ISSN 2520-2820 (online)
ISSN 2524-0978 (print)

UDC 004.93'1
DOI: 10.33310/2524-0978-2019-1-7-70-74

Iryna USTENKO
ustenko.irina@gmail.com
ORCID: 0000-0003-1541-2414
Andrii USTENKO
austenko0@gmail.com
ORCID: 0000-0002-0546-7019
Serhii USTENKO
ustenko.s.a@gmail.com
ORCID: 0000-0003-4968-1233
Mykolayiv

LINEARIZATION OF THE CONTOURS OF THE REGIONS RECOGNIZED BY NEURAL NETWORKS ON IMAGES

In this paper, a method is proposed for linearizing the contours of image areas that can be obtained after recognition by neural networks. The need for the method is due to the fact that the contours are described by a large number of points, usually located in increments of 1-5 pixels, which is redundant in many tasks, for example, if you need to find rectangular or contours consisting of straight sections in the images. The developed method is implemented in the C++ programming language in the form of the lineSegment method.

***Keywords**: linearization, pattern recognition, neural networks, rectangular contours.*

Formulation of the problem

Recognition of images (objects, signals, situations, phenomena or processes) is the task of identifying an object or determining some of its properties by its image (optical recognition) or audio recordings (acoustic recognition) and other characteristics [1].

The pattern recognition systems, apart from the semantic difference that is embedded in the concept of an image, differ from each other also in the method of presentation (syntax):

– in classical models, the image is usually described by a set of features, each of which characterizes a certain property of an object;

– in structural models, a certain statement, generated by the grammar characterizing the class, acts as an image;

– in text processing tasks, the role of an image is performed by a certain chain of characters or a template representation of this chain (for example, regular expressions).

There are three methods of pattern recognition [2]:

1. Brute force method – a comparison with a database is performed, where for each type of objects various display modifications are presented. For example, for optical pattern recognition, you can apply the method of sorting the type of object from different angles, scales, displacements, deformations, etc.

2. Produced a deeper analysis of the characteristics of the image. In the case of optical recognition, this may be the determination of various geometric characteristics.

3. The use of artificial neural networks. This method requires either a large number of examples of the recognition problem in training, or a special neural network structure that takes into account the specifics of this task. However, it is distinguished by higher efficiency and productivity.

Analysis of recent research and publications

Pattern recognition tasks are essentially discrete counterparts of optimal solution search problems (discrete programming). These include a wide class of tasks in which for some, usually highly heterogeneous, in-

ISSN 2520-2820 (online)
ISSN 2524-0978 (print)

complete, fuzzy, distorted and indirect information, it is required to establish whether the objects, situations or phenomena under study have a fixed finite set of properties allowing them to be classified into a certain class.

Another of the important areas of application of the theory of pattern recognition is the problem of predicting the behavior of objects or the development of the situation. The tasks of this type include the tasks of technical and medical diagnostics, geological forecasting, forecasting the properties of chemical compounds, alloys and new materials, forecasting the harvest and the progress of building large objects, detecting forest fires, managing production processes, etc.

The task of pattern recognition also arises in artificial intelligence systems. For example, in the understanding of a natural language, symbolic processing of algebraic expressions, expert systems, etc.

This topic is one of the most relevant in scientific research and a lot of publications are devoted to it, for example, the work [3-6].

One of the most effective and common ways to represent and solve the problems listed above are artificial neural networks (ANN). They are based on an approach that is based on the concept of learning with examples. In this case, it is necessary to have a sufficient number of examples to configure the system being developed, which, after training, will be able to obtain the required results with a certain degree of reliability.

Research in the field of ANN experienced three periods of activation. The first peak in the 40s was due to the pioneering work of McCulloch and Pitts. The second was in the 60s of the last century thanks to the perceptron of Rosenblatt. The third – since the beginning of the 80s, neural networks once again attracted the interest of researchers, which is connected with the Hopfield energy approach (1982) and the error back-propagation algorithm for teaching a multi-layer perceptron (multi-layer direct distribution) first proposed by Verbos.

In recent decades, the world has been rapidly developing a new applied field of mathematics, specializing in artificial neural networks. Artificial neural networks – mathematical models, as well as their software or hardware implementations, built on the principle of organization and functioning of networks of nerve cells of a living organism. The relevance of research in this direction is confirmed by the mass of various applications of neural networks. It is an automation of pattern recognition processes, adaptive control, approximation of functionals, forecasting, creation of expert systems, organization of associative memory, and many other applications.

A wide range of tasks solved by neural networks currently does not allow creating universal, powerful networks, forcing them to develop specialized networks that operate using various algorithms. Nevertheless, the development trends of neural networks are growing every year. One of the first tasks solved with the help of neural networks was pattern recognition on graphic images. Since then, quite a lot of completely new solutions have been proposed, many well-known solutions and algorithms have been improved [7-8].

Research Objective

The aim of the work is to develop a method for linearizing the contours of image areas that were obtained after recognition by neural networks. The need for the method is due to the fact that the contours are described by a large number of points, usually located in increments of 1-5 pixels, which is redundant in many tasks, for example, if you need to find rectangular or contours consisting of straight sections in the images.

Basic material

When arbitrary areas are recognized by neural networks, as a rule, a polygon with a large number of points is obtained that is difficult to process. In Fig. 1 shows the original image, and Fig. 2 – the area recognized by the neural network is highlighted. To solve this problem, a linearization method is proposed.

ISSN 2520-2820 (online)
ISSN 2524-0978 (print)

Fig. 1. Source image

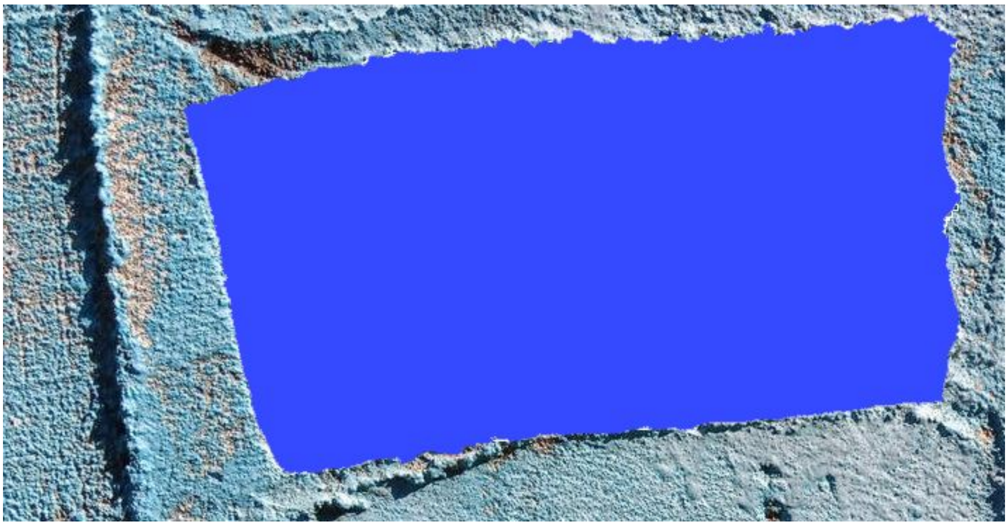

Fig. 2. The area allocated by the neural network

The initial data for the method are the coordinates of the points P_i of the closed contour (the data set begins and ends with one point) and the specified error ξ.

The general algorithm of the method is as follows:

1. The dependences of the coordinates x_i and y_i on the length of the contour s are constructed.

To calculate the length of the contour, the formula for calculating the distance between two points is used:

$$s_{i+1} = s_i + \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2},$$

where $i = 0,...,n-1$; n – amount of points; $s_0 = 0$.

2. Screening of points from the sets $x(s)$ and $y(s)$ is performed according to the algorithm given below.

3. Consider the remaining coordinates in the sets of points, are entered into the result set and:

– if $x(s_i) < y(s_j)$, then the next x coordinate is taken;

– if $x(s_i) > y(s_j)$, then the next coordinate y is taken;

– otherwise, the following x and y coordinates are taken.

The separation of points occurs recursively:

1. A straight line is drawn through the first and last points of each set:

ISSN 2520-2820 (online)
ISSN 2524-0978 (print)

$$x = k_x s + b_x;\ y = k_y s + b_y,$$

where $k_x = \dfrac{x_n - x_0}{s_n - s_0}$; $b_x = x_0 - s_0 k_x$;

$$k_y = \frac{y_n - y_0}{s_n - s_0};\ b_y = y_0 - s_0 k_y.$$

2. The point is located at the maximum distance from the obtained straight line (for each set separately). The distance is using elementary formulas:

$$d_x = \frac{|x_i - k_x s_i - b_x|}{\sqrt{1 + k_x^2}};\ d_y = \frac{|y_i - k_y s_i - b_y|}{\sqrt{1 + k_y^2}}.$$

3. If the distance is less than or equal to the specified error ξ, then it is added to the result set and the algorithm ends its work (terminal branch).

4. The algorithm is called for a set of points from the first to the found point (recursive branch).

5. The algorithm is called for a set of points from the found point to the last point (recursive branch).

The implementation of this algorithm is shown in Listing 1.

Listing 1

```
void lineSegment(double *s, double *x, int n, vector<int> & points,
                 double eps = EPS, int start = 0)
{
    double k = (x[0] - x[n - 1]) / (s[0] - s[n - 1]), b = x[0] - k * s[0], maxD = 0;
    int maxI = 0;
    for (int i = 0; i < n; i++)
    {
        double d = fabs(k * s[i] - x[i] + b) / sqrt(k*k + 1);
        if (d > maxD)
        {
            maxD = d;
            maxI = i;
        }
    }
    if (maxD <= eps)
    {
        points.push_back(start + n - 1);
        return;
    }
    lineSegment(s, x, maxI + 1, points, eps, start);
    lineSegment(s + maxI, x + maxI, n - maxI, points, eps, start + maxI);
}
```

In Fig. 3 demonstrated the work of the developed method. The number of source points is 80, the number in the output set is 37, the error is 1 pixel.

Conclusions

In this paper, a method was obtained and implemented in the C++ programming language, which allows to significantly reduce the number of points in the circuits obtained after recognition by neural networks, due to their linearization.

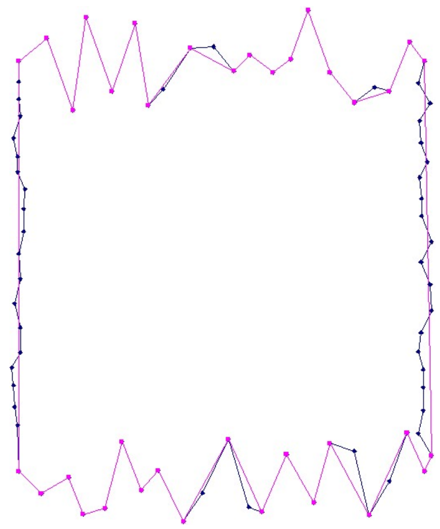

Fig. 3. Linearization results

REFERENCES

1. Колесников С. Распознавание образов. Общие сведения. *Компьютер-Информ*. URL: http://old.ci.ru/inform03_06/p_24.htm (дата звернення 16.03.2019)
2. Ustenko I., Ustenko S. Geometric recognition method of rectangular areas on images. *Геометричне моделювання та інформаційні технології*. № 2 (6). Миколаїв: МНУ імені В.О. Сухомлинського, 2018. С. 84-88.
3. Howard W.R. Pattern Recognition and Machine Learning. *Kybernetes*. Vol. 36. Issue 2. 2007. P. 275-275. DOI: 10.1108/03684920710743466.
4. Tveter D. The Pattern Recognition Basis of Artificial Intelligence. Wiley-IEEE Computer Society Pr, 1998. 388 p.
5. Bishop C.M. Pattern Recognition and Machine Learning. Springer Science+Business Media, LLC, 2006. 738 p.
6. Preston K., Carvalko J.R. On Determining Optimum Simple Golay Marking Transformations for Binary Image Processing. *IEEE Transactions on Computers*. Vol. C-21. Issue 12. 1972. P. 1430-1433. DOI: 10.1109/T-C.1972.223519.
7. Schmidhuber J. Deep learning in neural networks: An overview. *Neural Networks*. Vol. 61. 2015. P. 85-117. DOI: 10.1016/j.neunet.2014.09.003.
8. Scherer D., Müller A., Behnke S. Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition. *International Conference on Artificial Neural Networks*. 2010. P. 92-101. DOI: 10.1007/978-3-642-15825-4_10.

Ірина УСТЕНКО, Андрій УСТЕНКО, Сергій УСТЕНКО
Миколаїв

ЛІНЕАРИЗАЦІЯ КОНТУРІВ ОБЛАСТЕЙ, РОЗПІЗНАНИХ НЕЙРОННИМИ МЕРЕЖАМИ НА ЗОБРАЖЕННЯХ

У даній роботі запропоновано метод лінеаризації контурів областей зображень, які можуть бути отримані після розпізнавання нейронними мережами. Необхідність методу обумовлена тим, що контури описуються великою кількістю точок, як правило, розташованих з кроком 1-5 пікселів, що в багатьох задачах є надмірною, наприклад, якщо на зображеннях треба знайти прямокутні або такі, що складаються, в основному, з прямих ділянок, контури. Розроблений метод реалізований мовою програмування C++ у вигляді методу lineSegment.

Ключові слова: *лінеаризація, розпізнавання образів, нейронні мережі, прямокутні контури.*

Ирина УСТЕНКО, Андрей УСТЕНКО, Сергей УСТЕНКО
Николаев

ЛИНЕАРИЗАЦИЯ КОНТУРОВ ОБЛАСТЕЙ, РАСПОЗНАННЫХ НЕЙРОННЫМИ СЕТЯМИ НА ИЗОБРАЖЕНИЯХ

В данной работе предложен метод линеаризации контуров областей изображений, которые могут быть получены после распознавания нейронными сетями. Необходимость метода обусловлена тем, что контуры описываются большим количеством точек, как правило, расположенных с шагом 1-5 пикселей, что во многих задачах является избыточным, например, если на изображениях надо найти прямоугольные или состоящие, в основном, из прямых участков контуры. Разработанный метод реализован на языке программирования C++ в виде метода lineSegment.

Ключевые слова: *линеаризация, распознавание образов, нейронные сети, прямоугольные контуры.*

Стаття надійшла до редколегії 31.03.2019