FAST TRANSFORMATION OF IMAGES IN COMPRESSED FORM APPLIED TO DRAWINGS PROCESSING

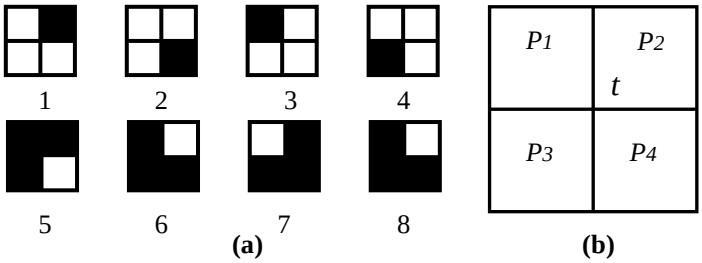
Vyacheslav Matsello, Vladimir Kiiko[1]

Abstract

The method for lossless compression of binary pictures is presented. Wide range of image processing operations can be effectively implemented directly to compressed images.

1. Corner representation (CR) – effective compression of binary images

We will call *corners* the vertices of polygons that constitute contours of binary picture. There are eight types of corners (Fig. 1,a). The set of coordinates of the corners represents one of possible lossless compression forms for binary pictures [3]. Term "lossless" means that this compression insures exact restoring of pixel representation of initial binary picture. Let t be the centre of a square consisting of 4 pixels (Fig. 1,b). $P1 \ldots P4$ be the values of these pixels (0 or 1). The point t is the corner if $|| P2 - P1 || - || P4 - P3 || = 1$.



(a) **(b)**

Fig. 1. Types of corners.

Number of corners can not exceed the number of pixels and usually is much less. Even for complicated line drawings corner representation provides compression rate of 10-20. But the main advantage is that many image processing operations can be implemented directly to pictures in compressed form. In this case the computational complexity of processing is proportional not to the number of pixels, but to the number of corners which is much less.

2. Image processing functions

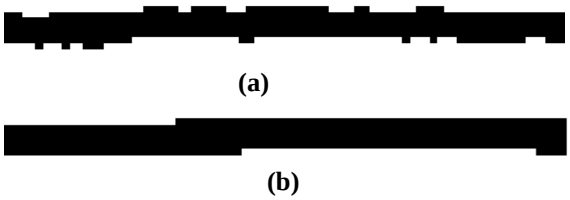
Some simple image processing functions can be easily performed using CR. For example, to translate, to scale or to rotate a picture by 90 degrees it is necessary to change coordinates of corners in certain way. To invert a picture it is enough to add each of four corners in vertices of the rectangle, bounding the picture, if there is no such corner, or delete it otherwise. Algorithms for

[1] IRTC ITS, 40, Pr. Akademika Glushkova, Kiev, Ukraine. Email: kiyko@image.kiev.ua; matsello@gmail.com

adding and subtracting of two binary pictures are given in [1]. Let us consider shortly another useful algorithms.

1.1. Contours smoothing

Discrete nature of digital pictures always results in distorted borders between black and white areas (Fig. 2,a). Such border information is useless, but it takes memory in CR. By deleting some corners we can make contours more smooth (Fig. 2,b). Simultaneously we reduce memory for storing CR and time for subsequent processing.



(a)

(b)

Fig.2. (a) – initial image, (b) – smoothed image.

1.2. Contours tracing

Effective algorithm for tracing contours during one pass scanning of CR is described in [2]. It can be used for detecting and measuring of image connected components. For example, "salt and pepper" noise can be removed by deleting of corners that constitute small connected objects on CR.

1.3. Erosion – dilation

Fast erosion-dilation algorithm for binary CR pictures is proposed in [1]. Its computational complexity does not depend on the number of pixels and the size of structuring element. It depends on the number of corners in CR of a picture and CR of structuring element.

1.4. Skeletonization

On the basis of CR the fast and effective skeletonization algorithm was developed [1]. It is width independent and requires testing of corners (not pixels). The algorithm is based on maximal squares detection and is structure preserving. It requires small working memory and its performance does not depend on the size of initial picture.

3. Conclusion

In the paper the CR method for lossless compression of binary images was described. Wide variety of fast and effective image processing algorithms were developed on the basis of CR. These algorithms were implemented in drawings processing software and have shown high performance and low memory requirements.

References

[1] V.M.Kiiko, M.I.Schlesinger, Fast dilation and skeletonization of compressed binary pictures, in: Henk J.A.M. Heijmans and Jos B.T.M. Roerdink (ed.), Mathematical Morphology and its Application to Image and Signal Processing, Kluwer Academic Publishers, pp. 347-354, 1998.

[2] V.V.Matsello, The algorithm for noise removing in line drawings, in Russian. Automatica, No. 2, pp. 86-89, 1986.

[3] M.I.Schlesinger, Mathematical tools for image processing, in Russian, Naukova Dumka, Kiev, Ukraine, 1989.