Available online at www.sciencedirect.com
ScienceDirect
J. Vis. Commun. Image R. 18 (2007) 35–44
JOURNAL OF VISUAL Communication & IMAGE Representation
www.elsevier.com/locate/jvci

Grayscale true two-dimensional dictionary-based image compression

Nathanael J. Brittain, Mahmoud R. El-Sakka *

Computer Science Department, The University of Western Ontario, London, Ont., Canada N6A 5B7

Received 13 September 2005; accepted 12 September 2006
Available online 25 October 2006

Abstract

Dictionary-based encoding methods are popular forms of data compression. These methods were initially implemented to reduce the one-dimensional correlation in data, since they are designed to compress text. Therefore, they do not take advantage of the fact that adjacent pixels in images are correlated in two dimensions. Previous attempts have been made to adapt dictionary-based compression schemes to consider the two-dimensional nature of images, but mostly for binary images. In this paper, a two-dimensional dictionary-based lossless image compression scheme for grayscale images is introduced. The proposed scheme reduces correlation in image data by finding two-dimensional blocks of pixels that are approximately matched throughout the data and replacing them with short codewords. Test results show that the compression performance of the proposed scheme outperforms and surpasses any other existing dictionary-based lossless compression scheme. The results also show that it slightly outperforms JPEG-2000s compression performance, when it operates in its lossless mode, and it is comparable to JPEG-LS's and CALIC's compression performance, where JPEG-2000 and JPEG-LS are the current image compression standards, and CALIC is a Context-based Adaptive Lossless Image Coding scheme.

Keywords: Image encoding; Lossless compression; Dictionary-based schemes; Two-dimensional compression; LZ schemes; Prediction

1. Introduction

In the field of image compression there are two major approaches, lossless and lossy. In lossless compression, when an image is compressed and then decompressed, the reconstructed image is an exact copy of the original. In lossy compression, some information about the image is discarded in order to achieve better compression. This means only a close replica of the original image can be retrieved from the compressed data. The compression scheme presented in this paper is a lossless scheme.

Among the most popular methods of lossless compression are dictionary-based schemes. Dictionary compressors encode a string of data by partitioning the string into many sub-strings, and then replacing each sub-string by a codeword. Communication between the compressor and decompressor is done using messages. Each message consists of a codeword and possibly other information. The dictionary in these schemes is the set of every possible codeword. LZ77 [1] and LZ78 [2] are two of the most famous dictionary-based compression schemes.

In LZ77, the dictionary is a portion of the most recently encoded data. This is also called the search buffer. Codewords for sub-strings are pointers to the longest match for the sub-string found in the search buffer. Each message consists of the codeword for the sub-string, the length of the match and the code of the next symbol.

There are many modifications to the original LZ77 scheme. Rodeh et al. introduced LZR [3], a scheme that uses LZ77 but with variable-size pointers. This means the pointer can index a sub-string anywhere in the previously encoded data, rather than just a previous portion. Storer and Syzmanski introduced LZSS [4], in which a flag bit is used to distinguish two types of messages, a pointer or a character. Bell introduced LZB [5], which also uses LZSS but with variable sized pointers as in LZR. LZH [6] is similar to LZSS, but it uses a Huffman encoder [7] to further

* Corresponding author. Fax: +1 519 661 3515.
E-mail address: elsakka@csd.uwo.ca (M.R. El-Sakka).

1047-3203/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jvcir.2006.09.001

compress the generated messages. In software, the PNG file format is based on LZ77.

The LZ77 approach assumes that similar patterns occur close to each other. In situations where a pattern repeats over a period larger than the search buffer size, the repetition cannot be taken advantage of. In LZ78 [2], no search buffer, is used. Instead, the dictionary in this scheme is an indexed list of some previously encountered sub-strings. In LZ78, each codeword consists of two parts, a pointer to the dictionary and the code of the next symbol.

As in LZ77, there are many modifications to the original LZ78 scheme. Welch introduced LZW [8], which is similar to LZ78, but its dictionary initially contains an entry for every possible symbol. Thus, LZW eliminated the need to include the code of the next symbol in messages. Miller and Wegman [9] introduced LZMW. LZMW is similar to LZW but is slightly modified when adding dictionary entries. Where LZW composes the last codeword sent with the next symbol to be encoded, LZMW composes the last codeword sent with the entire next codeword. Jakobsson introduced LZJ [10], which is similar to LZW but when the dictionary becomes full, codewords that have been used the least are replaced. Tischer introduced LZT [11]. In this scheme, the dictionary entries are arranged according to their recent use. When the dictionary becomes full, each new entry replaces the least recently used entry. In software, Unix Compress and the GIF file format are based on LZW.

Fiala and Greene introduced LZFG [12], which is similar to LZ77 because it uses a sliding window but also similar to LZ78 because only particular codewords are stored in the dictionary.

LZ77, LZ78, and their variants, take advantage of the fact that adjacent data values are highly correlated. These dictionary-based schemes are designed to compress text and so only reduce one-dimensional correlations in data. Therefore, they do not take advantage of the fact that adjacent data values (pixels) in images are highly correlated in two dimensions.

There have been few attempts to adapt LZ compressors to suit the two-dimensional nature of images. Perhaps the most straightforward attempt was to find a way to linearize the data and then use a one-dimensional compressor on the data [13]. However, tests showed that no one linearization is best for all images.

Storer and Helfgott [14] and Rizzo et al. [15] present a generalization of LZ77 to lossless compression of *binary* images. The algorithm, known as *two-dimensional sliding window block matching*, uses a wave heuristic to scan the image and a multi-shape *two-dimensional suffix trie* data structure to represent the dictionary, which is a window in previously encoded pixels. However, it is likely that this scheme will not perform well in the case of grayscale or color images, since the chances of finding large *exact* matches would be very small.

Dai and Zakhor [16] present a two-pass two-dimensional LZ77-based scheme for *binary* images. In this scheme, pixels are encoded by searching for an *exact* matching between these pixels and the already encoded pixels. Once such a match is found, these matched pixels are represented by the match location information. As in Storer and Helfgott and Rizzo et al. schemes, it is likely that, if this scheme is applied on grayscale or color images, it would not achieve very good results, due to the small chances of finding large *exact* matches.

Alzina et al. [17] introduced a *lossy* two-dimensional pattern matching compression scheme (2D-PMC) that is based on LZ77. The central part of this scheme is the search, in all previously encoded pixels, for approximate matches of rectangular blocks of certain predetermined sizes (typically 2×3, 3×2, 1×5, or 5×1). A block is encoded by a reference pointer to the previously encoded occurrence that produces the least error.

The dictionary-based scheme presented in this paper is designed to take advantage of the two-dimensional correlation between pixels in grayscale images. It is similar to Storer and Helfgott [14], Rizzo et al. [15], and Dai and Zakhor [16] two-dimensional dictionary encoding schemes, but it allows f*or approximate* matches since it is designed to compress grayscale images. It is also similar to Alzina et al. [17] 2D-PMC scheme, but it considers *any* rectangular block sizes. Moreover, the proposed scheme encodes residuals as well to produce lossless results.

The rest of this paper is organized as follows. Section 2 describes the proposed scheme in detail. Section 3 presents the results. Finally, Sections 4 and 5 offer the suggested future work and the conclusions of this paper, respectively.

2. The proposed GS-2D-LZ scheme

In this paper, a novel grayscale two-dimensional Lempel–Ziv image compression scheme (denoted GS-2D-LZ) is proposed. This scheme is designed to take advantage of the two-dimensional correlations in the image data. It relies on three different compression strategies, namely: two-dimensional block matching, prediction, and statistical encoding. The basic idea of the two-dimensional block matching encoding is to represent a block of uncompressed pixels by a pointer to the best approximate occurrence of that block in the compressed part of the image. This should reduce the interpixel redundancy of the block. In the case of not finding good enough approximate occurrences, prediction is used to reduce the interpixel redundancy of the block. Since GS-2D-LZ is a lossless scheme, residuals generated from approximate matches and predictions are encoded as well. To further compress the image, a statistical encoder is used to reduce the encoding redundancy as much as possible.

2.1. An overview of the GS-2D-LZ scheme

In GS-2D-LZ, an image is encoded in raster scan order processing one block of pixels at each step. For each block of pixels, an approximate match is searched for in previ-