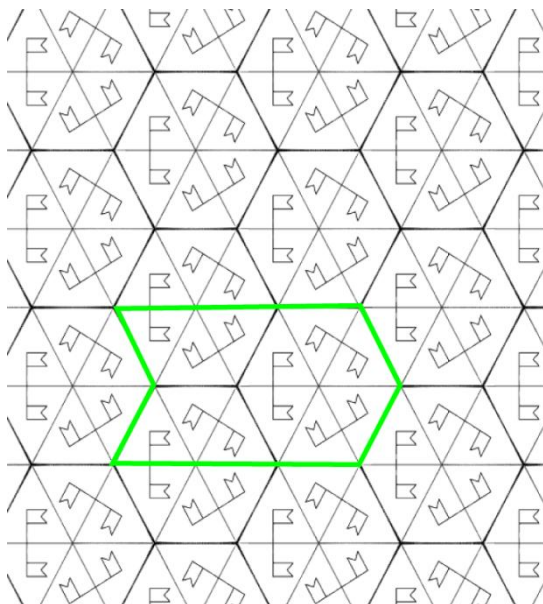


Figure 4: The repeated element with tree mirrors (in green)

4. Rendering strategy

An interesting and really fast method of rendering could be found thanks to the texture mapping. The texture mapping is a common technique in real time graphics. The first application to 3D graphics was been done in 1974 by Dr Edwin Catmull.

It is very simple, for each vertex, there is a particular texture coordinate which represent the position value of the vertex into a two dimensional image. Most of graphical API support the mapping of textures, for information, this project has been realized in OpenGL.

First and parallel to the previous part, it will be interesting to show how simulate a simple system with only one mirror. We consider a simple scene represented into a texture by four textures coordinates which will call A, B, C and D.

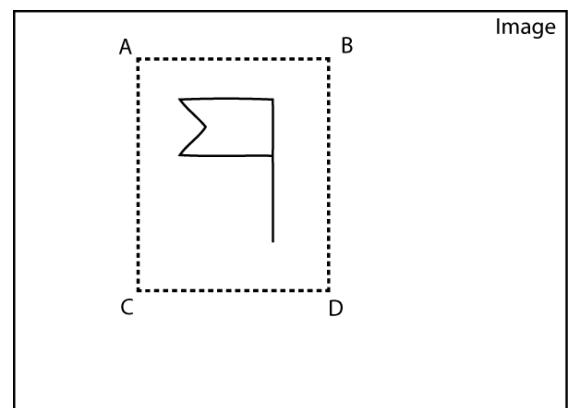


Figure 5: Sample of textures coordinate in image space

We represent the mirror thanks to six vertices and a particular organisation, the following figure shows this technique.

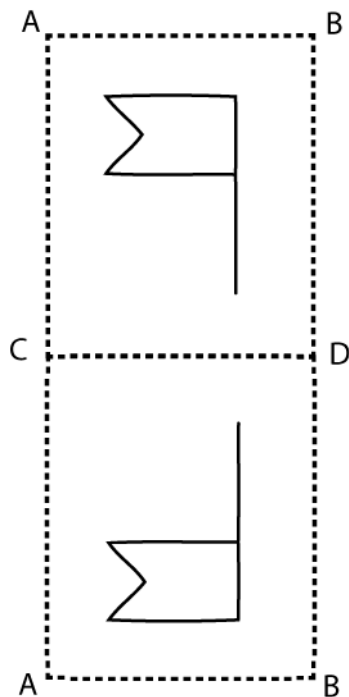


Figure 6: Simple reflection simulation

In the situation of two mirrors, this is the number of duplication wanted which defines the angle. It will be only possible with a pair number of duplications and equals or above four.

To create the shape to represent the effect with two mirror, we can simply following a circle and create triangles with wanted angle. In fact, this is the same method used when we want to creates the triangles at the top of the cylinder. There is only three textures coordinate represented by a triangle in the based image.

The texture coordinates on this shape are not really complicated, the centre is still with the same value

and the texture coordinates on the "circle" are successively the second other then the third, the second and again the third until the last triangle created. The following figure gives an example with six triangles (so, the angle is $\frac{\pi}{3}$).

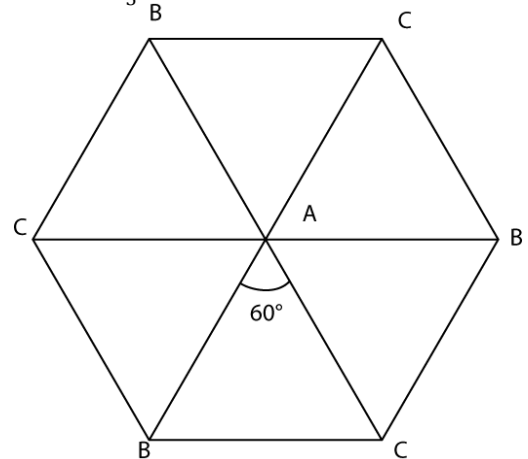


Figure 7: Sample of two mirrors shape

As it is showed before, when we have tree mirror, a particular shape is repeated:

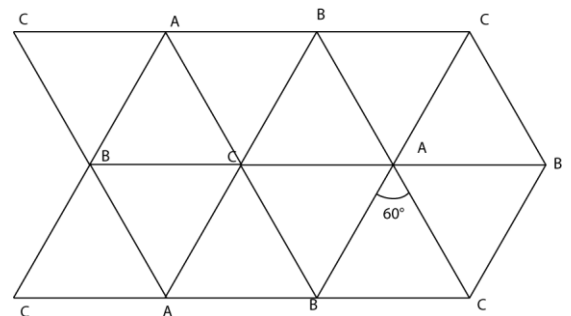


Figure 8: Element repeated for a tree mirrors kaleidoscope

It is possible to find easily textures coordinates in this shape simply started with a particular face which will represented the real object and looking for logical reflection for each mirror. Then, this shape is created with a fixed size, copy and translate several times to fill the screen and give an effect of infinite.

It can be necessary to give some words about the computation of textures coordinate on the square image. In the two situations, we have to define a triangle into a image to avoid the deformation when the texture is applied.

The position of the vertex in the image will define the texture coordinate. With the most of graphical API, we have to return this value between 0 and 1.

With two mirror, the triangle is isocèle. The main vertex, the vertex share by the two identical edges, will represent the center of the figure, it is possible to compute the value of the angle at the vertex dividing 2π by the number of duplication. With this value, we have enough to draw a representative triangle on the image without any distortion in the kaleidoscope simulation.

With tree mirror, it is simplest, the triangle is always equilateral, the angle at each vertex is equal to $\frac{\pi}{3}$.

For an interesting effect, it could be important to permit some main transformations on this last triangle, the rotation, scale and translate for example. In fact, it will be like the kaleidoscope is displaced into a two dimensionnal environment.

The texture coordinate could loop or not in the case of a overtaking of the triangle. In most of case, we kept it into the image and the problem doesn't appears.

5. Result and conclusion

Even if the effect rendered is more interesting with the modification of differents parameters in real time, some demonstration are following, on the left side, the original image with, in red, the triangle which represents the real object in the kaleidoscope, on the right side, the result.

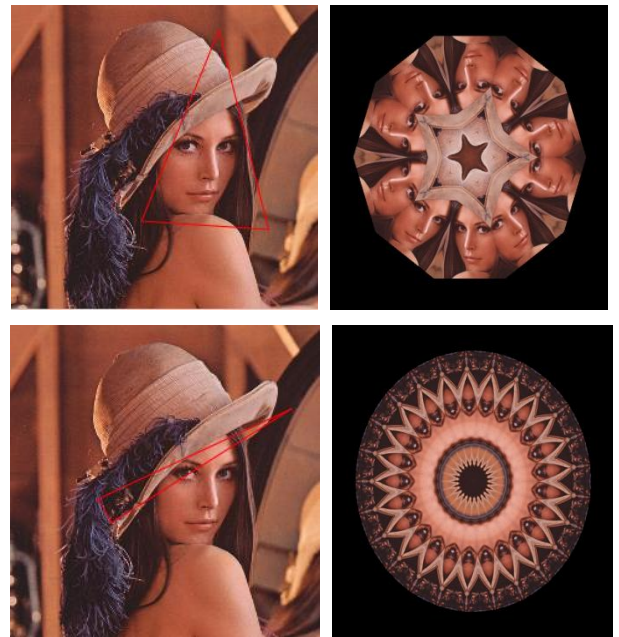


Figure 9: Sample of result of rendering of two mirrors (with Lena, common test sample of image processing algorithms)

This render is really faster, because it is display thanks to OpenGL. It is not necessary to reload the texture on each frame, the most of the time, when the parameters are changed, we just need to update texture coordinates.

The following samples show the result with tree mirrors set as an equilateral triangle.

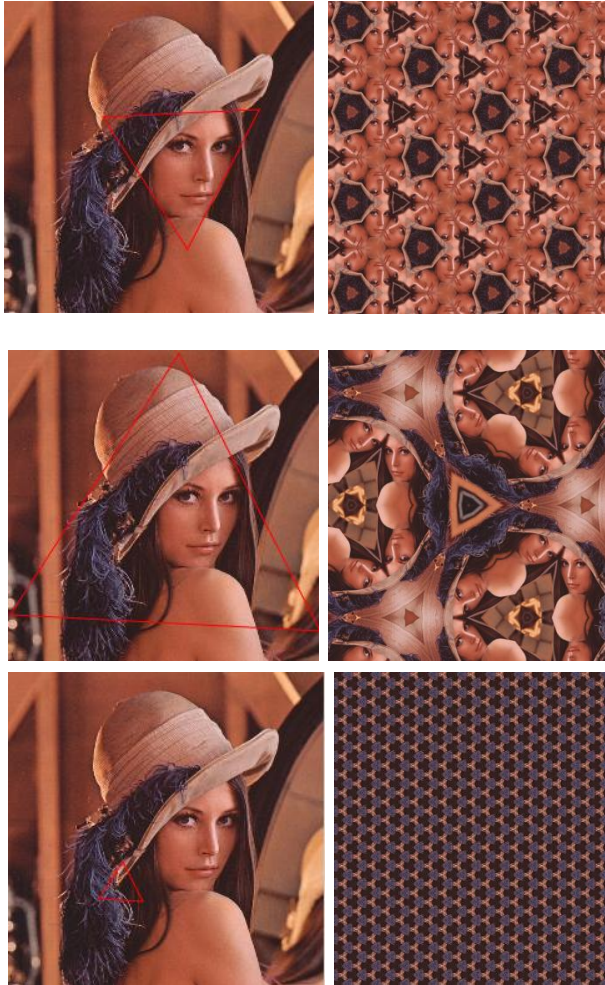


Figure 10: Sample of result of rendering of tree mirrors (with Lena, common test sample of image processing algorithms)

These techniques gives a really nice and impressive result, the computation is fast and can be implemented for a lot of domain, from the video game to the procedural texture generation.

To conclude, this simulation is only describe for two or three mirrors but it could be interesting to extend this study in order to have a fully procedural effect where the number of mirrors and theirs configurations will be a parameters modifiable.

6. References

- **Roe Goodman**, the Mathematics of Mirrors and Kaleidoscopes, April 2004, <http://www.math.rutgers.edu/pub/goodman/monthly.pdf>
- **Richard Kane**, Reflection groups and invariant theory, January 2003, <http://www.cms.math.ca/Publications/Reviews/2003/rev4.pdf>
- **W. W. Rouse Ball and H. S. M. Coxeter**, Mathematical Recreations and Essays, May 1987.

