APPLICATION OF SELF-ORGANIZING MAPS IN GRAPHICS TRANSFORMATION

ARTUR BARTOSZEWSKI[1]

Abstract

The article was presented non-standard use of self-organizing map (SOM). They were used to find the characteristic points of the image. This allows you to automate the process of finding the key points of images the needs of the morphing, that is, animation - the conversion of one image to another.

The author presents an algorithm, and the results of example implementation of this algorithm.

Key words: Artificial neural network, SOM, morphing, traingularization

1. Introduction

Artificial neural networks can be thought of as a modern computing systems, which allow the processing of information by imitating the processes in the human brain. Information entered into the network are numerical data based on which network performance may reflect the operation model of a completely unknown characteristics [6]. Adaptation of the neural network to solve a specific task is done by the training using the typical impulses and the corresponding desired response rather than by specifying the algorithm and saving it as a program.

The algorithm of self-organizing map (SOM), developed in 1982 by Teuvo Kohonen [4] is one of the most advanced models of neural networks, which are used in many different fields of science, which, for example, include:

- econometric analysis;
- classification and analysis of data from environmental monitoring [1];
- processing of survey data [3];
- text search of documents in response to a query formulated in the search [5].

Application of SOM algorithm is appropriate wherever there is a need for mapping multi-dimensional space on a plane. With the ability to self-organizing Kohonen maps allows you to adapt to previously unknown data about which little is known. This property of SOM network was used in the creation of a program that generates a mesh for morphing. Creation of such a mesh consists in indication of specific image points and is typically performed by humans. Because of their ability self-learning neural network can automate this process.

[1] Kzimierz Pulaski University of Technologies and Humanities in Radom

2. Traingularization and morphing

Morphing is a special effect in motion pictures and animations that changes one image into another through a seamless transition. Morphing software continues to advance today and many programs can automatically morph images that correspond closely enough with relatively little instruction from the user. This has led to the use of morphing techniques to create convincing slow-motion effects where none existed in the original film or video footage by morphing between each individual frame. Morphing is used far more heavily today than ever before. In years past, effects were obvious. Now, morphing effects are most often designed to be invisible.

Morphing is an attempt to find the intermediate state, or more likely, a series of intermediate states between two objects. Animation involves finding a weighted intermediate states depending on time t.

Objects which are transformed, can be described using a two or three-dimensional vectors (two-dimensional points in space or three-dimensional)[8]. However, concept of morphing often refers to the transformation of raster images. For raster image morphing is an attempt to find an intermediate state between two objects that are shown in the pictures. Morphing is not only the search of intermediate states between the two images, but the images of objects in between. The difference between finding an intermediate image (cross-dissolving) and looking for an image of the intermediate object (morphing) is shown in Figure 1.

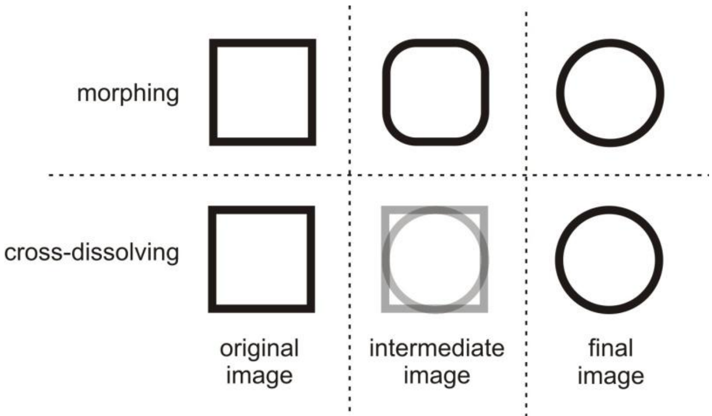

Figure 1. The difference between finding an intermediate image (cross-dissolving) and looking for an image of the intermediate object (morphing)

Morphing is a combination of dissolving (cross-dissolving), tweening, and warping.

Dissolving (cross-Dissolving) is a combination of two images with smooth transition - averaging. Tweening is the process of indirect interpolation between two key states. It is used in order to obtain impression a smooth transition between them. Warping (deformation) is a distortion of the image depends on the area of the image, the mathematical equivalent of printing on flexible surface (tension and compression in various places).

$$
\begin{aligned}
&image_{\text{intermediate}} = (1-t)\cdot image_1 + t\cdot image_2 \\
&where: 0 < t < 1
\end{aligned}
\qquad (1)
$$

The first and simplest method of combining the two images was merging and blending of images. However, this method works only for images representing similar objects. For example human faces of the same size (size of the picture) and similarly positioned relative to camera. In such a case the calculation of the intermediate image is the linear interpolation of the color of all pixels.

If P means the color of point of the first figure and Q is the color of the corresponding point in the second image, then:

$$
\begin{aligned}
&V = Q - P; \\
&Qi = Q + i*(V/n);
\end{aligned}
\qquad (2)
$$

where: *n - number of steps (number of inserted frames),*
i - number of the current frame,

If the objects are presented in figures vary in size and location, it becomes necessary to matching image before blending. Global image matching is (for example) the translation and scaling of objects.

However, often we encounter the situation where the objects differ not only in size and position, but differ in shape. In such cases, the mixing of colors does not work. Also global image transformation is not useful. The solution is to adjust the details of the two objects. For example in case of human face, it is necessary to match an eye into an eye, an ear into an ear, and so on. In this way, the local distortions (non-parametric) are introduced to the image. Global transformations are functions of several parameters. Local transformation u (x, y) and v (x, y) can be defined independently for each position x and y (for each detail). Once we know the set of transformation vectors u, v, we can easily convert each pixel (reverse transformation plus interpolation).

An example of transformation of the image is local warping. This image transformation technique is used for static images. This is useful for retouching images. Another application of this transformation is the projection of two-dimensional images in three-dimensional space.

For the transformation it is necessary to define the control points of corresponding curves. Interpolation other positions allows for full conversion. Checkpoints define deformation curves. Curves provide a smooth transition set of vectors. Warping operation principle is presented in Figure 2.

Warping is a transformation of a single image. Just put a regular grid on the surface of entire image and then perform appropriate deformation. Grid nodes (key points) don't need to be directly linked to the object shown in the figure. Their density influences the effect of deformation - more dense grid will reduce the radius of the local deformations.

In the case of morphing the problem is more complicated. Both the first drawing, as well as the target figure will have its own mesh adapted to the objects shown in the drawings. Mesh nodes are the key points that are placed on specific points of the image.

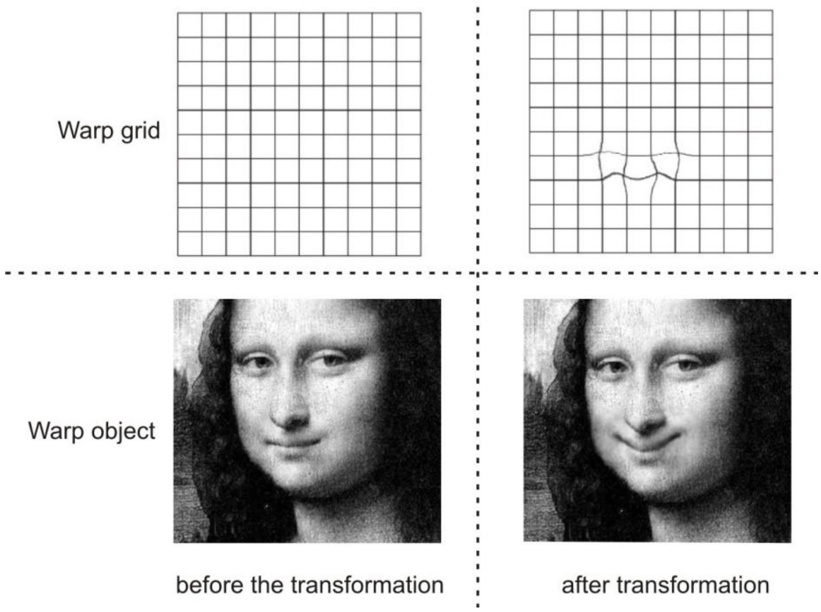

Figure 2. Warping operation principle

To make the transformation possible both grids must consist of the same number of triangles (the gridsmust have the same number of corresponding nodes). The first step in creating a grid is to determine the corresponding points in critical areas of the objects shown in the two figures. The next step is to define the triangle grid - grid stretched on these points (see Figure 3). In both of the images should be the same key points (grid nodes). These points, on both figures, can be placed in different locations. Grids define the corresponding triangles in the images. Each triangle is transformed separately from the starting to the final appearance.

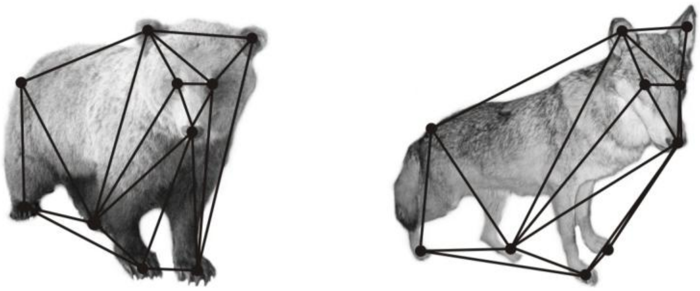

Figure 3. Matching grids and key points to objects

The vertices of the triangle are determined by the key points. Split the convex area, defined by a set of points, on triangles are called triangularization. There are many different results of triangularization. (see figure 4).

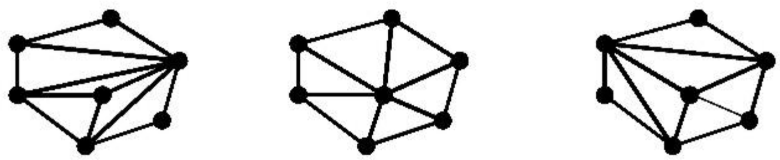

Figure 4. Different triangularization results of the same set of points

The results of the triangularization as shown in figure 4 varies with the quality of triangularization. Let a (T) = (a1, a2, .., a3t) be a vector angles triangularyzation T. Triangularyzation T1 would be better than triangularization T2 if (T1)> a (T2). Best triangulation maximizes the smallest angle (Delaunay triangulation).

The quality of the triangularization can be improved. In any convex quadrangle is possible to reverse the inner edge. If it improves the quality of the local triangularization also improves the quality of a global triangularization. If the reversal of the edge enhances the triangularization, the previous edge is called “illegal”.

There are many algorithms for scaling. The simplest of these is algorithm of triangularization of the complexity of O(n3)

Repeat for as long as possible

- Select 2 vertices.
- If the line connecting them does not intersect the previous line - save it (see figure 5).

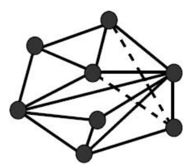

Figure 5. The simplest method of triangularization

As a result of the above described algorithm, we get a random triangularization. The easiest way of improving their quality is reversal successively all illegal edge. This operation should be repeated as long until these edges are present. An example of the optimization process is shown in Figure 6.

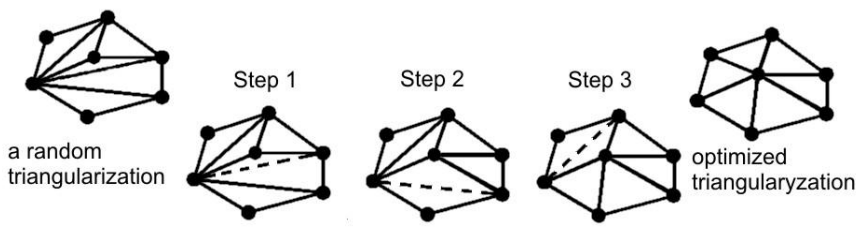

Figure 6. Improving the quality of triangularization by successive flip of illegal edges

The method described above is simplest trianguraziation, but inefficient and rarely used in practice. There are many better algorithms of triangularization. An example is the Delaunay algorithm which uses the division plane of on the Voronoi cells.

The problem is to set the key points (nodes of triangularization) for the morphing. The position of these points depend on the shape of the objects shown in the two figures. Key points must bind with each characteristic elements of both objects. From how to set the nodes depends on the quality of the effect (animation). Typically, this process is performed by the user program.

Automatic selection is key points might be hard. In this article I want to show one of the promising ways to solve this problem. The way this is the use of artificial neural networks.

2. Network structure

This article is presenting an example of using SOM network to find the key points of images. In this way, a set of points which can be used to create a grid.

SOMs, also called topological ordered maps, or Kohonen Self Organizing Feature Map (KSOMs)

- It maps all the points in a high-dimensional source space into a 2 to 3-D target space, s.t., the distance and proximity relationship (i.e., topology) are preserved as much as possible,
- Similar to specific clustering: cluster centers tend to lie in a low-dimensional manifold in the feature space,
- Clustering is performed by having several units competing for the current object,
- The unit whose weight vector is closest to the current object wins,
- The winner and its neighbors learn by having their weights adjusted,
- SOMs are believed to resemble processing that can occur in the brain,
- Useful for visualizing high-dimensional data in 2- or 3-D space,

These networks are used for pattern classification, such as continuous speech sounds, text and music. The most interesting uses of the stretching of the grid around the computer model of the scanned object.

SOM network construction can be present in two ways:

- topology of the network - that is a way of connecting neurons, the network topology influences the way of learning of individual neurons;
- space features - this is a mathematical space in which neurons move, position of neurons in the feature space brings a lot information, position in the feature space is determined by the weight of the neuron. Number of weight is associated with a number of dimensions of the feature space.

Network topology to be matched to a class of objects displayed in the figures. You can distinguish the following classes of objects:

- open curves;
- closed curves (this class also contains figures that are filled with the color highlighted contour);
- filled geometric shapes (both convex and non-convex);

In the first two cases, works a network of one-dimensional topology. Such a network may be open or closed. The open network (line segment having a beginning and an end) is useful for open curves. Closed network (network topology circle, the network having no beginning and no end) is better for closed curves. Two-dimensional SOM network topologies may be more useful in determining the key points in the pictures made of figures (shapes) filled with color (see figure 6).

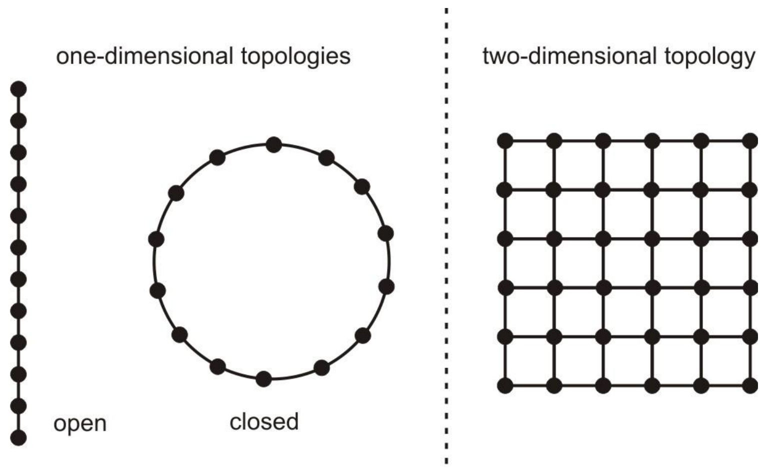

Figure 7. SOM network topologies

The program is using a one-dimensional network topology. Such a network is able to adapt to the curves and contours of searched objects. You can choose between an open and a closed network. The network is implemented using one-dimensional array of pointers to objects. Each neuron is represented by an object that holds its weight. Position the pointer to the neuron in the array determines its location in the topology. Neighboring neurons in the array are associated with each neighborhood.

In the program described here, a single neuron represents a point on the plane. Weights of neuron can be treated as coordinates (x, y), and they can be projected on the drawing. As a result of learning neurons will move from the initial (random) position and will stop at specific points in the drawings.

3. Network teaching

Teaching such a networks is to change the coordinates of neurons, so that they seek a pattern consistent with the structure of the analyzed data. Thus the network "strung out" around the data sets by matching them to its structure.

Output neurons (or more precisely their weights) can initiate random values.

Methods of changing the weights of output neurons is called teaching strategy. For SOM, there are two basic strategies:

- WTA (Winner Takes All). After the presentation of the network input vector, the neuron most similar to the present element (whose weights are most similar constituent input vector) is modified so that its weight was as close as possible to the input vector.

• WTM (Winner Takes Most). In this strategy, not only the most similar neuron but also its surroundings are modified. Frequently this modification is dependent on the distance from the neighbor's of winner.

In the described program a WTM strategy was used. WTA strategy is not effective because not all neurons are subject to the learning process. Neurons recognized at the beginning of a winning move in the direction of the object contour and adapt to it. However, there is a group of neurons that will not participate in learning process and will remain in the places where they appeared.

For WTM learning strategy, this problem does not occur because neurons are located close together in network topology are linked by neighborhood. Winning neurons entail neighboring neurons in the direction of the contour (Figure 8).

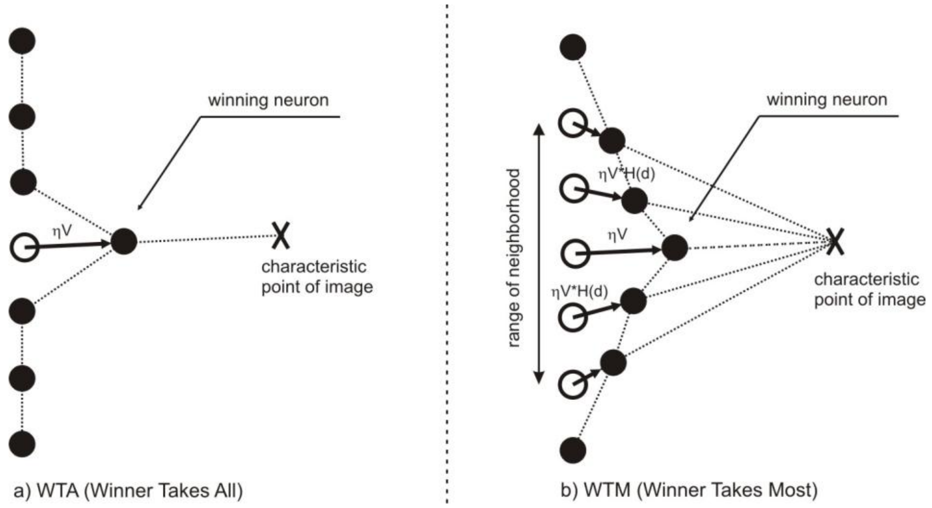

Figure 8. Strategies of learning one-dimensional SOM network

The learning process of the neuron (changes its position in the feature space) is described by the following formula (3).

$$N' = \eta(X - N) \cdot H(d)$$
$$\eta \to 0 \tag{3}$$

where: *N - the position of the neuron in the feature space*
N' - the position of the neuron in the feature space after the learning cycle
X - position specific point
η - factor of learning,
H - the neighborhood function,
d - distance between neurons in the network structure

Network learning algorithm used in the program:

1. Initial weights of neurons receive random values (small random values).
2. We randomly select one of the points of the image. If this point is black (the contour) perform the following operations:
   a. find the nearest neuron,
   b. we move neuron and all neurons within the neighborhood to a point (by a small amount - the distance from the point multiplied by learning step).
3. After studying all the points, we reduce the learning step and coverage the neighborhood.
4. We end learning, the learning step (factor of learning) drops to zero.

Implemented in the program network SOM learns from the first image. The result of the learning network is a set of key points adapted to the image. At the end of the learning process is given a new set of learning points (the final image projected animation).

The network learns and adapts to the new image.

The result of the program are two sets of points. These are the two versions (transformations) of the same mesh. The first version is adapted to the first image and the second image to the other. These meshes can be used to perform morphing of the images (see figure 9).

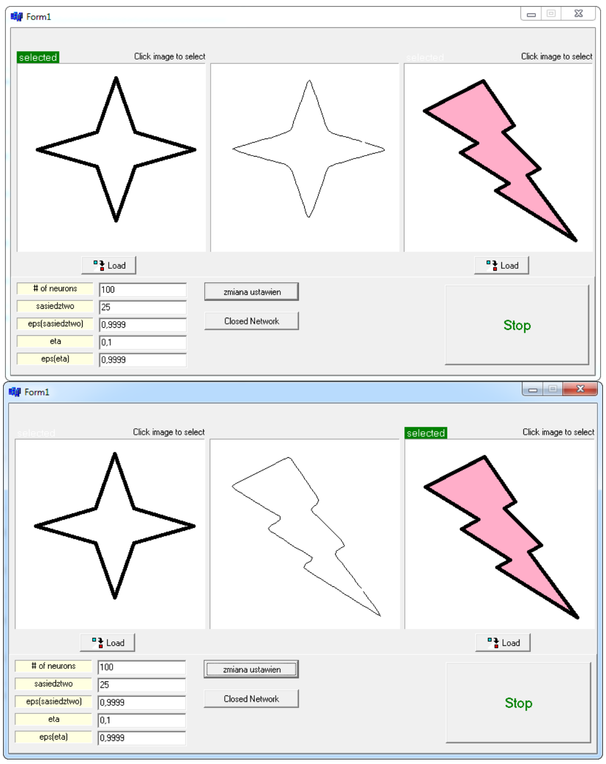

Figure 9. The effect of the program - a pair of meshes (initial and final)

It is worth noticing that during the operation the mesh changes in a smooth manner. The effect of the program can therefore be not only a pair of meshes (first and last), but a series of meshes (see Figure 10).

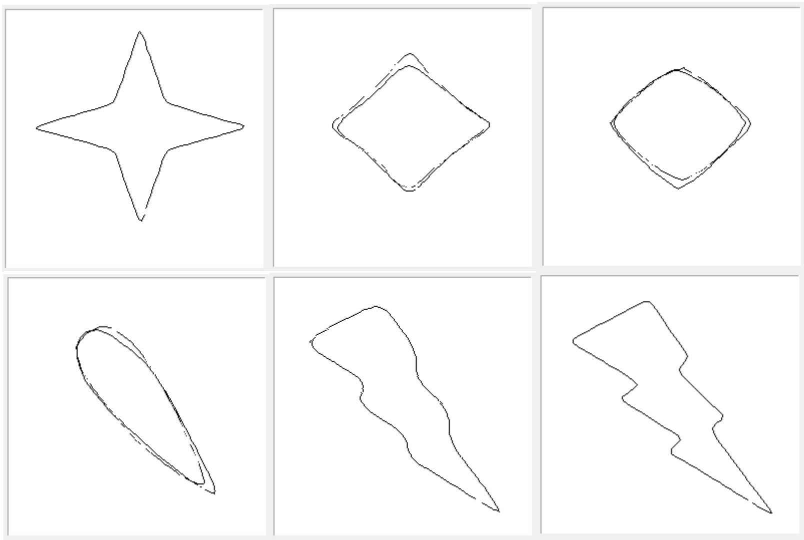

Figure 10. Successive phases of learning network (SOM network, which is learned from the first image smoothly adapts to the other)

4. Conclusions

The most common use of unsupervised learning neural network is clustering [7]. In the era of the power of information, each effective tool to explore the huge amount of data that a person is not able to analyze is very useful.

The fact that the SOM play a very important role in the fact that they are to this day widely used, developed and improved.

In this article it is shown one of the most interesting and non-standard SOM network applications.

Automatic generation of sets of key points tailored to the processed images can open many new interesting possibility of using morphing. For example, the morphing can be used as a visual effect, calculated in real-time in games and multimedia programs.

The problem that may hinder the use of such algorithms in the programs that run on personal computers will be necessary to use a neural network working in real time. These networks have a great demand for computing power.

You can specify two possible ways to solve this problem:

- optimization of neural network algorithms,

- optimal use of technology developed for personal computers.

The last working methods described here seems to be most promising. The development of personal computers, especially SIMD technologies used in modern GPU can allow for the implementation of a neural network with a high degree of complexity by using a commercially available and inexpensive equipment [2].

References

[1] Astel A., Tsakovski S., Barbieri P.L., Simeonov V., A comparison of SOM classification approach with cluster analysis and PCA for large environmental data sets, Wat. Res. 41 (19), 4566-4578 (2007)

[2] Bartoszewski A., Capabilities of hardware support in creation of artificial neural networks, [in] Computer science in the age of XXI century. Computer technologies in science, technology and education.A. Jastriebow [ed]. Technical Uniwersity of Radom, ITE. Radom 2012.

[3] Grabska-Chrząstowska J., Fiedor M., Tucholska K., Weryfikacja działania sieci Kohonena przetwarzających dane ankietowe zebrane wśród studentów krakowskiej AWF, Automayka, 10 (3), 503-511 (2006)

[13] Kohonen T., Self-organiznig formation of topologically correct feature maps, Biol. Cybern., 43, 59-69, (1982)

[5] Rauber A., Merkl D., Dittenbach M., The growing hierachical self-organiznim map: exploratory analysis of high-dimensional data, IEEE Transactions on Neural Networks, 13 (6), 1331-1342, (2002)

[6] Rutkowska D., Piliński M., Rutkowski L., Sieci neuronowe, algorytmy genetyczne i systemy rozmyte, Wydawnictwo Naukowe PWN, Łódź (1999)

[7] Tadeusiewicz R., Sieci neuronowe, Akademicka Oficyna Wydawnicza RM, Warszawa (1993).

[8] Foley J. D., van Dam A., Feiner S. K., Hughes J. F., Phillip R. L., Wprowadzenie do grafiki komputerowej, WNT Warszawa994.

WYKORZYSTANIE SAMOORGANIZUJĄCYCH SIĘ MAP W PRZEKSZTAŁCENIACH GRAFIKI

Streszczenie

W artykule przedstawione zostało niestandardowe wykorzystanie samoorganizujących się map (SOM). Zostały one użyte do znajdowania charakterystycznych punktów obrazu. Pozwala to na zautomatyzowanie procesu wyznaczania punktów kluczowych obrazów na potrzeby ich morfingu, czyli płynnej zamiany jednego obrazu w inny. Autor przedstawia algorytm oraz wyniki działania jego przykładowej implementacji.

Słowa kluczowe: sztuczne sieci neuronowe, SOM, morfing.