

BIL464 Multimedia Systems

2014-2015 Fall

Laboratory Experiment 5

Instructor : Mustafa Sert

December 18, 2014

Assistant(s) : S. Ezgi Küçükbay

Objectives

- LZW compression & decompression

1. LZW Compression (Group-1 is supposed to answer this question)

The basic algorithm for LZW compression is given below. Implement this pseudo code in Matlab.

In the pseudocode that follows, *pixelString* is a sequence of pixel values. *pixel = next pixel value* means "read the next pixel out of the image file." *pixelString + pixel* means "take the current *pixelString* value and concatenate *pixel* onto the end of it."

ALGORITHM

LZW COMPRESSION ALGORITHM

```

algorithm LZW
/*Input: A bitmap image.
Output: A table of the individual colors in the image
and a compressed version of the file.
Note that + is concatenation.*/
{
  initialize table to contain the individual colors in bitmap
  pixelString = first pixel value
  while there are still pixels to process {
    pixel = next pixel value
    stringSoFar = pixelString + pixel
    if stringSoFar is in the table then
      pixelString = stringSoFar
    else {
      output the code for pixelString
      add stringSoFar to the table
      pixelString = pixel
    }
  }
  output the code for pixelString
}

```

DEMO:

Read a gray-scale image and compress it with the given algorithm. Output the code and the final code table.

2. LZW Decompression (Group-2 is supposed to answer this question)

The basic algorithm for LZW decompression is given below. Implement this pseudo code in Matlab.

ALGORITHM

LZW DECOMPRESSION ALGORITHM

```

algorithm LZW_decompress
/*Input: Compressed bitmap image and table of individual colors in image.
Output: Decompressed image.*/
{
  stringSoFar = NULL
  while there are still codes to process in the code string {
    code = next code in the code string
    colors = the colors corresponding to code in the table
    if colors == NULL /*Case where code is not in the table*/
      /*stringSoFar[0] is the first color in stringSoFar*/
      colors = stringSoFar + stringSoFar[0]
    output colors
    if stringSoFar != NULL
      put stringSoFar + colors[0] in the table
    stringSoFar = colors
  }
}

```

DEMO:

Read a LZW encoded code and display the original content and the final code table.