CHAPTER 3

WATERMARK COMPRESSION IN DIGITAL IMAGE WATERMARKING

3.1 INTRODUCTION

Digital imaging techniques produce different size images, suitable for particular applications. The images out of proper size make problems for some secondary applications. For example large size medical images require more storage space, need more transmission time and bandwidth in teleradiology. Besides these obstacles, during image communication an image contents may be changed due to noisy communication channels or hackers manipulation. Medical image data is very sensitive and cannot afford changes to its original dataset. Digital watermarking techniques have been used to detect and recover illegal changes made to images. The watermarking of a medical image with heavy payload causes image perceptual degradation and affects patient diagnosis. To reduce watermark payload and maintain the image perceptual, diagnostic qualities standard, the watermark needs to be compressed. The compression should be in such a way to ensure the compressed data originality. For sensitive data like medical image watermarks, lossless compression is used. Watermark lossless compression ensures no loss of data while reducing the size of dataset. Watermarking of medical images with reduced number of bits preserves the image perceptual and diagnostic qualities. This chapter focuses on watermark lossless compression using different techniques including LZW. LZW compression technique is also applicable to binary watermarks. The binary version watermarks contain of large number of repeating sequences of binaries and most suitable for lossless compression using LZW technique. LZW is a better approach to compress data having repeating sequences of data elements like binaries (Nelson et al., 1989).

3.2 METHODOLOGY

This chapter methodology is watermark lossless compression using LZW technique. For this purpose, ten different samples of ultrasound medical images are chosen for watermark generation and compression experiments. Each sample is divided into ROI and RONI; ROI is the central and rich in information part, while RONI is the remaining image. Watermark is generated from combination of image selected pixels as ROI and secret key. The LZW selection for watermark compression is made after its comparison with other conventional compression techniques such as PNG, GIF, PBM, JPEG and JPEG 2000 of lossless versions. This selection is made on the basis of more bits reduction and good compression ratio.

3.2.1 Watermark Generation and Compression

A 100 * 100 size pixels segment of each image sample is selected as ROI. ROI size is allowed to vary but here we keep it fixed for results comparison of different compression techniques. A watermarking secret key is generated and applied to watermark to get a secured watermark. Watermark security is necessary to hide the originality of this important data from attackers. Here watermark is the combination of ROI and ROI hash code, while secret key is the binary of ROI hash repeated equal to the watermark size. A XOR operation is performed between watermark and secret key to obtain the secured watermark. LZW is a dictionary based compression technique applied to data for elimination of repeating sequences (Nelson et al., 1989). PNG, GIF, PBM and JPEG are conventional compression methods have been tested on the same size watermark compression to compare the results with LZW. For LZW compression, first the watermark is converted to binary and stored in a single row binary array. This conversion gives binaries unique sequences for LZW effective compression. During compression, every unique sequence is replaced by a decimal number called string code. More number of bits reductions makes LZW as the best choice of binary watermark lossless compression in medical image watermarking.

Before starting LZW compression of a binary watermark, an array as a dictionary is initialized with two stings, ‘0’ and ‘1’. These are only two unique values in watermark binary stream otherwise the dictionary is initialized to the distinct characters in an alphabetic string. The combinations of binaries of binary watermark give different repeating sequences. During watermark compression process unique strings are formed based on binary sequences and inserted into dictionary. Every unique string is allotted a decimal code and inserted to another array called codes table. LZW Algorithm 3.1 checks the availability of newly constructed string in the dictionary. In case of its uniqueness, the string and its allotted code are inserted into dictionary codes table as shown in Figure 3.2. This process of unique strings formation, codes allocation and insertion continue till the whole watermark is compressed. At the completion, codes table values give the LZW compressed watermark. The following Algorithm 3.1 explains step by step process of watermark lossless compression. The final dictionary and codes table are used for watermark lossless decompression after the watermark extraction at destination (Alarabeyyat et al., 2012).

Algorithm 3.1: LZW algorithm for binary watermark compression

```
1.    Convert watermark to binary
2.    Dictionary={‘0’,’1’}
3.    String = get the first binary from binary watermark
4.    WHILE get the next available binary and continue
5.          Next-binary = Get the next binary
6.    IF String + Next-binary exist in dictionary then
7.          String = String + Next-binary
8.    ELSE
9.          Assign decimal code to String and insert into code table
10.         Add String + Next-binary to the dictionary
11.         String=Next-binary
12.   END of IF
13.   END of WHILE
14.   Output dictionary and code table
```