CMiC Image Compression Standard

Andrew Goodney - Claremont-McKenna College Spring 2016

April 19, 2016

1 Introduction

This document describes the CMiC image compression standard. The standard has been developed as the basis for an undergraduate level course in data compression. Basing a project around this specification will give students experience implementing several lossless and lossy data compression algorithms. The goal of the standard is not large compression ratios, but rather that students can pull together several compression techniques that might be learned across a semester long course. The standard is defined only by the valid bitstream and a reference decoder. Students will also gain a basic understanding of how images are represented in a computer.

1.1 Compression Techniques Used

The CMiC image format uses the following data compression techniques, each of which is described in more detail in the following sections:

- Sub-band decomposition using the 2D discreet wavelet transform
- Differential encoding of pixel data
- Quantization of the DWT coefficients
- Huffman encoding of the differential and quantized components.

1.2 CMiC Processing Overview

Figure 1 shows a block diagram of the processing steps used in the CMiC image compression standard. CMiC v1 only supports greyscale images, however future versions of this standard will include support for multi-channel images, as well as sub-sampled chroma channels. The data processing steps that follow apply to a single channel. First image data is broken into four sub-bands using the 2D discreet wavelet transform. Choice of wavelet is up to the implementer as the chosen wavelet is included in the header. The LL band

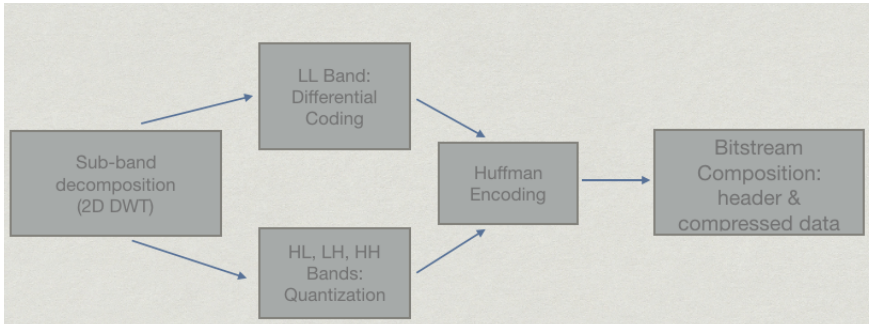

Figure 1: CMiC Block Diagram

data is differentially encoded. The HL, LH and HH bands are quantized with an integer 'q' that is also chosen by the designer and carried in the image header. The differentially encoded and the quantized DWT coefficients are then Huffman encoded. To simplify the image format, the Huffman coding dictionary is also included in the header.

1.3 Sub-band Coding with the 2D DWT

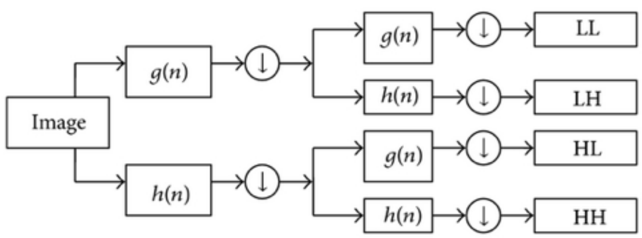

Figure 2: 2D DWT Block Diagram

Sub-band coding is a popular technique for breaking down a signal into two or more bands based on the frequencies present. It is technique that underlies most audio compression schemes. When applied to images, one level of decomposition yields four sub-bands that correspond to the low frequency components ('LL', essentially a 1/2 resolution version of the image, though the dynamic range might expand), the horizontal detail components ('LH') , the vertical detail components ('HL') and the diagonal detail components ('HH').

The 2D discreet wavelet transform is a popular sub-band decomposition technique for images (though there are other filters that can be used), and conceptually it is usually implemented by a series of cascading high and low pass filters, as shown in figure 2. In figure

Figure 3: 2D DWT Decomposition Example

3 we see the 2D DWT decomposition of an example image. We can see that most of the information in the image has been retained in the LL region, while the other regions contain relatively little information. This observation is the reason why the 2D DWT is a popular decomposition technique for image compression. With the information decomposed into subbands, we can use knowledge of the human visual system to chose appropriate lossy data compression techniques for each subband, while maintaining good image quality for decompressed images.

1.4 Differential Encoding of LL Band Coefficients

As can be seen in the previous section, the LL band carries much of the image information therefore we would like to preserve as much as this data as possible. Helpfully, most images have local areas that change slowly from pixel-to-pixel. Therefore the LL data is a good target for differential encoding, which preserves the underlying data but reduces the dynamic range of the data values. This in turn increases the compression that is possible using Huffman Coding. In order to make differential encoding easier to implement we do accept some loss of information by quantizing (or rounding) the LL pixel values to the nearest integer. The 2D array of pixels should then be flattened into a 1D array and the differential encoding step performed.

The differential encoding can be shown with the following example. Suppose the array `[1,2,3,4,5]` is to be differentially encoded. After encoding this array can be represented by `[1,1,1,1]` representing the pair-wise differences between adjacent elements. If the starting value, `1` in this case is put at the front of the encoded array, then all information necessary to reconstruct the original array is available to the decoder.

1.5 Quantization of the LH, HL, HH Band Coefficients

Research into image compression and transmission has yielded a good understanding of how the human visual system perceives images. One result, that is also used in the JPEG compression standard, is that the human visual system is less sensitive to absolute accuracy in higher frequency bands. Thus much like JPEG we can coarsely quantize the coefficients in the LH, HL and HH bands with out much degradation in reconstructed image quality. In CMiC the LH, HL and HH coefficients are first rounded to the nearest integer and divided by the "q" factor. This reduces the dynamic range of the coefficients, and for many images results in a large number of zero values coefficients. The quantized coefficients will then compress better with a Huffman code.

1.6 Huffman Coding of the Coefficients

Huffman coding is a well known lossless compression technique that given data with skewed statistics (*i.e.* some symbols occur much more often than others) produces an optimal or near optimal encoding to binary for that data. In CMiC we take the differentially encoded LL coefficients, and the quantized LH, HL, and HH coefficients together as the source data and generate a binary Huffman encoding of that data. The coefficients are first flattened and concatenated into a 1D array in the following order: LL, LH, HL, HH.

2 CMiC Data File Format

The CMiC data file format defines the bitstream representing a CMiC compressed image. It is divided into two sections:

- A header in human readable JSON format
- A binary data section for the Huffman encoded coefficients

2.1 JSON Header

CMiC uses the JSON data interchange format to produce a human readable header. The first line of the header is a JSON dictionary that describes the file with the following keys:

- `version`: 'CMiCv1' for the version of the CMiC image format described in this document
- `height`: image height in pixels
- `width`: image width in pixels
- `wavelet`: wavelet name used in the 2D DWT

- `q`: quantization factor used on LH, HL, and HH coefficients

The second line of the header is a JSON dictionary that gives the Huffman encoding table for the mapping from integers to binary code needed to reconstruct the coefficient arrays. The keys of the dictionary are the integers found in the coefficient array and the values are a binary string representing the binary code used for that integer.

For example: `{‘0’: ‘0’, ‘-1’: ‘001’, ‘1’: 011, ‘2’: ‘111’}`

2.2 Binary Huffman Coded Coefficients

Following the two header lines the binary Huffman coded coefficients are written the the file. The amount of data will be variable, however it will always decode to an array of `width x height` number of integers that can be used to reconstruct the LL, LH, HL and HH coefficient arrays needed to reconstruct the image.