Digital Signal Processing 51 (2016) 73–81

Contents lists available at ScienceDirect

Digital Signal Processing

www.elsevier.com/locate/dsp

Single chains to represent groups of objects

Hiram H. López-Valdez [a], Hermilo Sánchez-Cruz [b,*], Magdalena C. Mascorro-Pantoja [c]

[a] *Departamento de Matemáticas, Centro de Investigación y de Estudios Avanzados del IPN (CINVESTAV-IPN), Apartado Postal 14-740, CP. 07000 Ciudad de México, México*

[b] *Departamento de Ciencias de la Computación, Centro de Ciencias Básicas, Universidad Autónoma de Aguascalientes (UAA), Av. Universidad 940, Col. Cd Universitaria, CP. 20131 Aguascalientes, Aguascalientes, México*

[c] *Departamento de Matemáticas y Física, Centro de Ciencias Básicas, Universidad Autónoma de Aguascalientes (UAA), Av. Universidad 940, Col. Cd Universitaria, CP. 20131 Aguascalientes, Aguascalientes, México*

ARTICLE INFO

Article history:
Available online 29 January 2016

Keywords:
Single chains
Chain codes
Contour shapes
Compression
Context-mixing

ABSTRACT

A chain code is a common, compact and size-efficient way to represent the contour shape of an object. When a group of objects is studied using chain codes, previous works require to obtain one chain code for each object. In this paper we assign a single chain to a group of objects, in such a way that all the properties of each object of the group can be recovered from the single chain. In order to achieve higher levels of compression, we propose a lossless method, that consists of representing a group of objects by means of a single chain, and then to apply a context-mixing algorithm. Regarding other methods of compression of the state-of-the-art, our experiments demonstrate that the best compression performance is achieved when our lossless method is applied. In this case more than 15% of a better compression level is reached.

1. Introduction

A *chain code* is a common and compact way to represent a contour shape.

The "Freeman chain code", proposed by Freeman in 1961 [1], is known as $F8$ *chain code* ($F8$ for short). It is composed of eight directions and travels through the center of the pixels of a contour shape on the basis of eight connectivity. Each movement direction is codified using a symbol $\alpha \in \{0, 1, 2, \dots, 7\}$ in counter clockwise direction (see Fig. 1).

On the other hand, $F4$ *chain code* ($F4$ for short) travels through the edges of the pixels of a contour shape using four connectivity. A movement direction is codified using a symbol $\alpha \in \{0, 1, 2, 3\}$ (see Fig. 2). It is also known as a *crack* code because it covers the contour shape along edges of border pixels [2–6].

In 1999, Bribiesca [7] proposed the *vertex chain code*, denoted by *VCC*. Among its most important features, *VCC* is composed by the symbols 0, 1 and 2. It has some information between the contour shape and the inner part of the object. This code represents the changes made of a contour shape by computing the number of affected pixels (see Fig. 3).

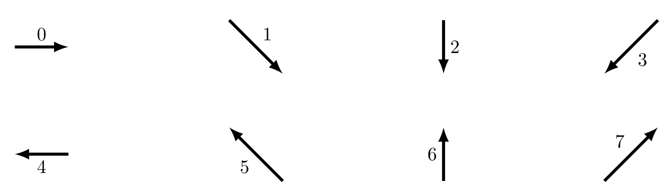

Fig. 1. Symbols to encode with $F8$.

Fig. 2. Symbols to encode with $F4$.

Sánchez-Cruz and Rodríguez-Dagnino [8] proposed, in 2005, the $3OT$ *chain code* ($3OT$ for short). They compared $3OT$ with $F4$ and obtained a better result thanks to the use of the symbols 0, 1 and 2 to label the changes generated in relation to orthogonal directions. $3OT$ has the same number of symbols than *VCC*, however it is composed of 3 vectors to codify the contour: *reference*, *support* and *change*. The symbol 0 represents no changes between reference and support, 1 represents a change equal to reference and 2 represents a change in contrary sense of the reference (see Fig. 4).

Relied on $F8$, also in 2005, Kui and Žalik [9] proposed a new chain code, that we called here $AF8$ *chain code* ($AF8$ for short). This code is based on changes obtained with every pair of $F8$ code vectors when following the contour, i.e. every vector of change in

* Corresponding author. Fax: +52 449 9108401.
E-mail addresses: hlopez@math.cinvestav.mx (H.H. López-Valdez), hsanchez@correo.uaa.mx (H. Sánchez-Cruz), mcmascorro@correo.uaa.mx (M.C. Mascorro-Pantoja).

http://dx.doi.org/10.1016/j.dsp.2016.01.007
1051-2004/

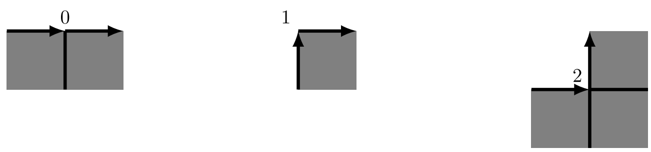

Fig. 3. Symbols to encode with *VCC*.

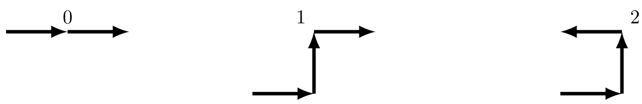

Fig. 4. Symbols to encode with 3*OT*.

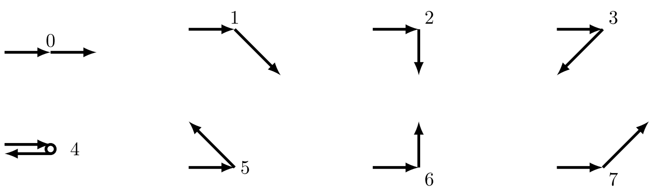

Fig. 5. Symbols to encode with *AF*8.

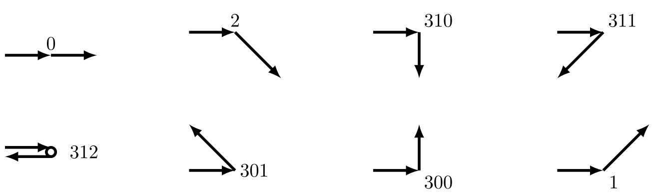

Fig. 6. Symbols to encode with *NAD*.

the contour is compared with the previous one, and depending on the angle, a symbol is assigned (see Fig. 5). *AF*8 was compared with *F*4, *F*8, *VCC* and 3*OT* [10], and the results showed that *AF*8 reported more advantages when using the Huffman algorithm. However, in 2009 the combination 3*OT*-Arithmetic coding [11] performed better compression levels than *AF*8-Arithmetic.

From $F4, VCC, 3OT, F8$ and *AF*8 other chain codes have been derived. The derived codes are obtained when combining the symbols that appear in the contours, making a probabilistic model to modify the number of bits required to store the coded contour shape. For example, E_VCC, V_VCC and C_VCC chain codes were proposed in [12]. The E_VCC chain code was obtained by considering that *VCC* uses two bits to represent three symbols. The V_VCC chain code arises by considering a variable-length of *VCC*. The C_VCC chain code is based on applying the Huffman algorithm on the *VCC* code-symbols. On the other hand, the M_3OT chain code was proposed when considering groups of symbols of 3*OT* [13], whereas *MDF*9 chain code was proposed by considering the *AF*8 patterns in pieces of discrete straight lines of the contour shapes [14]. Recently, *NAD* chain code was introduced in [15] and obtained more compression levels in [16]. It is a variation of *AF*8, where instead of using the symbols $0, \ldots, 7$, the authors use four symbols grouped as follows: $0, 2, 310, 311, 312, 301, 300$ and 1 (and labeling the angles according to Fig. 6).

The *AAF*8 *chain code* is the newest *basic code* (it is not derived from another known code), introduced in [17]. *AAF*8 chain code is composed of three vectors (two angles), in which, regarding the reference vector, a symbol of change direction is given, independently of the support vector direction. It is considered in a basis of eight connectivity. Table 1 shows the basic and the derived codes.

NAD is the latest chain code that surpasses the compression levels of the previous codes. We have compared our method with it. However, 3*OT* and *AF*8 are also worthy to be compared because they are very close to the *NAD* compression ratios, according to recent obtained results by Žalik et al. [16]. From these works, it can be observed that looking for better modifications of basic chain codes, together with the help of information theory, we can obtain a better compression performance. Of course, we must take into account that compression is also related to recognition, in the sense of being able to recognize redundancy in the information given by the geometry of the objects.

Table 1
Basic codes and their derived codes.

Proposed year	1961	1999	2005	2008	2010	2014
Basic codes	**F8, F4**	**VCC**	**3OT, AF8**			*AAF*8
Derived codes			*E_VCC*, *C_VCC* *V_VCC*	*M_3OT*	*MDF*9	**NAD**

In order to complete the comparisons, we also made our implementations using *F*8 and *F*4, because they are the classical chain codes used in literature.

All the chain codes above mentioned have the common particularity that they were designed to represent a contour shape of an isolated object. However, to achieve even higher compression levels, we propose to change the paradigm, instead of handling objects with isolated chains we now represent groups of objects using a *single chain*. The idea is simple: we concatenate the chain codes of each contour shape of the group. In general the concatenation of chains is always possible, and in order to recover the single chains that form the concatenated chain, it is not necessary to have more information, like the length of each single chain and the position of each object.

The purpose of the paper is to be able to transmit and maintain the shapes without loss of information from an original repository. Often the object repositories are given in different files, which can be numerated to sort the different objects. However, to save memory storage space, the set of objects can be placed in a single file, making sure the information on the form is unchanged, keeping it in a minimum bounding rectangular frame. As we demonstrate in this work, we avoid all extra storage thanks to the characteristics of the chain codes of objects. However, there are other interesting applications that do not need such an extra storage information, as we explain in Section 4.4.

Of course, if a real scene want to be recovered, the positions of objects should be taken into account, then an extra storage information has to be spent, like the distance between objects. This distance can be stored between chain as the number of pixels among starting pixel between each pair of objects.

1.1. Overview to contour coding methods

Since Freeman proposed the first chain code [1], a considerable amount of papers using chain codes for a wide variety of issues in different fields have appeared. On the one hand, they have been studied theoretically to propose unique descriptors and to reach high compression levels for binary objects [1,7–18]. When a group of objects is studied, authors of these previous published articles usually encode object by object.

On the other hand, chain codes have been used to real applications, like map representations and compression [2,19,20], to look for dominant points [21–24], to extract and to encode edges from 3D scenes [25–27], for analysis and shape recognition [28–30], for recognition of skeletal structures [31], and more recently for analysis and document compression [32–34].

For recognition tasks, obtaining a unique descriptor for each object that permits us object classification is a challenge. Perhaps this conviction has made the researchers do not propose until now concatenating the descriptors for groups of objects.

The novelty of our proposed method is to no longer consider isolated objects. As we have demonstrated our method facilitates