Hybrid Animation Model of Multi-object in Fractal Form based on Metamorphic Interpolation and Partitioned-Random Iteration Algorithms

Tedjo Darmanto[1], Iping Supriana Suwardi[2], and Rinaldi Munir[3]

[1]Sekolah Tinggi Manajemen Informatika & Komputer, AMIK-Bandung
[2,3]School of Electrical Engineering and Informatics, Bandung Institute of Technology
[1]tedjodarmanto@gmail.com, [2]iping@informatika.org, [3]rinaldi@informatika.org

Abstract: Traditionally simple object is drawn to the screen by drawing functions in pixels, so the slight different form of object should be redrawn by different function. On the other hand, naturally fractal object is drawn by a single algorithm through drawing pixels at the position according to the code of fractal such as IFS code, and interactively to change the form of or to animate the fractal object is accomplished simply by changing the code of fractal at any time. So in this case, animation model of objects in fractal form has an advantage over animation model of objects in traditional form. IFS code is one way of the inverse problem representation of object in fractal form. The IFS fractal objects can be reconstructed by random iteration algorithm and the multi-object of fractal can be reconstructed by partitioned-random iteration algorithm from the IFS code sets. To have the metamorphic animation, the interpolation between the IFS code of source and target objects can be done by a metamorphic animation algorithm. The hybrid animation model is the combination model of a metamorphic animation on one type of object and the multi-object animation independently with or without synchronized mode based on the metamorphic interpolation of IFS code coefficients for the metamorphic animation and the partitioned-random iteration algorithm for the multi-object animation.

Keywords: Multi-object, hybrid animation model, metamorphic interpolation, partitioned-random iteration algorithm, synchronized mode, IFS code

1. Introduction

Normally an animation of an animated object is conducted easily by drawing repeatedly and slightly changing the position or size. A geometrical simple object can be drawn by means of drawing functions, such as draw-line, draw-rectangle, draw-oval etc. A Problem is raised when there is the need of making an animation of a complicated object by means of drawing functions. Animation of fractal object has many advantages, first it has objects that are easy to be controlled over and also it has relatively smaller size of animated objects needed in fractal form than in non-fractal form when those are stored. The second advantage of fractal model is it can be constructed easily to represent a lot of natural objects, such as trees, clouds etc., rather than non-fractal model. The other advantage of the fractal model is it can be zoomed and still has good resolution over the non-fractal model [1]. The IFS (iterated function systems) code sets as the inverse problem of any objects in fractal form contain the scaling, rotation and translation factors of affine transformation and represented by the coefficient-a, b, c, d, e and f [4]. The number of function in a set depends on the complexity of the fractal object. Each transformation equation pin point to the next position of point within the object that depends on the previous one can be expressed as mathematics expression as presented in the Figure-1 below. Those code set with probability factor-p can be reconstructed become objects of fractal by means of random iteration algorithm and become multi-objects of fractal by means of partitioned-random iteration algorithm [20]. So at any time the position and or the size, and also the form of IFS fractal object can be changed easily simply by drawing many pixels

Received: June 3rd, 2013. Accepted: September 6th, 2013

randomly at the position determined by IFS code over and over again. The most remarkable advantage of fractal object is the easy way of its manipulation such as evolutional changes between two slightly different IFS code sets by interpolating of its factors as a metamorphic animation and can exhibit a special effect such as rotational effect around Z- axis perpendicular to the 2D space [19].

$$w\begin{bmatrix}x\\y\end{bmatrix} = \begin{bmatrix}a & b\\c & d\end{bmatrix} * \begin{bmatrix}x\\y\end{bmatrix} + \begin{bmatrix}e\\f\end{bmatrix}$$

Figure 1. Affine transformation equation

2. Fractal and IFS Code

In this section there are many IFS fractal objects and multi-objects versions with the reconstruction algorithm model are presented and used in the simulations of the hybrid animation model between the metamorphic and non metamorphic animations with and without synchronization modes that are presented in the simulations section.

A. IFS Fractal Object

The inverse problem result of an object can be represented as the set of IFS code in text format form that represents the self-iterative affine transformation functions (self-affine). According to collage theorem, part of object can be encoded as a single self-affine function and

Table 1a. 2D IFS circle-like fractal type-1

12 functions of 2D IFS code of circle-like (1) fractal						
a	b	c	d	e	f	p
0.10	0.00	0.00	0.00	0.00	-1.00	8.03
0.09	-0.10	0.05	0.05	0.50	-0.84	8.03
0.05	-0.17	0.09	0.09	0.84	-0.50	8.03
0.00	-0.20	0.10	0.00	1.00	0.00	8.03
-0.05	-0.17	0.09	-0.10	0.84	0.50	8.03
-0.09	-0.10	0.05	-0.17	0.50	0.84	8.03
-0.10	0.00	0.00	-0.20	0.00	1.00	8.03
-0.09	0.10	-0.05	-0.17	-0.50	0.84	8.03
-0.05	0.17	-0.09	-0.10	-0.84	0.50	8.03
0.00	0.20	-0.10	0.00	-1.00	0.00	8.03
0.05	0.17	-0.09	0.10	-0.84	-0.50	8.03
0.09	0.10	-0.05	0.17	-0.50	-0.84	8.03

Table 1b. 2D IFS circle-like fractal type-2

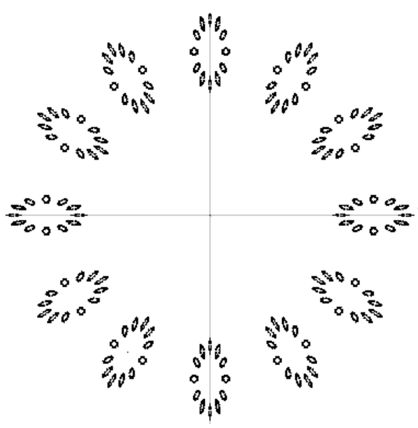
12 functions of 2D IFS code of circle-like (2) fractal						
a	b	c	d	e	f	p
0.20	0.00	0.00	0.10	0.00	-1.00	8.03
0.17	-0.05	0.10	0.09	0.50	-0.84	8.03
0.10	-0.09	0.17	0.05	0.84	-0.50	8.03
0.00	-0.10	0.20	0.00	1.00	0.00	8.03
-0.10	-0.09	0.17	-0.05	0.84	0.50	8.03
-0.17	-0.05	0.10	-0.09	0.50	0.84	8.03
-0.20	0.00	0.00	-0.10	0.00	1.00	8.03
-0.17	0.05	-0.10	-0.09	-0.50	0.84	8.03
-0.10	0.09	-0.17	-0.05	-0.84	0.50	8.03
0.00	0.10	-0.20	0.00	-1.00	0.00	8.03
0.10	0.09	-0.17	0.05	-0.84	-0.50	8.03
0.17	0.05	-0.10	0.09	-0.50	-0.84	8.03

Table 2a. 2D IFS rectangle fractal type-1

4 functions of 2D IFS code of rectangle (1) fractal						
a	b	c	d	e	f	p
0.8	0.05	0.00	-0.0	-0.02	1.30	25.0
0.0	0.00	-3.57	-0.2	0.18	0.60	25.0
-0.8	-0.05	0.00	0.0	0.02	-0.30	25.0
0.0	0.00	3.57	0.2	-0.18	0.40	25.0

Table 2b. 2D IFS rectangle fractal type-2

4 functions of 2D IFS code of rectangle (2) fractal						
a	b	c	d	e	f	p
0.8	0.02	0.00	0.0	-0.02	1.90	25.0
0.0	0.00	-7.17	-0.2	0.13	0.88	25.0
-0.8	-0.02	0.00	0.0	0.02	-0.44	25.0
0.0	0.00	7.17	0.2	-0.13	0.59	25.0



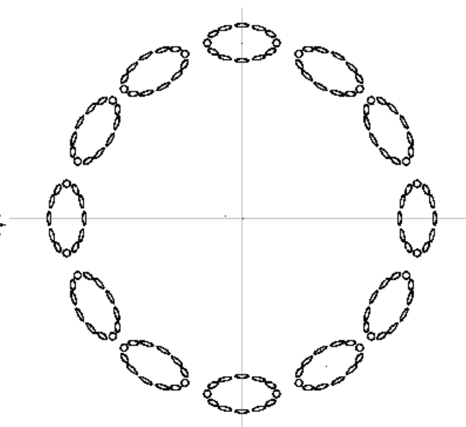


Figure 2a. Twelve points object type-1
Figure 2b. Twelve points object type-2

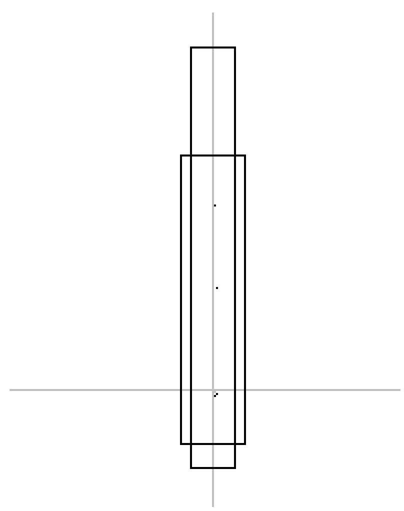


Figure 3. Two rectangles: one is in top the other

the collection of self-affine functions can be decode back or generated becoming an object of fractal as the composition of each part of the object iteratively by random iteration algorithm based on its probability factor of each function [2,3,5]. As the first two examples of object that will be used in the first of two animations, a pair of circle-like and a couple of rectangle IFS fractals are displayed in Figure 2a and b, and in Figure 3. As the IFS code sets with probability factor for each function in percentage, related to the examples of the IFS fractals mentioned above are presented in Table 1a and b and in Table 2a and b below respectively. The first pair of IFS code sets consist of 12 functions for each object and representing circle-like object that has 12 ticks around closed to the edge of the clock, so each function representing one tick mark for every five minutes all around of 60 minutes. In the first object the form of each tick is like an oval standing laterally around the centroid as illustrated in Figure 2a. But in the second object the form of each tick is like an oval standing radially around the centroid as illustrated in Figure 2b. The second pair of IFS code sets consist of 4 functions for each object and representing rectangle object and can be used as a couple of clock arms, and are used as the hour and minute pointers of multi-object of clock [20] as illustrated in Figure 3.

As the second two examples of object that will be used in the second animation, a pair of cloud-like and a triple of rectangle IFS fractals are displayed in Figure 4a and b, and in Figur 5. As the IFS code sets related to the examples of the IFS fractals mentioned above are presented in Table 3a and b and in Table 4a, b and c below respectively. The first pair of IFS code sets consist of 10 functions for each object and representing cloud-like object that has 10 clusters distributed in layers horizontally and vertically. The first object of cloud-like fractals in Figure 4a is the sheared translation mode in x-direction version of the second object in Figure 4b and vice versa, so the scenario of translational and shearing animation of cloud-like object is prepared. The triple of IFS code sets consist of 5 functions for each object and representing rectangle object with a hole in centroid represented by the last function as a triple of windmill propeller as illustrated in Figure 5, have the same size, but in the different orientation by 120 degrees from one blade to others. By rotating a triple blades as a propeller with the same speed around centroid, so the scenario of rotational animation of windmill propeller is prepared.

Table 4a. 2D IFS rectangle fractal type-3a

5 functions of 2D IFS code of rectangle (3a) fractal						
a	b	c	d	e	f	p
-0.8	-0.06	0.00	0.00	0.04	1.35	24.0
0.0	0.00	2.55	-0.20	-0.25	0.66	24.0
0.8	0.06	0.00	0.00	-0.04	-0.25	24.0
0.0	0.00	-2.55	0.20	0.25	0.44	24.0
0.3	0.00	0.00	0.01	0.00	0.00	4.0

Table 4b. 2D IFS rectangle fractal type-3b

5 functions of 2D IFS code of rectangle (3b) fractal						
a	b	c	d	e	f	p
0.173	-0.36	-0.30	0.63	-1.19	-.645	24.0
0.954	-2.00	0.55	-1.15	-0.45	-.547	24.0
-.173	0.36	0.23	-0.63	0.23	0.095	24.0
-.954	2.00	-0.55	1.15	-0.51	-.003	24.0
0.015	-0.01	-0.01	0.025	0.00	0.00	4.0

Table 4c. 2D IFS rectangle fractal type-3c

5 functions of 2D IFS code of rectangle (3c) fractal						
a	b	c	d	e	f	p
0.173	0.36	0.30	0.63	1.19	-.645	24.0
0.954	2.00	-0.55	-1.15	0.45	-.547	24.0
-.173	-0.36	-0.23	-0.63	-0.23	0.095	24.0
-.954	-2.00	0.55	1.15	0.51	-.003	24.0
0.015	0.01	0.01	0.025	0.00	0.00	4.0

Table 5. 2D IFS cloud-like fractal type-1

10 functions of 2D IFS code of cloud-like (1) fractal						
a	b	c	d	e	f	p
0.91	-0.103	0.09	0.21	-0.028	0.13	10.0
0.68	-0.075	0.07	0.15	0.185	0.22	10.0
0.46	-0.050	0.05	0.10	0.259	0.31	10.0
0.44	-0.039	0.02	0.12	-0.008	0.29	10.0
0.40	-0.040	0.03	0.10	0.325	0.38	10.0
0.40	-0.040	0.03	0.10	0.061	0.36	10.0
0.09	-0.003	0.01	0.03	0.219	0.44	10.0
0.09	-0.003	0.01	0.03	0.123	0.45	10.0
0.09	-0.003	0.01	0.03	0.088	0.42	10.0
0.09	-0.003	0.01	0.03	-0.043	0.38	10.0

Table 6. 2D IFS cloud-like fractal type-2

10 functions of 2D IFS code of cloud-like (1) fractal						
a	b	c	d	e	f	p
0.81	0.620	0.09	0.309	-0.184	0.13	10.0
0.60	0.459	0.07	0.238	-0.074	0.22	10.0
0.40	0.301	0.05	0.166	-0.108	0.31	10.0
0.42	0.314	0.02	0.145	-0.361	0.29	10.0
0.37	0.278	0.03	0.139	-0.136	0.38	10.0
0.37	0.278	0.03	0.139	-0.376	0.36	10.0
0.08	0.059	0.01	0.035	-0.313	0.44	10.0
0.08	0.059	0.01	0.035	-0.421	0.45	10.0
0.08	0.059	0.01	0.035	-0.420	0.42	10.0
0.08	0.059	0.01	0.035	-0.502	0.38	10.0

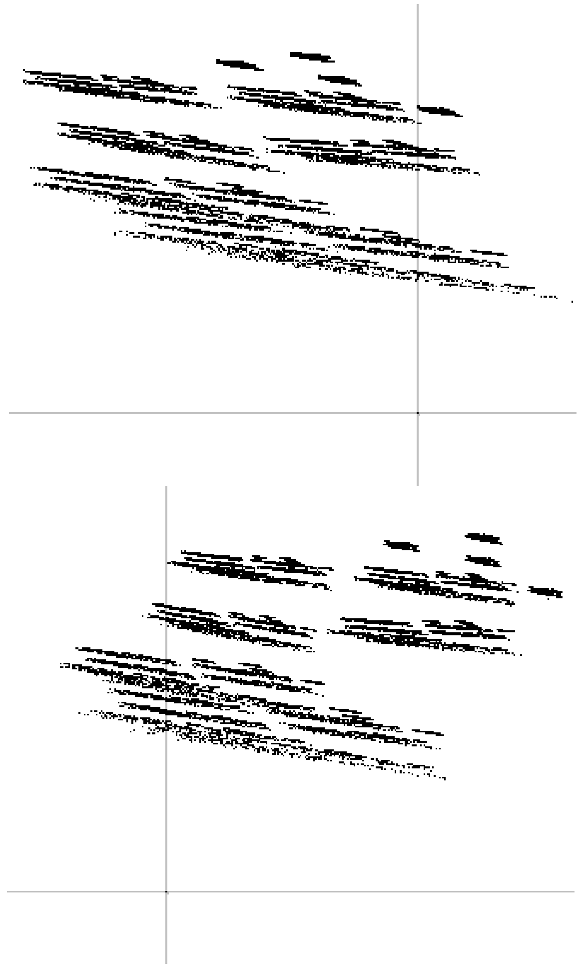


Figure 4a. Cloud-like object type-1
Figure 4b. Cloud-like object type-2

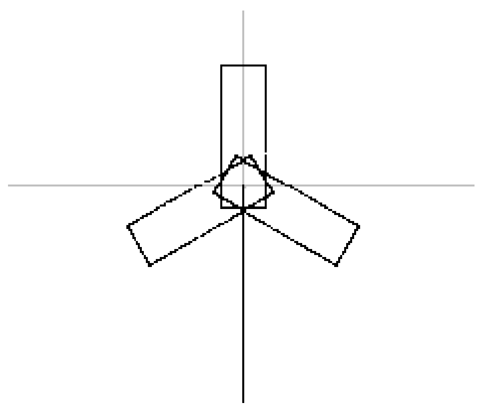


Figure 5. Windmill propeller

B. Multi-object of Fractal

A Composition of several primitive objects can be constructed becoming a multi-object of non-primitive object in fractal form by means of partitioned-random iteration algorithm. The simple clock model of fractal as an example of multi-object in the fractal form that consists of three primitive objects (two modified rectangles and one circle-like object) is displayed in Figure-6 below. To compose at least two fractal objects as a new multi-object of fractal, the partitioned-random iteration algorithm as a modified version of the random iteration algorithm is needed. This modified version of algorithm is based on the use of partitioned random number generation for each object in a multi-object, so the name of algorithm is coming from. In the Figure-7 below there is an illustration of the decision tree form that describes the mechanism of the partitioned-random iteration algorithm based on the probability values of self-affine functions of each object in the multi-object as the partitions of the whole [20].

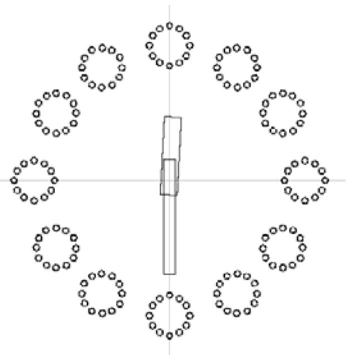


Figure 6. Multi-object of clock in fractal form

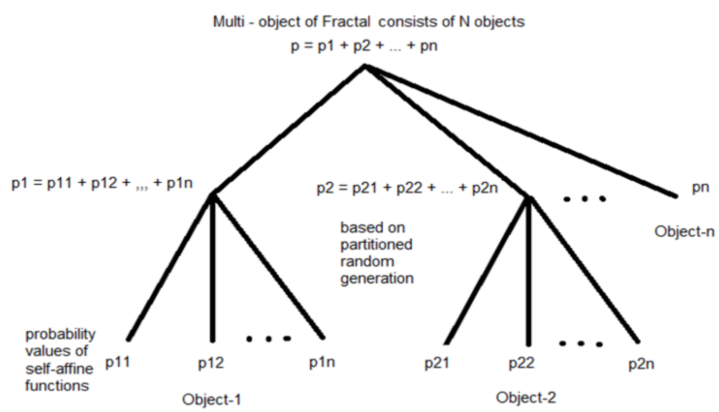


Figure 7. Decision tree form of partitioned-random iteration algorithm [20]

C. Rotational Operation

The main requirement to accomplish the rotational animation in both hybrid animations is the rotation around z-direction procedure, that change the next six 2D IFS code coefficients based on the current six IFS code coefficients iteratively by a small deviation angle (dt) [4] as described in the six equations below:

$$a' = a*\cos(dt)*\cos(dt) - (b+c)*\cos(dt)*\sin(dt) + d*\sin(dt)*\sin(dt) \quad (1)$$

$$b' = (a-d)*\cos(dt)*\sin(dt) + b*\cos(dt)*\cos(dt) - c*\sin(dt)*\sin(dt) \quad (2)$$

$$c' = (a-d)*\cos(dt)*\sin(dt) - b*\sin(dt)*\sin(dt) + c*\cos(dt)*\cos(dt) \quad (3)$$

$$d' = a*\sin(dt)*\sin(dt) + (b+c)*\cos(dt)*\sin(dt) + d*\cos(dt)*\cos(dt) \quad (4)$$

$$e' = e*\cos(dt) - f*\sin(dt) \quad (5)$$

$$f = e*\sin(dt) + f*\cos(dt) \quad (6)$$

3. Related Works

Since the introduction of iterated function system (IFS) in 1985 by Barnsley and Demko [3] and popularized by Barnsley's fern fractal [5] in a book entitled "Fractals Everywhere" in 1988 and revised version in 1993, there are so many researchers proposed a new algorithms and methods to enhance the previous IFS method. In the context of this paper, moreover there are also several research results in morphing area, such as are described in the following sub-sections.

A. IFS Fractal Algorithms and Methods

In their paper, Takezawa et.al. [6] proposed a high-accuracy quantization method for IFS parameters in fractal image coding by using a genetic algorithm (GA). The development of IFS-parameter quantization techniques is significant for the image coding, because its errors make more serious problems in the iteration procedures than the other quantization topics. Even if the errors are small, high-quality reconstructed images are not necessarily obtained. Therefore, the high-accuracy quantization methods are required for the parameters. The proposed method provides higher quality reconstructed images than a conventional method which merely minimizes the errors. Chang et.al [7], in their paper proposed a fixed-point-searching algorithm which can automatically determine the original size and the coordinates of a fractal image directly from its IFS code. With the proposed modification on the translation parameters in IFS code, the decoded image can be resized and relocated to become a desired one. Abiko et.al. [8] in their paper proposed a moment based encoding algorithm for iterated function system (IFS) coding of non-homogeneous fractal images with unequal probabilities. The proposed algorithm employs a variable elimination method using Gröbner bases with floating-point coefficients in order to derive a numerically solvable equation with a single unknown. The algorithm also employs a varying associated-probabilities method for the purpose of decreasing the computational complexity of calculating Gröbner bases. Raynal et.al. [9] exploited Genetic Programming (GP) techniques to generate randomly or interactively artistic "fractal" 2D shapes of non-linear IFS fractals efficiently by two kinds of scheme. Chu et.al. [10] in their paper introduced a new fast algorithm for generating fractals instead of the deterministic algorithm and the random iteration algorithm for a large scale computation purpose, without iterative checks to converge. Dasgupta et.al. [11] introduced an evolutionary algorithm to search for iterated function systems to encode black and white images that cannot be known in advance by using a variable length genotype to represent candidate solutions. Wadströmer [12] proposed an automatization of Barnsley's manual algorithm for the solution of the inverse problem of iterated function systems by identifying the fragments, of which the collage is composed, and then computing the parameters of the mappings. The automatic algorithm searches through a finite set of points in the parameter space determining a set of affine mappings. The algorithm uses the collage theorem and the Hausdorff metric. So the inverse problem is solved by the automatic algorithm to retrieve an IFS, including the number of mappings, from a digital binary image approximating the attractor induced by the IFS.

In their paper, Gue'rin et.al. [13] proposed a new descriptor for a shape feature by combining Iterated Function System (IFS) model and the notion of free form curves to add a real flexibility of fractal approximation techniques enriching the set of contractive operators which are candidate to model the self-similarity. This new descriptor named projected IFS model allows the reconstruction of a shape using a projection via the control points. It is adapted to the representation of both smooth shapes and fractal shapes. Zhang et.al. [14] in their paper proposed a new method to display IFS attractor based on fixed point of an invertible affine transformation with the property of self-similarity between different regions of IFS attractor. Experimental results show this method is effective and resolves defects of random algorithm for computing IFS attractor.

B. Morphing of Fractal

Through the work on fractal-based algorithm for metamorphic animation that is studied by Chen et.al. [15], the size of objects can be measured, for which the traditional way based on Euclidean geometry is failed. An image morphing based on pixel transformation approach that is proposed by Rahman et.al. [16] depict the transformation of pixels with their neighborhoods; this method is organized with the replacement of the pixel values of a source image and convolving the neighbor with the help of a mask that is fast and efficient for image morphing. In their paper, Zhuang et.al [17] studied a morphing IFS fractal by calculating local attractor's coarse convex-hull and selecting rotation matching between IFS's. The morphing procedure of two IFS's fractal attractors is done by interpolating the parameters of the iterated function. Normally morphing animation is dealing with two different objects as the start and target objects, but if the morphing animation is dealing with more than two different objects, a new approach is needed such as the method which is based on a family of multi-transitional IFS code approach [18,19].

4. Simulations

There are two kinds of simulation conducted in this paper as a comparison example between the synchronized and non synchronized mode versions. The first simulation is dealing with the hybrid metamorphic and non metamorphic animation without synchronization between the three rectangles representing the windmill propeller and the 10 functions of transitional 2D IFS cloud-like fractal. The second simulation dealing with the hybrid metamorphic and non metamorphic animation with synchronization between the two rectangles representing two arms of clock and the 12 functions of transitional 2D IFS circle-like fractal representing 12 tick around the arms. The first simulation exhibits the non synchronization effect between the windmill propeller rotated around the centroid and the sheared translation movement of cloud-like object from left to right as illustrated in Figure-8 below and vice versa. So in this simulation there is no correlation between the metamorphic animation of cloud-like object and the non metamorphic animation of the multi-object of windmill. The second simulation exhibits the synchronization effect between the position of the minute arm and the form of ticks as illustrated in Figure-9 below. So in the second animation there is a correlation between the metamorphic animation of 12 ticks object and the non metamorphic animation of the arms of the multi-object of clock.

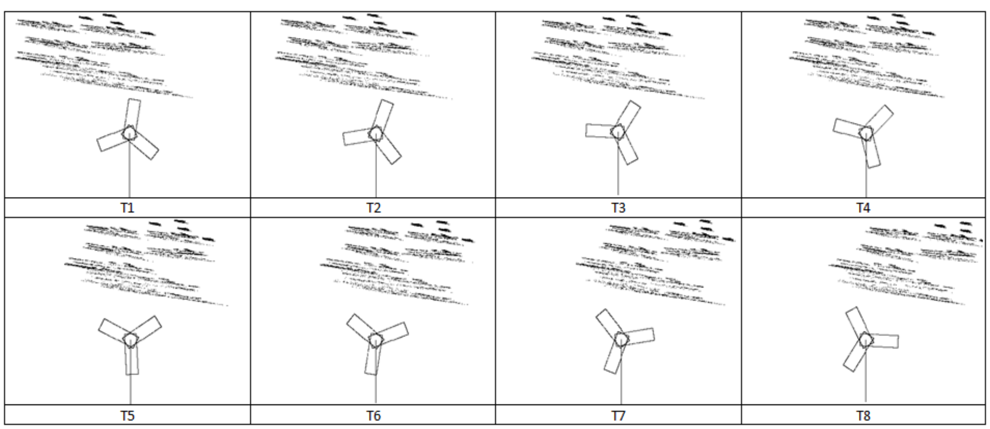


Figure 8. The transitional images of hybrid animation of windmill propeller and cloud (one pass left to right direction movement of cloud-like only without synchronization mode)

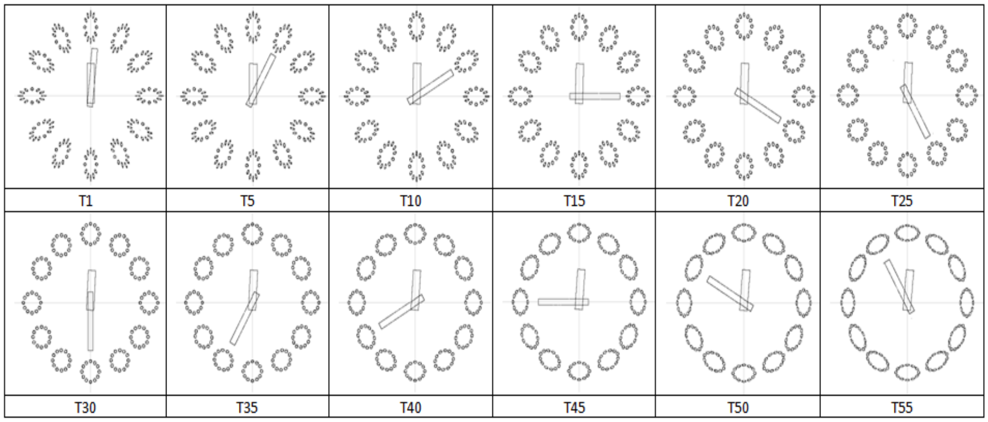


Figure 9. The transitional images of hybrid animation of the hour and minute arms of clock and the 12 ticks around (with synchronization mode between one pass rotational animation of the minute arm and one pass metamorphic animation of the ticks)

5. Conclusions

In this paper at least two things can be concluded. First, the multi-object of fractal as a new single object of fractal can be constructed and animated that consist of several objects of primitive fractal. Each primitive object of fractal as part of a new multi-object of fractal can have the different kind of animation, but as a whole represents an integrated animation of multi-object of fractal. The second conclusion is a metamorphic animation that has a special effect as a unique animation process can be combined with a non metamorphic animation such as rotational animation to make the hybrid animation model of the multi-object fractal as a whole is more attractive with or without synchronization mode.

6. References

[1] Mandelbrot, Benoit B. 1982. “The Fractal Geometry of Nature”. W.H. Freeman and Company.

[2] Hutchinson, John E. 1979. “Fractals Self-similarity”. *Indiana University Mathematics Journal 30*

[3] Barnsley, M. F., and Demko, S. 1985. “Iterated Function Systems and the Global Construction of Fractals”. Proceedings of the Royal Society of London. Series A, *Mathematical and Physical Sciences, Vol. 399, No. 1817 (Jun. 8, 1985), pp. 243-275*

[4] Oliver, Dick. 1992, “Fractal Vision: Puts Fractals to Work for You”; Sam Publishing co.

[5] Barnsley, Michael F. 1993. “Fractals Everywhere”. 2nd edition. Morgan Kaufmann,. Academic Press

[6] Megumi Takezawa, Hiroyuki Honda, Jun Miura, Miki Haseyama, & Hideo Kitajima. 1999, “A Genetic-Algorithm Based Quantization Method for Fractal Image Coding”, *0-7803-5467-2/99 ©1999 IEEE*

[7] Chang, H.T. , Chang, T.C., & Wen, K. L. 1999, “Hierarchical Fixed-Point Searching Method for Resizing Binary Fractal Image with IFS Code Modification”, *0-7803-5467-2/99 ©1999 IEEE*

[8] Abiko, Toshimizu & Kawamata, Masayuki. 1999 , “IFS Coding of Non-Homogeneous Fractal Images Using Gröbner Bases”, *0-7803-5467-2/99 ©1999 IEEE*

[9] Raynal, F., Lutton, E., Collet, P., & Schoenauer, M., "Manipulation of Non-Linear IFS Attractors Using Genetic Programming". *0-7803-5536-9/99 ©1999 IEEE*

[10] Hsueh-Ting Chu and Chaur-Chin Chen. 2000, “A Fast Algorithm for Generating Fractals”, *0-7695-0750-6/00 ©2000 IEEE*

[11] Dasgupta, D., Hernandez, G & Niño, F. 2000, "An Evolutionary Algorithm for Fractal Coding of Binary Images", *IEEE Transactions On Evolutionary Computation, Vol. 4, No. 2, July 2000*
[12] Wadströmer, Niclas. 2003, "An Automatization of Barnsley's Algorithm for the Inverse Problem of Iterated Function Systems", *IEEE Transactions On Image Processing, Vol. 12, No. 11, November 2003, 1057-7149/03 © 2003 IEEE*
[13] Gue'rin, E., Tosan, E. & Baskurt, A. 2000,"Fractal Coding Of Shapes Based On A Projected IFS Model", *0-7803-6297-7/00 ©2000 IEEE*
[14] YiShun Zhang, Yang Yang. 2001, "Displaying IFS Attractor Based on Fixed Point ", *0-7803-7010-4/01/2001 ©2001 IEEE*
[15] Chuanbo Chen,Yunping Zheng & M Sarem. 2006, "A Fractal-based Algorithm for the Metamorphic Animation", *0-7803-9521-2/06 ©2006 IEEE, pp.2957-2963*
[16] M.T. Rahman, Al-Amin, M.A. Jobayer, A.R. Chowdhury & M.A Bhuiyan, 2007, "A Novel Approach of Image Morphing Based on Pixel Transformation", *1-4244-1551-9/07 ©2007 IEEE.*
[17] Zhuang, Yi-Xin, Xiong, Yue-Shan, Liu, Fa-Yau. 2011, "IFS Fractal Morphing based on Coarse Convex-Hull". *Information Technology and Artificial Intelligence Conference IEEE Joint International 2011*
[18] Darmanto, T., Suwardi, I.S., & Munir, R. 2012, "Cyclical Metamorphic Animation of Fractal Images Based on a Family of Multi-transitional IFS Code Approach", *IEEE Conference on Control, Systems & Industrial Informatics 2012 , 878-1-4673-1023-9/12 ©2012 IEEE, pp.231-234*
[19] Darmanto, T., Suwardi, I.S., & Munir, R. 2012, "Zooming and Rotational Effects of Metamorphic Animation On Circle-like Fractal Image Based on Multi-transitional IFS Code", *International Conference on ICT for Innovative Society 2012*
[20] Darmanto, T., Suwardi, I.S., & Munir, R. 2013, "Animation Model of Multi-object in Fractal Form Based on Partitioned-random Iteration Algorithm", *International Conference on Electrical Engineering and Informatics 2013, Procedia Technology 8C (2013) pp. 96 – 101*

Tedjo Darmanto, He was born on 1957 in Bandung, Indonesia. He was obtain his Magister degree (2000) from Informatics department, Bandung Institute of Technology. Currently he still studying at School of Electrical Engineering and Informatics, Bandung Institute of Technology as doctoral student since 2011. Currently he is working as a lecturer at STMIK "Amik Bandung". His research interests are on fractals and animation.

Iping Supriana Suwardi received the docteur-ingénieur degree in Informatics from the Institute National Polytechnique, Grenoble, French in 1985. He is currently a Professor of the School of Electrical and Informatic Engineering, Bandung Institute of Technology, Indonesia.
His research interests include: information automation, dynamic computer graphic, image analysis, recognition system, and image interpretation. He has authored or coauthored over 50 published articles.
He is the inventor and the implementer of the Digital Mark Reader (DMR). DMR is software employed for evaluating examination results automatically using computer scanning. DMR iswidely used in Indonesian education institution.

He is also the inventor and the implementer of the Digital Scan Meter (DSM). DSM is software employed for reading and reporting the electric power consumption. The software convert the image photo contains numbers into the digital format for billing requirement. DSM is widely in used in the National Electrical Company in West Java, Indonesia.

Rinaldi Munir was born in Padang, Indonesia, on December 10, 1965. He received the B.Sc. degree in Engineering in 1992 from Informatics Department at Institut teknologi Bandung (ITB), Indonesia. Master's degree and Ph.D he also received at ITB, respectively in 1999 and 2010. Currently he is a lecturer in computer science at School of Electrical Engineering and Informatics, ITB.