LSB Matching Revisited Algorithm

David Torpey

May 20, 2017

Contents

1 Introduction

The algorithm proposed in 'LSB Matching Revisited' works on gray-scale cover images. The message, which must be given in a binary representation, is embedded using two cover image pixels at a time. Essentially, a message m will be embedded into a cover image I, which will result in a steganographic image S. After the embedding of the message into the cover image, the LSB of the stegonographic image's ith pixel is equal to the the ith bit of the message: m_i. Futhermore, the value of the $(i+1)$th message bit m_{i+1} is a function of y_i and y_{i+1}.

2 Algorithm

The algorithm allows for the selection of addition or subtraction of y_i to carry information. This is because the selection of either addition or subtraction can set a particular binary function to a desired value. However, such a binary function has to have the following two properties:

$$f(l-1,n) \neq f(l+1,n), \forall l,n \in \mathbb{Z} \tag{1}$$

$$f(l,n) \neq f(l,n+1), \forall l,n \in \mathbb{Z} \tag{2}$$

The function proposed in the paper that satisfies these two properties is:

$$f(y_i, y_{i+1}) = \texttt{LSB}(\lfloor \frac{y_i}{2} \rfloor + y_{i+1}) \tag{3}$$

The algorithm to perform the embedding is depicted below. The input into the algorithm is a pair of cover image pixels x_i and x_{i+1}, and two message bits m_i and m_{i+1}. The output is then a pair of steganographic pixels y_i and y_{i+1}.

input: a pair of cover image pixels x_i, x_{i+1}
    two message bits m_i, m_{i+1}
output: a pair of stego image pixels y_i, y_{i+1}

if $m_i = \text{LSB}(x_i)$
  if $m_{i+1} \neq f(x_i, x_{i+1})$
    $y_{i+1} = x_{i+1} \pm 1$
  else
    $y_{i+1} = x_{i+1}$
  end
  $y_i = x_i$
else
  if $m_{i+1} = f(x_i - 1, x_{i+1})$
    $y_i = x_i - 1$
  else
    $y_i = x_i + 1$
  end
  $y_{i+1} = x_{i+1}$
end