

## AUTOMATED BALANCING METHOD OF VECTOR ILLUSTRATION AND ITS SOFTWARE IMPLEMENTATION

Oleksandr AL'BOSCHIY <sup>1</sup>, Oleksandr DOROKHOV <sup>2</sup>,  
Yevhen HRABOVSKIY <sup>3</sup> and Mariya NAUMENKO <sup>4</sup>

### Abstract

The article proposes to develop a method for determining the conformity of vector illustration to the laws of composition and design, which will help to engage in artistic work for people without special skills and abilities. The concept of the centre of the composition is analysed. The main criterion of equilibrium is determined - the distance between the optical and composite centre. As the first stage of the developed method, it is offered to define the area of an element and to bring an arbitrary figure to a polygon. An algorithm for determining the area of an object of arbitrary shape has been developed. The process of colour density determination is analysed, and its algorithm is developed. An algorithm for determining the equilibrium of the composition is proposed. A script has been created that implements the method of determining the compliance of vector illustration with the laws of composition and design. A script based on a specific one has been tested during a number of tests. According to the test results, the conclusions about the correct operation of the script were confirmed. A prototype of an automation system for creating vector illustrations in Adobe Illustrator has been created.

2000 *Mathematics Subject Classification*: 52-04, 94A08

*Key words*: vector illustration, design, script, balance, centre, distance, composition.

## 1 Introduction

In modern conditions of the information society the value of aesthetic and emotional-psychological function of polygraphy production increases. In this regard, the quality of the publication is essential. The visual perception of the

---

<sup>1</sup>National Academy of the National Guard of Ukraine, Ukraine

<sup>2</sup>Simon Kuznets Kharkiv National University of Economics, Ukraine,  
University of Tartu, Estonia, e-mail: [aleks.dorokhov@meta.ua](mailto:aleks.dorokhov@meta.ua)

<sup>3</sup>Simon Kuznets Kharkiv National University of Economics, Ukraine

<sup>4</sup>National Academy of the National Guard of Ukraine, Ukraine

image depends on the correct construction of its composition. When the composition is unbalanced, it is difficult for a person to understand what the focus of the illustration is and therefore he can no longer correctly perceive the idea of the publication.

In scientific papers [1, 5, 10, 4], various approaches are presented regarding the visualization of the presentation of information in electronic publications. The issues of algorithmic processing of illustrations are analysed in articles [9, 6, 7]. Key aspects of the gesture interface in a virtual environment are discussed in [11]. Scientific articles [3, 8] suggest approaches to optimize the download speed of vector illustrations and the quality of their processing.

However, the specialized literature lacks a comprehensive technique for automating the balancing of vector illustrations. The aim of the research is to develop a method for determining the conformity of vector illustration to the laws of composition and design, which will help engage in artistic work for people without special skills and abilities.

## 2 Materials and methods of research

The following research methods were used to create a method of automated balancing of vector illustrations. The method of synthesis was used to form a method for determining a balanced composition, selecting a suitable editor, and implementing a script. The method of analogy was used to develop a criterion for compliance with the concept of "balanced composition" for a vector editor.

The generalization method was used to identify differences and features of use to solve the problem for the selected vector editor. The empirical method was used to implement the script and test performance. The method of mathematical modelling was used to solve the problem of developing a mathematical model of a balanced composition. The method of abstraction was used to determine the characteristics on which the equilibrium of the composition depends.

## 3 Research results

Creating a work of art in any field of art is impossible without compositional construction. After all, composition is the most important means of building a whole. With the help of the composition the inseparable unity of content and form is realized.

Therefore, the construction of a harmonious composition is a priority in the creation of any illustrative material. After all, it is the illustrative material that creates the first impression on the reader. The balance in the composition is directly related to the concept of center. Being a complex structured system, the composition involves the presence of several types of centers.

In the composition there are the following centers: *Geometric*, *Optical*, *Plot*, and *Logical*. The geometric center is the point of intersection of the diagonals of

a rectangular image. It is not the result of compositional construction but affects the structure of the composition.

If the "center of gravity" of the object coincides with the geometric center of the image, the composition is generally perceived as stable, static, unchanging. The center of gravity can be not only the compositional center, but also the axis of symmetry about which the objects are located.

If the compositional (center of mass) and optical center do not coincide, then the graphic composition is not balanced.

To determine the quantitative characteristic, how exactly the composition is balanced, it is necessary to compare these two centers, to determine the distance between them as shown in Figure 1.

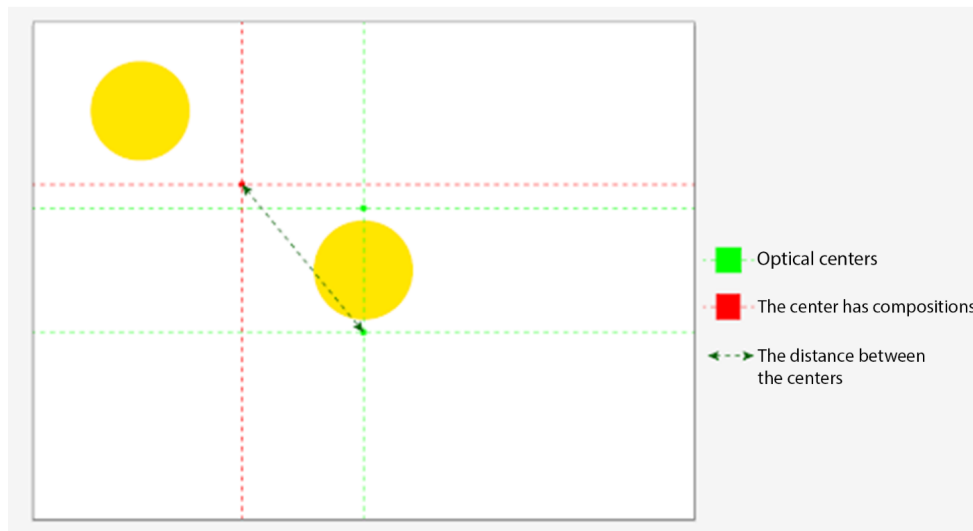


Figure 1. Balance criterion -the distance between optical and composite center.

The *optical center* is a point about one-eighth the height of the image above the geometric center, or five-eighths of the distance from the bottom edge of the composition. If the structural element of the composition is placed in the optical center, it will be read first.

If, on the contrary, it is necessary to create a semantic pause, to delay the speed of perception of a certain element - the object is removed from the optical center.

Thus, the ratio of the location of the elements of the composition to the optical center is a factor influencing the linearity or non-linearity of the perception of the composition.

The *plot center* contains the main or important story element or group of elements.

The *logical center*, which is considered to be the plot center - is an element that stands out due to the logic of the composition.

The *mass center* is a focusing place around which the elements are united on the principles of subordination, it corresponds to the main part of the composition and can coincide with the geometric center of the scheme. The mass center makes it possible to create the integrity of the compositional links of the elements.

The *gravity center* is created by formal elements of the image: dots, strokes, spots, color, textures, pointers, lines, etc.

The compositional center is not always a point, it can be represented by a certain extended area. The gravity center is a structural unit around which and in relation to which all the elements of the composition are built.

The gravity center is an element of visual composition that reads primarily through the construction of the composition; conditional center of gravity of the composition; the element which is allocated on the form, operability, carries in itself semantic loading.

It should also be taken into account that at a specific value of this distance, the composition is no longer balanced, so on the basis of empirical research it is necessary to determine the range of acceptable values for this criterion.

The area of admissible values will be considered the distance from the geometric center, i.e., it will have the shape of a circle.

Having determined the main criterion for assessing the balance, you can develop a method according to which the future script will perform its work.

That is, the developed technique consists of the following steps: *Determination of the optical center of the composition, Determination of the center of mass of the composition, Calculation of the distance between the centers, Equilibrium of the composition, Determination of the optical center of the composition.*

Based on the fact that the optical centers are above and below the physical center by about one-eighth of the height of the image, or at a distance of five-eighths (three-eighths) from the lower edge of the composition, you can determine the coordinates  $x, v_1, y_2$ :

$$x = b/2, v_1 = (a * 3)/8, y_2 = (a * 5)/8, \quad (1)$$

where  $a$  is the height of the leaf;  $b$  - sheet width.

To calculate the center of mass, you need to calculate the area of the element. Although in most cases Simpson's method for determining the area is much more accurate, in this case you need to use the trapezoidal method. That is, the first step in determining the area of the element by this method is to bring an arbitrary shape to a polygon. To implement this step, you need to know all the points on the contour of the figure, and then connect them.

The more points on the contour you know, the more accurate you will be to approximate the figure to the polygon as shown in Figure 2.

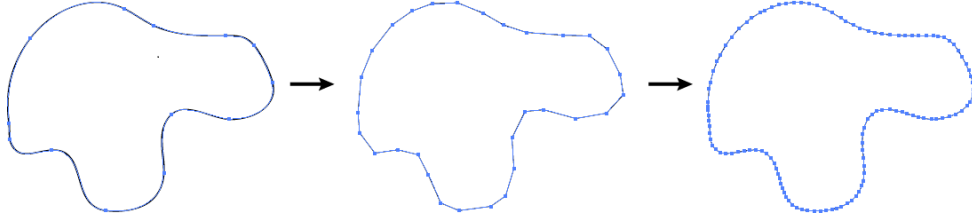


Figure 2. Bringing a figure of arbitrary shape to the landfill.

The figure shows three cases: on the left - Original image; in the center - Image converted to polygon with 30 number of points; on right - Image converted to polygon with 300 number of points. You can define points on a contour by solving the function of the *Bezier* curve, because in the vector editor all figures are described by a certain formula of the *Bezier* curve. So, knowing this formula, you can easily calculate the coordinates of all points on the curve and write them to the array. From which they will then be used to determine the area of the element. In order to automate the construction of a figure of arbitrary shape in a vector editor, we use the method of dividing the figure into primitives [2].

The algorithm for determining the area of such a figure is shown in Figure 3. The formula for finding the area of the figure will look like this:

$$S = \left| \sum_{i=1}^n \frac{y_{i+1} + y_i}{2} \cdot (x_{i+1} - x_i) \right|, \quad (2)$$

where  $y_i$  is the coordinate of a point on the curve along the  $y$ -axis;  $x_i$  is the coordinate of a point on the curve along the  $x$ -axis. It should be especially noted that the points  $(x_i, y_i)$  are ordered points on the curve and we have  $(x_{n+1}, y_{n+1}) = (x_1, y_1)$ . Also it is necessary to mention that this formula gives an approximation value of the area. Calculations of the coordinates of the points of the corresponding figures according to mathematical formulas are used to construct vector graphics of straight and curved lines, as well as standard geometric figures. Equations describing the so-called *Bezier* curve are used to construct the line curve. In the vector editor, *Bezier* curves are described by four points: two engagement points and two control points (right and left directions).

The engagement points can be called  $(x_1, y_1)$  and  $(x_2, y_2)$ , and the control points -  $(x_3, y_3)$  and  $(x_4, y_4)$ , which are described by equations:

$$x(t) = (a_x \cdot t^3) + (b_x \cdot t^2) + (c_x \cdot t) + x_1, \quad (3)$$

where

$$a_x = x_4 - x_1 - c_x - b_x, b_x = 3 \cdot (x_3 - x_2) - c_x, c_x = 3 \cdot (x_2 - x_1), t \in [0; 1]. \quad (4)$$

$$y(t) = (a_y \cdot t^3) + (b_y \cdot t^2) + (c_y \cdot t) + y_1, \quad (5)$$

where

$$a_y = y_4 - y_1 - c_y - b_y, b_y = 3 \cdot (y_3 - y_2) - c_y, c_y = 3 \cdot (y_2 - y_1), t \in [0; 1]. \quad (6)$$

The algorithm on Figure 3 is to determine the area of an object of arbitrary shape.

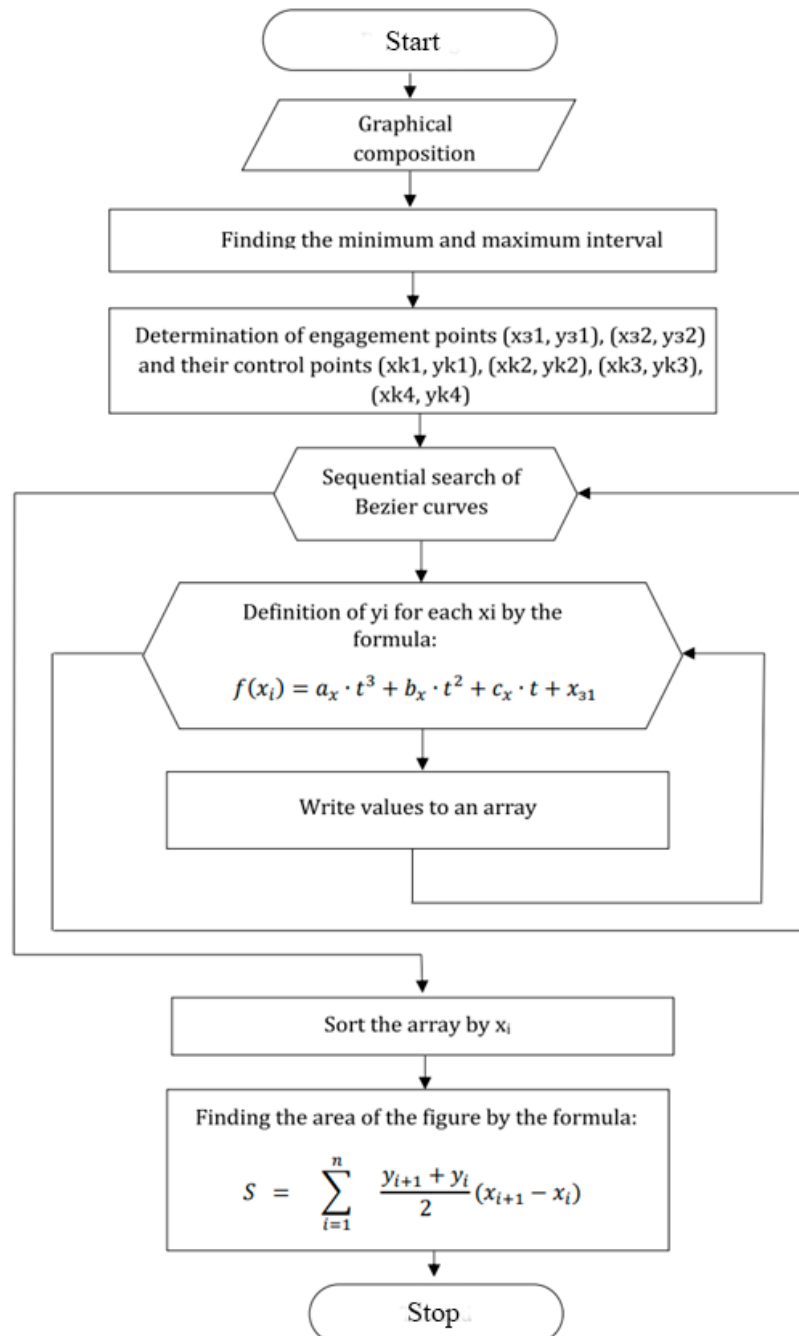


Figure 3. Algorithm for determining the area of an object of arbitrary shape in the graph.

This algorithm consists of the following steps:

- 1) Defining points on the contour of the figure.
- 2) Bringing the figure to the landfill.
- 3) Determining the area.

Algorithm's iterations can be stopped at the end of sorting the array by  $x_i$ .

Determining the color density of a vector illustration element is complicated by the fact that in this case it is impossible to calculate the color components of each pixel. This is because in vector editors the object is characterized not only by the color of the fill, but also the contour.

There are also several options for filling objects: *Homogeneous fill*, *Gradient* and *Pattern fill*.

You can also apply a clipping mask to objects (an object or group of objects whose shape masks the display of objects below). If in the case of gradient fill it is still possible to determine the color density by calculating the average value of colors on the gradient scale, then in the case of fill pattern, or when using a clipping mask - it is impossible to do for a vector image.

Therefore, to correctly determine the color density of an element, it is necessary to rasterize it, and then determine the color density of each pixel and find the total density of the object. This calculation will allow you to determine the center of mass more accurately and, ultimately, to balance the composition.

Determination of pixel color density is determined by the following formulas:

$$p_{rgb} = \frac{R + G + B}{765}, \quad (7)$$

where  $R, G$  and  $B$  are the color components in the  $RGB$  model,  $765 = 255 + 255 + 255$  - the sum of the values of each color in the  $RGB$  model, measured from 0 to 255;

$$p_{cmk} = \frac{C + M + Y + K}{400}, \quad (8)$$

where  $C, M, Y$  and  $K$  are the color components in the  $CMYK$  model,  $400 = 100 + 100 + 100 + 100$  is the sum of the values of each color in the  $CMYK$  model, measured from 0 to 100.

Thus, the process of determining the color density looks as shown in Figure 4.

The aim of the algorithm shown on Figure 4 is to correctly determine the color density of the element in the graph.

Iterations of the algorithm can be stopped when, as a result of sequential search of the composition elements, all generated points will correspond to the contour points of the object.

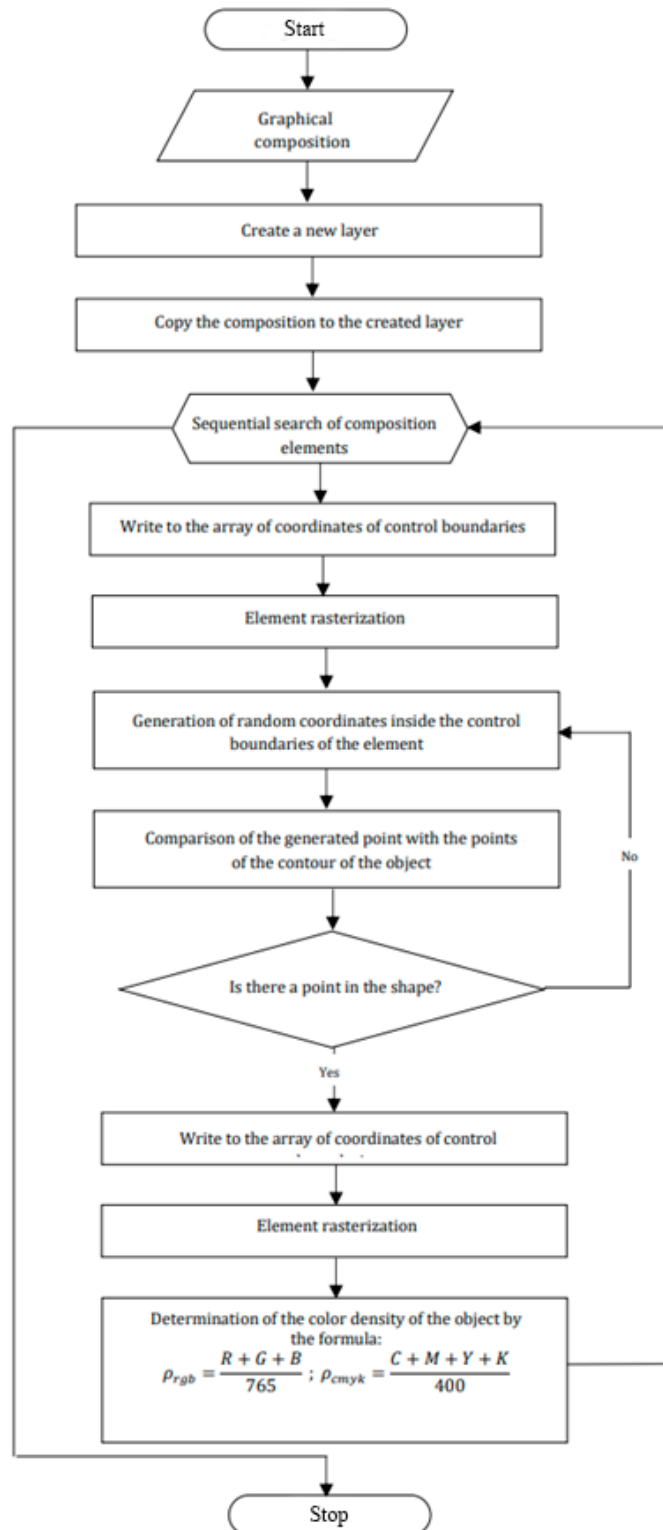


Figure 4. Algorithm for determining color density.



Let us consider calculating the distance between centers.

If you imagine that the center of mass and the geometric center are just two points on a plane in the *Cartesian* coordinate system, then the distance between them is as follows:  $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ , where  $(x_1, y_1)$  and  $(x_2, y_2)$  are the coordinates of the center of mass and the optical center.

Determining the distance between the centers is a criterion for the balance of the composition. That is, this is the final stage of calculations in the developed method. After that, the next step is its direct balancing.

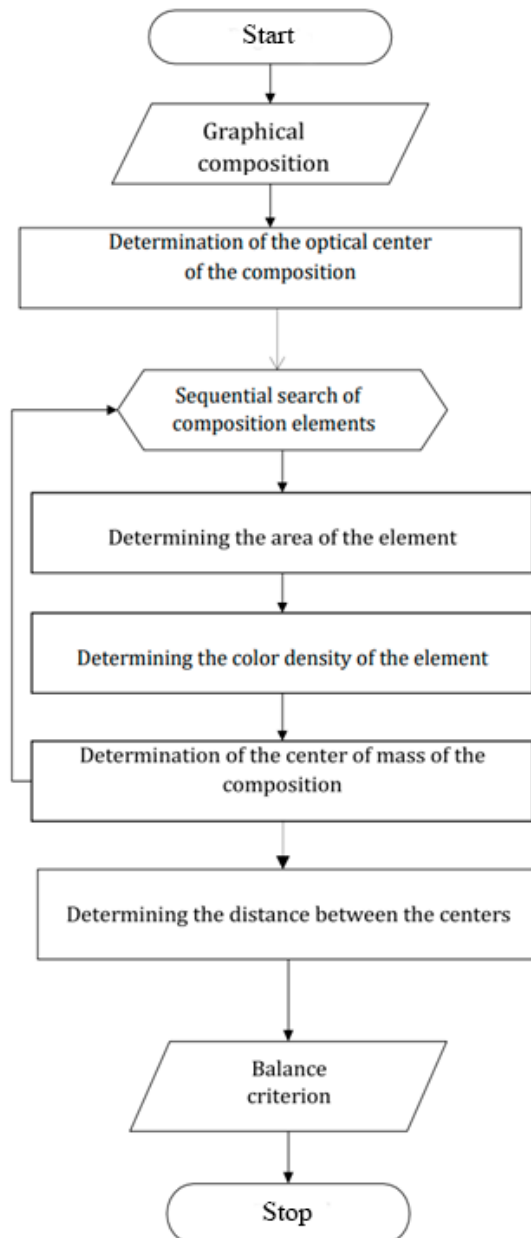


Figure 5. Algorithm for determining the equilibrium of the composition.

Thus, the general algorithm of the developed method is given in Figure 5. The aim of the algorithm (Figure 5) is achieving balance of the composition in the graph. Iterations of the algorithm can be stopped when the distance between the centers of the composition will meet the equilibrium criterion.

This algorithm reflects the developed method, which not only analyses the composition, but also, based on calculations, makes it balanced. That is, because of the implementation of this method, the initial image is balanced.

According to the results of the analysis, it was found that one of the most important principles of harmonious composition, which should be followed, is the principle of equilibrium. But it is very difficult to build such a composition for people who do not have the knowledge and skills in composition and design.

The capabilities of modern vector editors are not enough to create an interesting balanced composition. Therefore, the future system should use the principle of randomness in finding suitable solutions.

To accomplish this task, it is advisable to create a script that would solve the following problems: Evaluation of the composition for balance, Random movement of the elements of the composition to harmonize it.

Thus, this automated system will help to engage in the visual work of people without special skills and abilities, reduce the time for such a difficult and troublesome operation as building a composition and will bring an element of randomness and diversity to this process. The main function of the script, which calls all other functions (code snippet) is the next:

```
function createNewGeneration()
{
var i = 0;
createPopulation();
while (population[0].distance > 5 && i < iterationCount )
{
mutation();
crossover();
createPopulation(); sortByDistance();
i++;
}
```

It sequentially calls functions until the distance between the centers is greater than a given number and a given number of iterations.

After the script has generated the appropriate coordinates for the selected elements, the assignment of new coordinates is performed using the next function:

```
function createPopulation()
{
var size = populationSize - mayDie;
for (var i = mayDie; i < size; i++)
{
var items = new Array(pathItemCount);
for (j = 0; j < pathItemCount; j++)
```

```

{
items[j] =
{
left: getLeft(Math.random() * doc.width,
doc.pageItems[j].width),
top: getTop(Math.random() * doc.height,
doc.pageItems[j].height),
width: doc.pageItems[j].width,
height: doc.pageItems[j].height,
  x: 0,
y: 0,
mass: getMassItem(doc.pageItems[j])
};
items[j].x = getCxItem(items[j]);
items[j].y = getCyItem(items[j]);
}
population[i] =
{
items : items,
distance : getDistance(items)
};
}
mayDie = populationSize;
}

```

This function creates new populations. At the first pass, it creates an array of elements, where it writes all the parameters of the elements of the composition. In subsequent iterations, it creates new populations of coordinates of these elements as shown in:

```

left: getLeft(Math.random() * doc.width,
doc.pageItems[j].width),
top: getTop(Math.random() * doc.height,

```

They are formed randomly, so another function was created, which described the conditions so that the elements do not fly beyond the edges of the sheet as shown in:

```

function getLeft(newLeft, width)
{
if (newLeft + width > doc.width) return doc.width - width;
return newLeft;
}
function getTop(newTop, height)
{
if (height - newTop > 0) return height;

```

Also, this function calculates the distance between the optical center and the center of mass. Distribution of the object into intervals also use this fragment of code). The function that returns the the optical center coordinates is shown in:

```
var opticalCenter =
{
x: doc.width/2,
y: doc.height*3/8
```

To calculate the center of mass, you must calculate some data, and the first is the area of the element. According to the method described above, determining the area consists of several stages is shown in Figure 6.

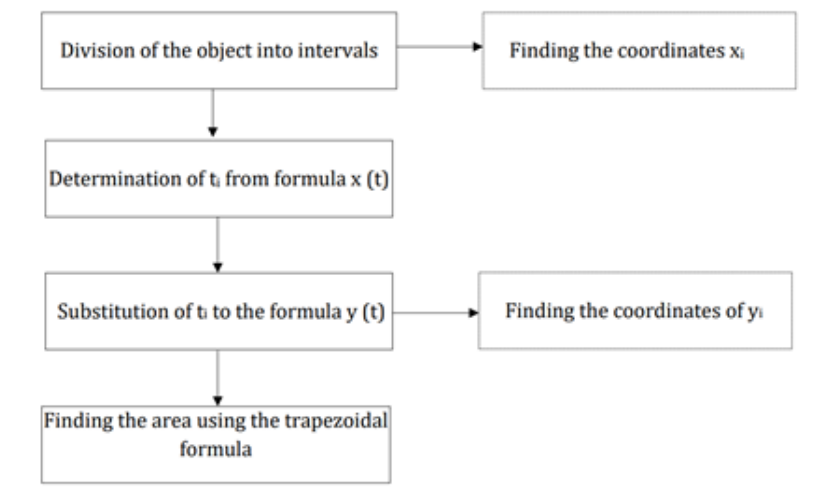


Figure 6. Algorithm for determining the area of the element.

Finally, a code snippet, used to determine the color density of an element, is the next:

```
function getRoItem(item)
{
if (doc.documentColorSpace == DocumentColorSpace.RGB)
return (item.fillColor.red + item.fillColor.green
+ item.fillColor.blue)/300;
if (doc.documentColorSpace == DocumentColorSpace.CMYK)
return (item.fillColor.cyan +
item.fillColor.yellow +
item.fillColor.magenta +
item.fillColor.black)/400;
}
```

Determination of color density solves the problem of imbalance of the composition.

## 4 Experimental results

The main results of the experimental work of the script, that is, the distance (in centimeters) between the optical center of the graphic composition and the center of mass. Testing of the script is based on a certain number of tests, according to these results then conclusions are made about the correct operation of the script.

Table 1.

Experimental results data (distance in centimeters) between the optical center of the graphic composition and the center of mass  
(for script initial testing / for script validation, secondary testing)

Test number	1		2		3		4		5	
Iteration number										
1	77,3	126,5	49,7	74,5	110,0	88,2	92,6	55,4	90,5	93,5
2	73,4	98,1	32,6	74,5	80,5	82,5	91,9	55,1	90,3	68,3
3	70,0	91,0	31,9	74,5	76,2	77,1	86,3	54,3	90,1	64,3
4	65,4	83,7	25,6	67,3	75,5	76,5	80,5	54,2	89,9	59,8
5	65,2	83,7	24,8	67,2	75,0	71,4	79,9	46,9	84,3	55,6
6	62,4	76,4	24,6	60,0	74,8	68,2	74,4	46,6	78,9	55,1
7	61,7	76,1	18,1	59,2	74,8	63,3	71,0	46,1	78,589	51,4
8	61,7	75,6	17,8	58,5	69,7	59,4	65,6	38,8	78,3	47,8
9	65,9	68,9	12,1	58,5	69,7	54,7	62,1	31,5	73,3	45,2
10	56,4	68,1	8,9	57,7	64,7	50,5	56,7	30,4	68,7	40,3
11	53,9	61,0	4,0	51,3	64,1	46,2	51,9	30,1	68,2	38,6
12	51,1	60,7		50,2	59,0	42,4	51,4	22,8	64,1	38,4
13	48,4	53,6		42,9	54,0	41,5	47,6	22,8	63,6	38,1
14	46,5	46,2		42,1	49,0	37,5	47,3	22,8	59,7	38,1
15	46,0	38,9		34,9	48,8	33,0	43,0	21,9	59,1	37,5
16	45,8	38,9		34,8	43,8	32,6	38,8	14,6	54,6	37,4
17	41,5	31,6		34,8	43,6	28,4	38,65	8,2	54,6	37,2
18	41,1	30,8		27,5	43,5	24,5	34,7	8,2	54,2	37,2
19	37,9	23,5		20,4	42,2	20,0	30,1	7,9	54,0	36,6
20	37,3	22,7		13,6	42,0	15,8	26,0	7,9	54,0	31,5
21	37,0	15,4		12,5	41,8	15,2	22,2	7,6	54,0	26,8
22	36,8	15,1		11,7	41,6	11,8	17,5	1,2	53,8	26,3
23	34,9	14,8		4,4	36,6	6,9	17,0		49,6	21,1
24	32,1	14,8			36,6	6,9	17,0		45,6	16,0
25	27,4	14,8			36,2	6,6	13,0		42,4	15,9
26	27,1	7,5			36,2	2,3	9,4		41,8	15,9
27	24,6	0,2			36,1		4,4		41,3	10,8
28	24,4				36,1				39,7	5,7
29	24,0				30,9				36,5	5,1
30	19,3				30,9				36,1	0,9

The created script solves the problem of composition imbalance by comparing the optical center and the center of mass. Also, at each iteration of the script, the distance between these centers is measured for each population of 20 given. The script terminates when this distance is 5 points, or the number of specified iterations has reached the end. Given all this, you should do 2 times 10 tests and set 30 iterations for each test. Table 1 illustrates the first 5 tests. During the second test, the distance decreased rapidly, so the script passed only 11 iterations.

According to these results, you can create graphs that will better see the effect of the script and reduce the distance between the optical center and the center of mass of the composition as shown in Figure 7.

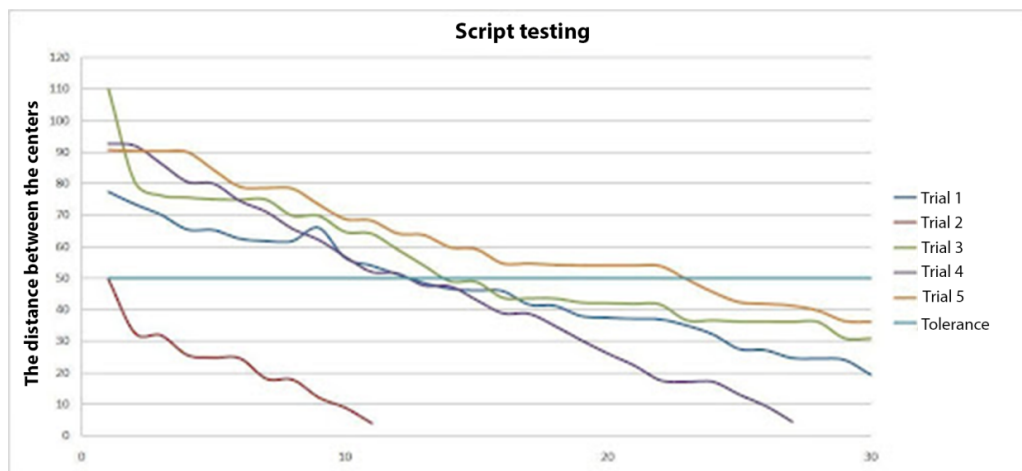


Figure 7. Script testing (according to Table 1 data).

Figure 7 shows that, in each test, the goal was achieved - to make the composition balanced - because at the end of the script, the distance between the centers was less than 50 points.

And on the second and fourth tests, the distance of less than 5 points was reached much earlier than the iterations were set.

Also, results for the evaluation of the correctness of the script are more clearly shown in the form of a graph shown in Figure 8.

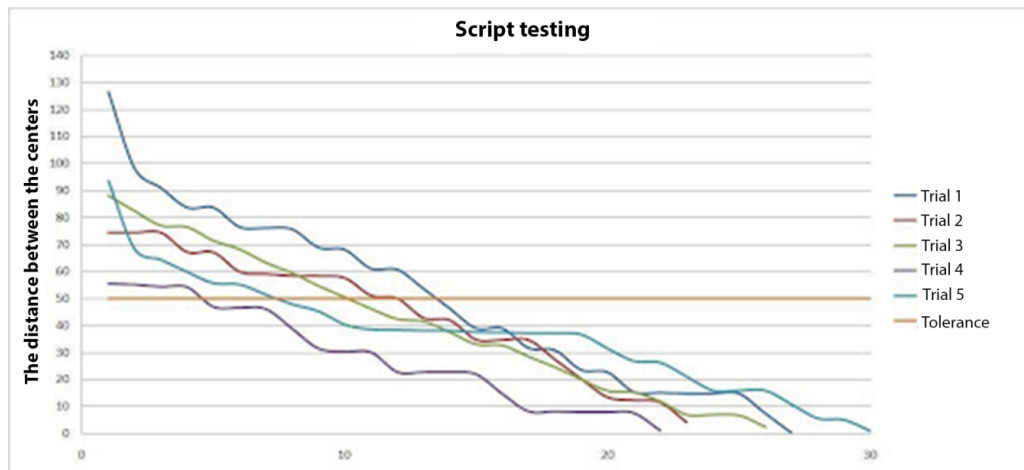


Figure 8. Script validation (according to Table 1 data).

These graphics clearly illustrates the decrease in the distance between the centers during the passage of the script.

This indicates that the script is working correctly. So, Figure 9 shows the resulting balanced composition.

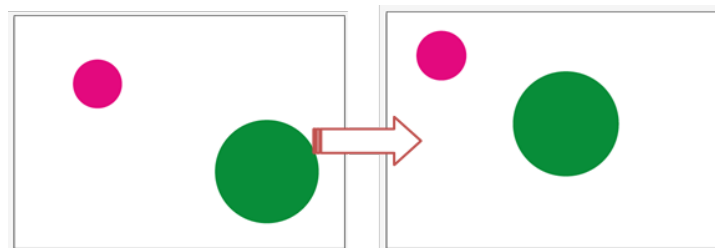


Figure 9. Resulting balanced composition.

Thus, a mathematical model of composition balancing has been described.

In the framework of which the analysis of the concept of "equilibrium", algorithms of stochastic optimization and determination of the area of arbitrary figures was carried out. It should be noted that compositional equilibrium is easier to achieve when designing symmetrical shapes, because the axis of symmetry already creates the preconditions for such equilibrium, although it does not guarantee it.

## 5 Conclusions

Thus, in this work an empirical study was conducted, which proves the importance of using the principle of equilibrium in creating illustrative material. The tendency to adhere to the principle of balance in most modern masterpieces of art suggests that this principle plays an important role in creating compositions.

Accordingly, an equilibrium criterion was proposed to assess how balanced the composition is. During the implementation of research methods, a mathematical model of balanced composition and a technique for balancing a vector image were developed.

A prototype of a system for automating the creation of vector illustrations in Adobe Illustrator was created. This is achieved through the introduction of genetic algorithms, which are based on the principles of stochastic optimization.

## References

- [1] Gardiner, E., *The Electronic Book*, Oxford University Press, 2010.
- [2] Harris, J. and Withrow, S., *Vector graphics and illustration. A Master class in digital image-making*, RotoVision, 2008.
- [3] Hood, N., *Quality in MOOCs : Surveying the terrain*, Burnaby: Commonwealth of Learning. **4** (2016), 54-74.
- [4] Hrabovskyi, Y., Brynza, N. and Vilkhivska, O., *Development of information visualization methods for use in multimedia applications*, EUREKA: Physics and Engineering. **1** (2020), 3-17.
- [5] Lupton, E., *Thinking with type: A critical guide for designers, writers, editors, students*, Princeton Architectural Press, 2007.
- [6] Martin, R., *Twenty challenges for innovation studies*, Science and Public Policy. **43** (2016), no. 3, 432-450.
- [7] Mitchell, D. and Ream, R., *Professional responsibility*, Springer, 2014.
- [8] Pushkar, O., Hrabovskyi, Y. and Gordyeyev, A., *Development of a method for optimizing the site loading speed*, Eastern-European Journal of Enterprise Technologies **6** (2020), no. 2, 21–29.
- [9] Safonov, I., Kurilin, I., Rychagov, M. and Tolstaya, E., *Adaptive image processing algorithms for printing*, Heidelberg Press, 2018.
- [10] Saylor, M., *The mobile wave: How mobile intelligence will change everything*, Vanguard Press, 2012.
- [11] Vultur, O. M., Pentiuc, S. G. and Lupu, V., *Real-time gestural interface for navigation in virtual environment*, Development and Application Systems **5** (2016), 303-307.