

Paweł BAŁUKA<sup>1</sup>  
Izabela MIELNICZEK<sup>1</sup>  
Valeriia DRUZ<sup>1</sup>  
Jakub CHABKO<sup>1</sup>  
Marian KUZMA<sup>1,2</sup>

## OPTICAL FEEDBACK. I STATIC EFFECTS

The feedback is a common phenomenon. This paper gives examples of feedback in computer science (iterations). The primary aim of this paper is to observe the effects of optical feedback in a monitor-camera system. The static effects of optical feedback are presented in this paper. The problem of symmetry of the obtained images is discussed. It is explained how a multiple three-lens copying machine generates the Sierpinski triangle, which is a basic and very well-known fractal. In the next paper ( Part II ) the dynamic effects of optical feedback will be discussed.

**Keywords:** Fractals, Iteration, Sierpinski carpet, Mandelbrot set.

### 1. INTRODUCTION

The term feedback means that the same operation is performed repeatedly, i.e. the output from one iteration starts the input of the next. Processes with feedback are fundamental in all natural sciences. They were already introduced by Sir Isaac Newton and Gottfried W. Leibniz about 300 years ago in the form of the principles of dynamics. Many technical devices operate based on feedback principles. Almost all computer programs are based on feedback or iterations. Many biochemical processes in the functioning of living organisms resolve to feedback. Modelling of natural phenomena also uses feedback loops. Laws of this kind determine the position and velocity of a particle based on their values at a previous moment. The trajectory of an individual particle can be understood as the realization of such a more general law. It is also irrelevant whether the process under consideration is a discrete one.

---

<sup>1</sup> Correspondence address: Marian Kuzma, Technical College, State Academy of Applied Science in Przemyśl, Książąt Lubomirskich 6, 37-700 Przemyśl, Poland; e-mail: mkuzma@ur.edu.pl

<sup>2</sup> Institute of Material Engineering, College of Natural Sciences, University of Rzeszow, Rzeszow, 16a Rejtana Str., 35-959 Rzeszow, Poland

## 2. PRINCIPLES OF FEEDBACK

The principle of feedback is explained in Figure 1. The same operation is repeated several times, with the result of one operation being also the initial value of the next.

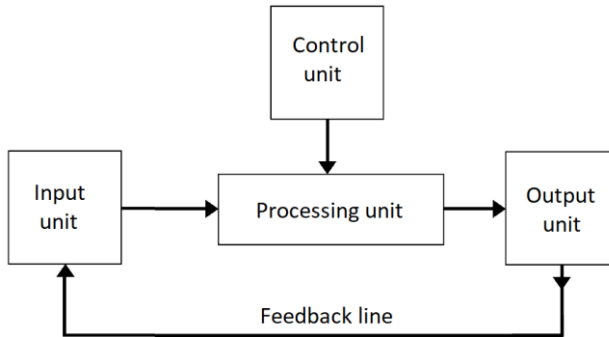


Fig. 1. Schematic diagram of a feedback system

The simplest feedback device consists of four systems:

1. input unit,
2. control unit,
3. output unit,
4. processing circuit.

These units are connected by four communication channels (see Fig. 1). The whole system is governed by a clock, which supervises what is happening in all the units and counts the cycles. We can divide the cycles into preparatory and processing cycles, each of which can be divided into basic steps:

Preparatory cycles:

1. loading information into the input unit,
2. loading information into the control unit
3. transporting information from the control unit to the processing unit.

Processing cycles:

1. transporting information from the input unit to the processing unit,
2. processing information from the input unit
3. transferring the result to the output unit,
4. loading the result from the output unit into the input unit.

Feedback is interchangeably referred to by the term iteration. The execution of a single run cycle is called a single iteration. The operation of the feedback device is started with one preparatory cycle. We then perform a certain number of run cycles.

The number of cycles may depend on the observations which are made when controlling the current output. For this reason, often another unit is included in the coupling: a comparison unit, the calibrator (Fig. 2).

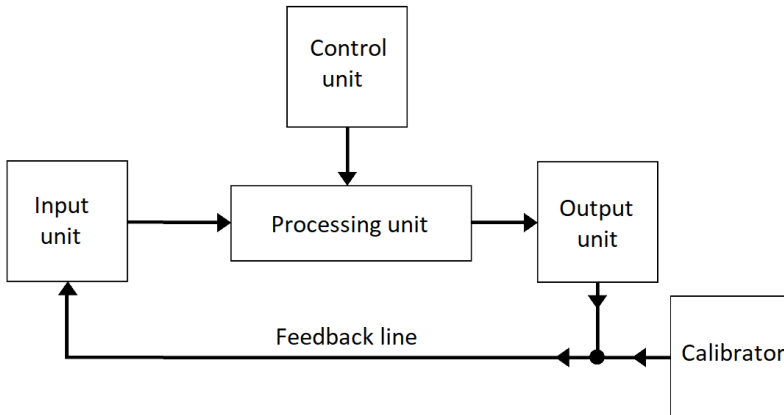


Fig. 2. Scheme of the feedback with calibration unit

### 3. EXAMPLES OF FEEDBACK IN COMPUTER SCIENCE

Iteration is widely used in computer science. In the case we are discussing, we will demonstrate the operation of iteration in the C language. This is one of the basic programming languages, which amounts to calling the appropriate commands, and getting the results in the console window.

The use of iteration helps to optimize the program. Instead of calling the same action repeatedly, we use a loop that performs it automatically. This helps to shorten the program's writing, and in many cases significantly improves its operation.

Program example 1, shows the simplest use of iteration. Figure 3a shows the source code of the program, while Figure 3b shows the operation diagram. The number of iterations is limited by the variable "height", and the "for" loop. The purpose of the "for" loop is to stop the program when it reaches a value equal to the variable "height". As a result, we get a triangle built of symbols ("\*"). Each subsequent line is a repetition of the previous one, with the value increased by 1. The effect is shown in Figure 4a.

a)

```

Program 1; „TRIANGLE”

#include <stdio.h>
int height=5;

main(){
  for(int i=1;i<=height;i++){
    for(int j=0;j<i;j++){
      printf("*");
    }
    printf("\n");
  }
  return 0;
}
    
```

b)

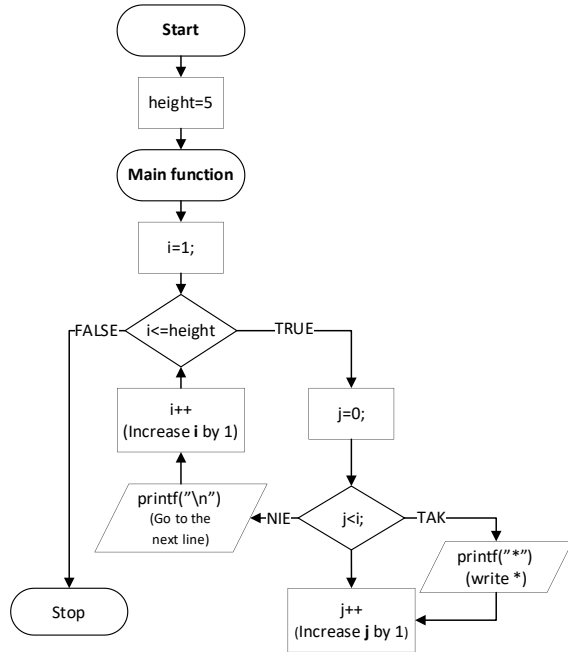
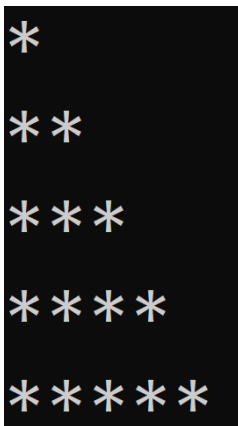


Fig. 3. Example of program 1-"TRIANGLE", source code(a), block diagram of operation (b)

Iteration in computer science can take many forms. By extending the previous program with more functions, one that draws a Christmas tree can be obtained (Fig. 4b).

a)



b)



Fig.4. The effects of : Program 1 (a); Program 2 (b)

a)

```

Program 2; „CHRISTMAS TREE”

#include<stdio.h>
int width=3;
int height=3;

main()
{
int space = width * height;
for (int x = 1; x <= height; x++)
{
for (int i = 1; i <= width; i++)
{
for (int j = space; j >= i; j--)
{
printf(" ");
}
for (int k = 1; k <= i; k++)
{
printf("* ");
}
printf("\n");
width++;
}
return 0;
}
    
```

b)

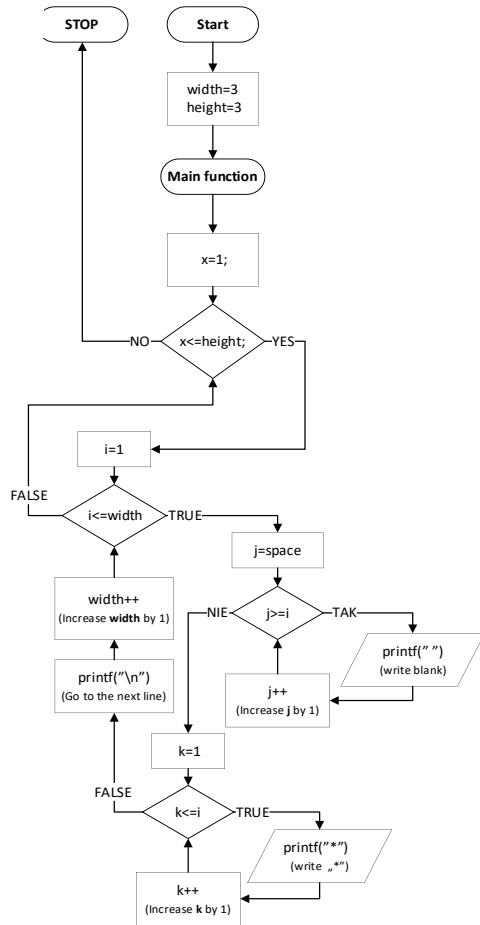


Fig. 5. Example of programme2-"CHRISTMAS TREE", source code(a), block diagram of operation (b)

Program 2 is very similar to the previous one; see Fig. 5a for the source code, and Fig. 5b. for the diagram of the operation. By extending the program with additional functions, this means that iteration is no longer limited to individual symbols. The members, which are arranged in triangle shapes, are also repeated. The effect of the operation is shown in Figure 4b.

## 4. FEEDBACK AS A GENERATOR OF FRACTALS

### 4.1. Multiple copying machine - Sierpinski fractals

The camera-monitor system can be compared to the so-called Multiple Copying Machine. An example of such a device is a photocopier, the form of the image received depends on the number of lenses used. A photocopier is built from the lens(es), and the desktop on which the drawing is placed. A copy of the drawing (image) is obtained at the output, the characteristics of which will depend on the type, and the number of lenses. As a result, an enlarged or reduced image is received. A diagram of the photocopier is shown in Figure 6.

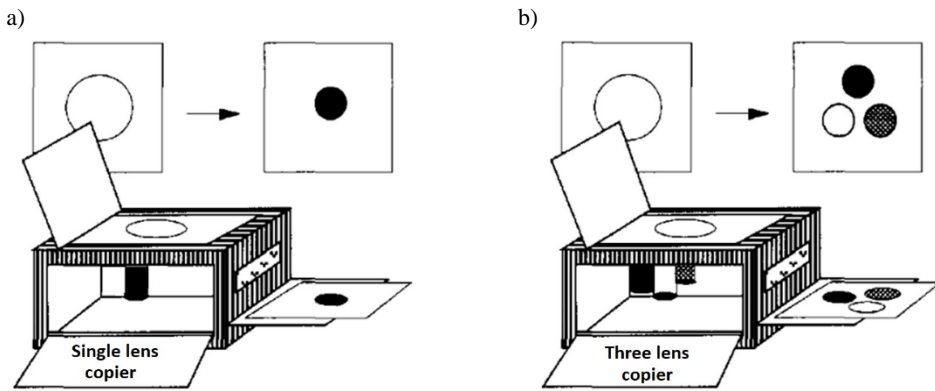


Fig. 6. A diagram of the photocopier. Single copy machine (a), multiple (triple) copy machine (b)

Placing multiple lenses in a photocopier (e.g. 3, see Fig. 6b), will result in each lens producing an image in a different part of the sheet. In addition, it may appear that the copy from each lens is of a different color. The effect of a photocopier with three lenses is shown in Figure 7b.

If the photocopier cyclically copies the last resulting copy, the effect shown in Figure 7a will be achieved. When doing so, it is necessary to remember to keep the established conditions of the enlargement (reduction).

The obtained copies decrease rapidly, however they do not fade to the point when the number of iterations increases. The relentless repetition of the process leads to the fact that the image clearly strives to achieve the form of a fractal called the Sierpinski triangle.

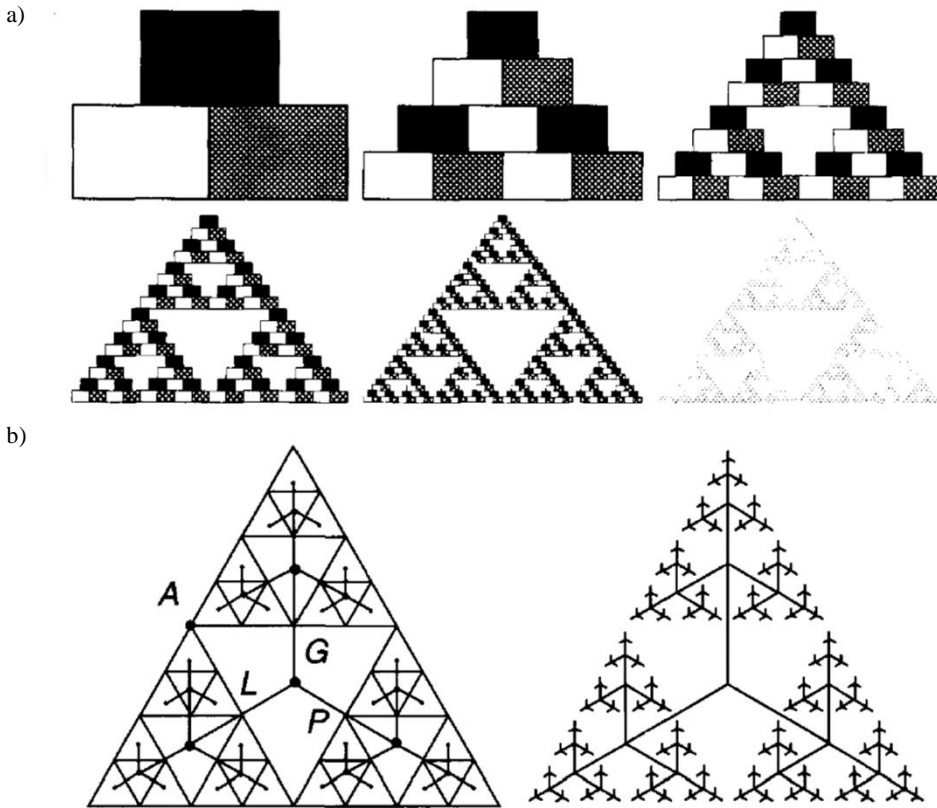


Fig.7. Initial iterations of the Sierpiński triangle construction (a); Sierpiński triangle tree (b).



In Peitgen's book [1,2], a note full of appreciation of Waclaw Sierpinski can be found:

*"The Sierpinski triangle was introduced by the great Polish mathematician Waclaw Sierpinski (1882 -1969) in 1916. Sierpinski was a professor in Lviv and Warsaw. He was one of the most prominent and influential Polish mathematicians of his time and gained worldwide recognition. Even one of the craters of the moon bears his name."*

Geometrically, the Sierpinski triangle can be obtained by removing the corresponding triangles in successive iterations starting from the third, as shown in Figure 7a. If we mark the centers of these triangles as nodes and connect them to the nodes of the next iteration, we get the Sierpinski triangle tree (Fig. 7b). The tree represents not only the structure of the Sierpinski triangle, but also its geometry. Another

example worth citing and created by Waclaw Sierpinski, is the Sierpinski carpet. It is a fractal that is formed by dividing a square into 9 parts and removing the middle one. In the next step, we again divide the resulting squares into 9 parts and remove the middle part from each one. The stages of the formation of the Sierpinski carpet are shown in Figure 8. At this point it is also worth mentioning that regardless of which shape or set of characters was used in the first step of the process, the same final figure is achieved in each case.

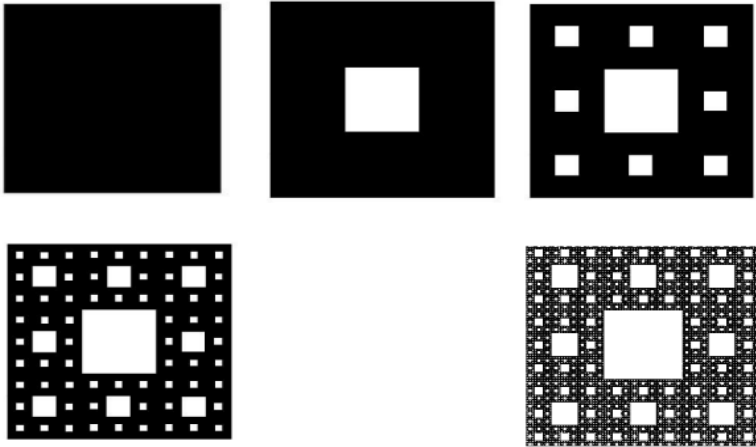


Fig.8. Stages of the Sierpinski gasket developing

## 5. MANDELBROT SET

*"The Mandelbrot set is certainly the most popular fractal, probably the most popular object of contemporary mathematics of all. Some people claim that it is not only the most beautiful but also the most complex object which has been seen, i.e., made visible. Since Mandelbrot made his extraordinary experiment in 1979, it has been duplicated by tens of thousands of amateur scientists around the world. They all like to delve into the unlimited variety of pictures which can develop on a computer screen. Sometimes many hours are required for their generation; but this is the price you have to pay for the adventure of finding something new and fantastic where nobody has looked".*

The Mandelbrot set is based on the concept of complex numbers iteration and is generated by iteration according to the formula:

$$z_{n+1} = z_n^2 + p \quad (1)$$

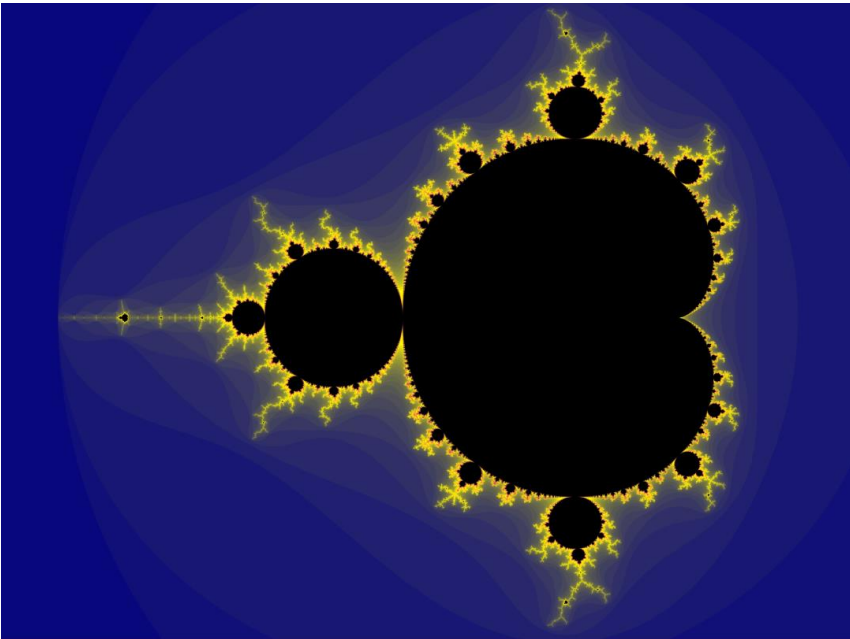
where:  $p$  is any complex number, and  $z_0 = 0$ .



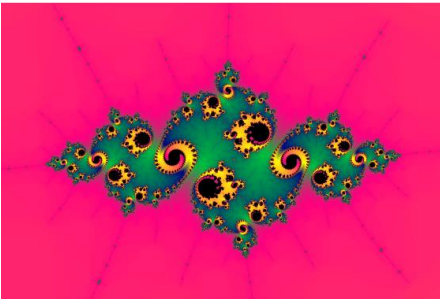
It is hard to believe that such a simple formula (1), is responsible for creating such a complicated image. This is due to iteration, because the more often the process of generating numbers according to formula is repeated (1), the more accurate the image is obtained. At the same time, it should be remembered that only values that do not go to infinity belong to the Mandelbrot set.

In Figure 9a the generated Mandelbrot set can be observed. The first thing we notice are the different colors. The colors are not chosen at random, as they are responsible for representing how quickly the values of the Mandelbrot set "escape" to infinity.

a)



b)



c)

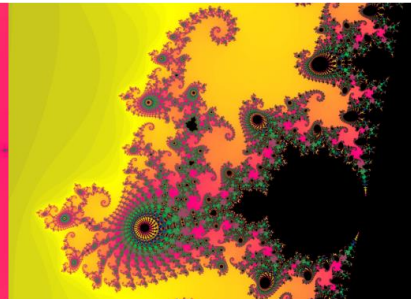


Fig. 9. The Mandelbrot set (a), element of the Mandelbrot set-"Julia set" (b), element of the Mandelbrot set -"Valley of the seahorses" t (c).

Figures 9b and 9c show characteristic elements of the Mandelbrot set. The images were obtained as a result of the enlargement of the set in suitable places. It is here that the complexity that occurs in the set can be observed. Even after the enlargement of the image obtained in Figure 9a, we continue to receive very complex objects.

## 6. THE EXPERIMENT

### 6.1. Apparatus

When talking about iteration, we should imagine a real feedback machine. In this paper, we will examine the operation of a feedback machine consisting of a computer and a video camera (Fig. 10). Such a system was described in the famous book by Peitgen, Jurgens and Saupe ( H.O. Peitgens, H. Jurgens. D. Saupe, "Introduction to Fractals and chaos" [2] ). The first versions of this experiment from the 1970s were performed on a camera-TV system [3].

The camera is placed on an optical bench. This allows for a smooth and accurate adjustment of the camera-monitor distance. The camera is connected to a computer via a USB connector. The resultant image is displayed on the monitor with the "Camera" application.

The angle of rotation of the camera is measured using a protractor, which is mounted on the same tripod as the camera. The behavior of such a machine can be controlled by many external factors. The main factors are the camera-monitor distance and the camera's angle of tilt relative to the monitor.

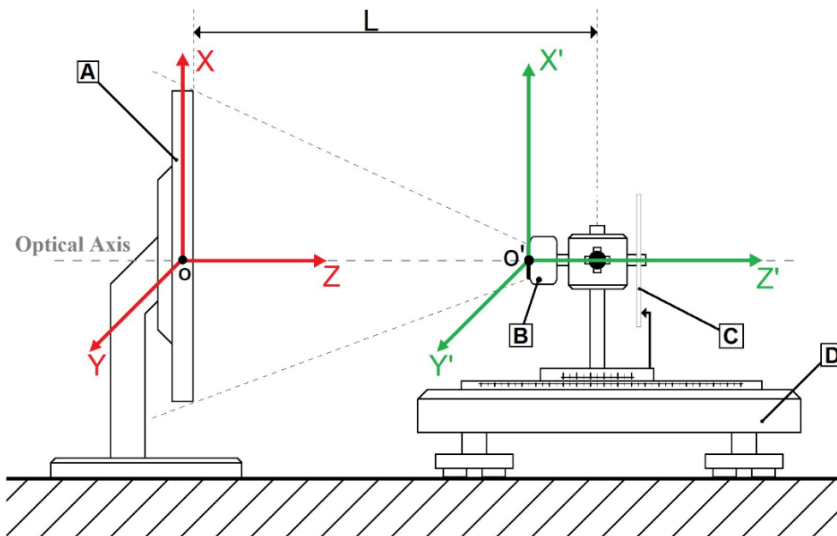


Fig. 10. Scheme of the experimental setup: A-monitor, B-camera, C-protractor, D-optical bench

## 6.2. The measurement method

The experiment should be conducted in a dark room. Before testing the optical feedback system, the camera should be positioned so that it is parallel to the monitor. We associate a stationary  $x, y, z$  coordinate system with the monitor and the bench, while a moving  $x', y', z'$  coordinate system is associated with the camera. We set the camera-screen distance  $L=L_0$ , so that the camera view covers the entire monitor (1:1). With this setting, the optical feedback effect is already observed (Fig. 11a). This can be easily seen when an object, such as a pen, is placed between the camera and the monitor (Fig. 11b). To observe other effects, the camera shall be rotated relative to the horizontal axis ( $z$ -axis), by an angle  $\alpha$ . By changing the distance of the camera from the monitor  $L_1 < L_0 < L_2$ , and the angle  $\alpha$ , respectively, different optical feedback effects can be obtained.

## 7. THE ANALYSIS OF THE RESULTS

As can be seen from the images shown in Figure 11, in all cases a specific dynamic works in the phenomenon. The result is that the "picture-in-picture" effect can be observed. This is due to the fact that each duplication of the image takes place with a certain delay. The frequency of iterations depends on the distance of the camera from the screen. The shorter the distance, the more image duplications are noticed (see Fig. 12a). On the other hand, the farther the camera is placed, the smaller the effect of "image-in-image" formation. (see Fig. 12b).

In the case where:  $L = L_1 < L_0$  only a part of the screen will appear on the screen at the first iteration (see Fig. 12a). The image is enlarged at a ratio of 1:m ( $m > 1$ ).

The resulting images strive to a certain final image, called the attractor. In our case, the number of iterations is limited, for example by the screen resolution. Often replicated and reduced in size, the image is concentrated to a very small area. Each pixel of the screen is made of three subpixels, which are intended to illuminate a suitable color of light (RGB). Superimposing of all colors (RGB), results in glowing in white color. Owing to this, the attractor is, in most cases, an area of a very bright color.

The enlargement can be interpreted in the language of dynamics as moving towards the edge of the screen with successive iteration, as can be seen in Figure 11a (this is clearly visible in Figure 11b, where the pencil placed between the screen and the monitor "wanders" to the left edge of the screen). When the camera is too close to the monitor, the evolution of the image over time can be difficult to determine.

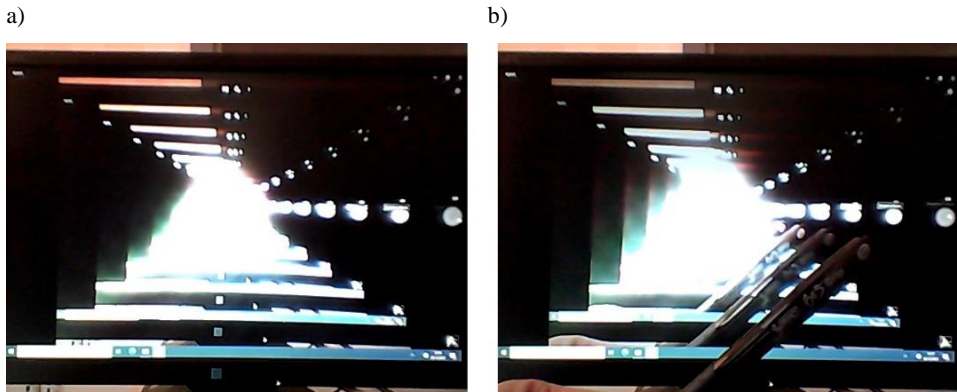


Fig. 11. The resulting image obtained for distance  $L_0=59\text{cm}$ .(a), An object (a pencil) placed between a camera and a monitor (b.)

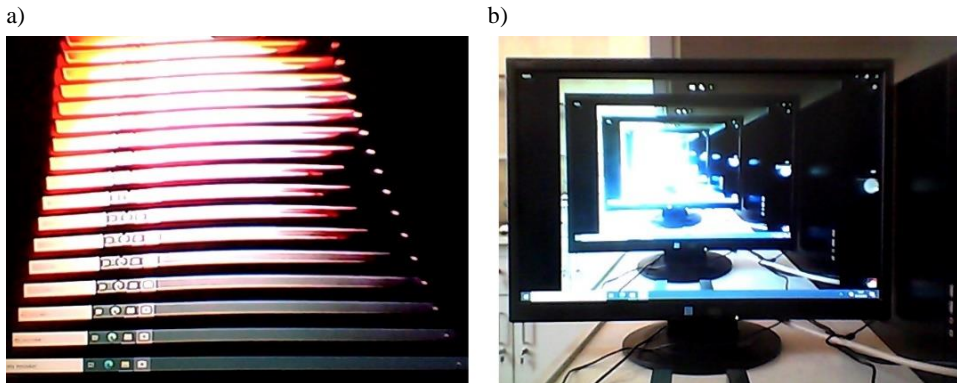


Fig. 12. Images obtained for distances:  $L_1 = 51\text{cm}$ . (a);  $L_2 = 69\text{cm}$ . (b).

On the other hand, for distances greater than  $L_0$ , the image of the monitor is reduced in the first iteration. This offers the image that is reduced. From the point of view of dynamics, this is a movement towards the center of the screen (see Fig. 12b).

Distance is not the only parameter that defines the dynamics of the iteration, which also depends on the angle of rotation of the camera. For the camera-monitor distance  $L_0$ , the most interesting effects are observed when the camera is rotated by a certain angle. The enlargement in each iteration is 1:1, i.e. the screen size should not change. We should receive images of the monitor rotated by the angle  $\alpha$  in each iteration. By increasing the angle, some symmetry in the image can be obtained.

In the case shown in Figure 13a, the hexagonal symmetry can be observed, unlike in Figure 13b, where the fourfold symmetry can be noticed.

The symmetry of the resulting dynamics is due to the fact that each duplicated image is rotated by an angle  $\alpha$  with respect to the previous one.

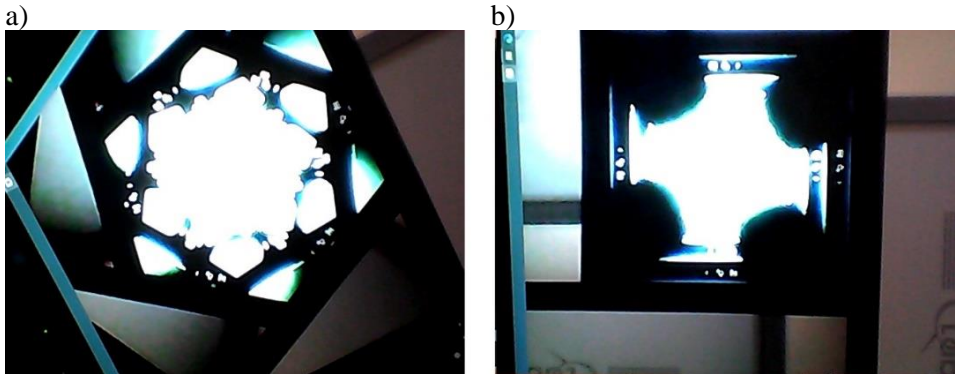


Fig.13. The image obtained for: distance  $L_o = 59\text{cm.}$ , angle  $\alpha = 65^\circ$ (a); distance  $L_o = 59\text{ cm.}$ , angle  $\alpha = 90^\circ$ (b)

In fact, for angles that are not a simple divisor of  $360^\circ$ , the resulting images are complicated and do not at all follow the 1:1 magnification rule, and the angle of rotation. In addition, a dynamic change in the shape and symmetry of the images is observed. The complication of iteration effects is due to the mechanism of image formation in the computer, and some delay.

The symmetry of images is usually  $C_n$ , it is thus a set of axes of rotation perpendicular to the plane of the drawing for  $n = 360^\circ/\alpha$  and multiples of this axis. This symmetry is described by the point group  $C_n$  [5]. The elements of symmetry which are mirrored planes are rarely seen, so the symmetry of the point group is  $D_n$ . Only in Figure 13b the symmetry of the square  $D_4$  and thus consisting of the axes of rotation:  $C_4, C_2, C_4^{-1}, I, \sigma, \sigma'$  can be seen, where  $\sigma$ , is the plane of reflection passing through the centers of the sides of the square under study, and  $\sigma'$  is the plane of reflection passing through the vertices of the square. The symbol  $I$ , on the other hand, stands for the symmetry element, inversion.

The resulting images do not always adopt certain symmetries. The formation of spirals can often be observed.

In some cases, when the camera rotation angle is too small, we can observe the formation of incomplete spirals. Such a situation is shown in Figure 14b, with parameters:  $L_1=51\text{cm.}$ , and  $\alpha=5^\circ$ . On the other hand, when the angle  $\alpha$  is large enough, we can observe the situation in Figure 14a. In this case, a spiral with certain parameters is formed.

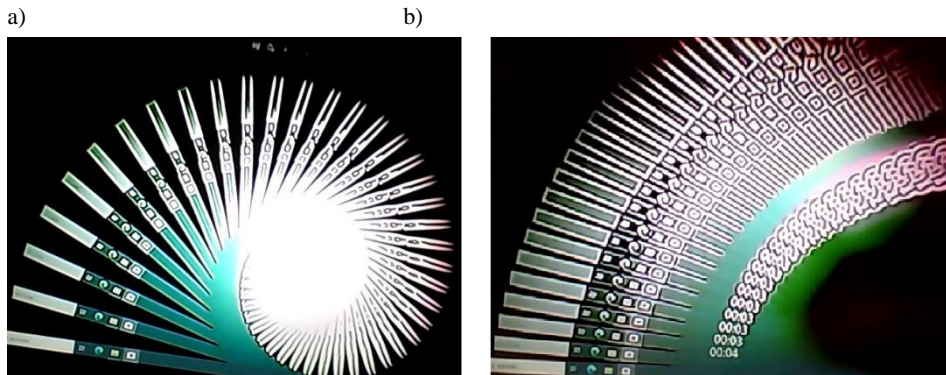


Fig. 14. The image obtained for: distance  $L_0 = 59$  cm, angle  $\alpha = 10^\circ$ (a); distance  $L_1 = 59$  cm, angle  $\alpha = 5^\circ$ (b)

The shape, and the way in which the spiral from Figure 14a develops, slightly resembles a fragment of the Mandelbrot set. This is clearly visible in Figure 15.

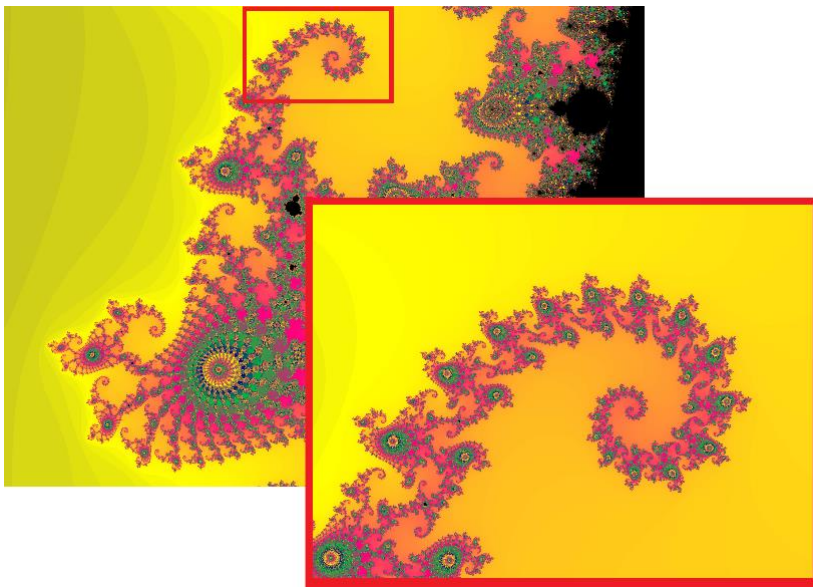


Fig. 15. Parts of the Mandelbrot set.

## 8. CONCLUSIONS

Optical feedback is a very complex phenomenon. It is characterized by a large number of possible effects. The resulting images can be described by two parameters: the distance  $L$  of the camera-monitor, and the angle  $\alpha$  of rotation of the camera relative to the monitor.

It should be noted that the apparatus for studying this phenomenon is very simple. On the other hand, the resulting effects are difficult to reproduce and almost unique. This is due to the fact that even after setting the basic parameters, still the course of the experiment depends, for example, on the prevailing light or the type of apparatus used.

On the other hand, the mathematical apparatus used to create the effects is complex (see, for example [6]). Static images are often fractals, and they can often be modeled based on the Sierpinski triangle or carpet. In contrast, the dynamic evolution of images often resembles the evolution of Gaston Julia sets [7], modeled by means of computer graphics in 1977 by Mandelbrot (born in Poland in 1924).

## REFERENCES

- [1] H.-O. Peitgen, H. Jurgens, D. Saupe, *Granice Chaosu. Fraktale*, Wydawnictwo Naukowe PWN 1995, (in Polish) .
- [2] H.-O. Peitgen, H. Jurgens, D. Saupe, *Introduction to Fractals and Chaos*, Springer-verlag New York 1992.( Polish translation, H.-O. Peitgen, H. Jurgens, D. Saupe, *Granice chaosu. Fraktale*, PWN,Warszawa 1995 see [1]).
- [3] R. Abraham, *Simulation of cascades by video feedback, in Structural Atability, the Theory of Catastrophes, and Applications*, P. Hilton (ed), Lecture Notes in Mathematics t. 525, Springer , Berlin 1975, s. 10-14.
- [4] H.-O. Peitgen, H. Jurgens, D. Saupe, *Chaos and Fractals*, Springer-Verlag New York, p. 783.
- [5] F.A. Cotton, *Chemical Applications of Group Theory*, Wiley-Interscience, N.Y. 1963.
- [6] P. Crutchfield, *Space-time dynamics in video feedback*, Physica 10D, 229-245,(1984).
- [7] G. Julia, J. Math. Pure Appl. 8, 47-245 (1918).

Internet sources:

- [8] <https://mandel.gart.nz/> – page for the generation of the Mandelbrot set .

DOI: 10.7862/rf.2023.pfe.4

Received: 11.09.2023

Accepted: 17.11.2023

Published online: December 2023