A Steganographic Method Using Learning Vector Quantization

Larisa Gorbashko [1)], Vladimir Golovko [2)]

Brest State Technical University, Moskovskaja str. 267, 224017 Brest, Belarus
1) lagorbashko@bstu.by
2) gva@bstu.by

Abstract: *The new technique for embedding image data is presented. The message is subjects to vector quantizer by neural network. The modified data is inserted into the coiner in the wavelet transform domain. The vector quantization enables to increase the capacity of embedded data. The experimental results indicate that performance of vector quantizer by neural network is higher then quantizer by standard algorithm.*

Keywords: steganography, neural network, vector quantizer, wavelet transform.

I. INTRODUCTION

Steganography is one of the fundamental ways by which data can be kept confidential. Steganography has evolved into the practice of hiding a message within a larger one. Legitimate purposes of steganography can include watermarking images for copyright protection or confidential information to protect the data from possible unauthorized viewing.

The main properties of image steganography are transparency, capacity, security and robustness. Transparency refers to the visual similarity between the original image and stegoimage. If large amounts of data are embedded in an image, then the visual quality of the image degrades which indicates the presence of steganography. Capacity refers to the volume of data that can be embedded in an image.

Many digital steganography algorithms have been proposed in recent literature [1,2]. However, most of the existing algorithms haven't high embedding capacity. The increasing of an embedded data volume is an important problem of modern digital steganography.

The embedding of hidden data takes place into the time domain or frequency region of signal carrier. In last case stegoimage has a more high robust and stability to attacks. All recent steganographic researches realize the embedding of data in frequency domain [2,3].

Here we have combined modern steganographic methods with advantages of neural network such as accuracy and performance. It is proposed a hiding algorithm of image within multimedia data, for example an image too. The key moment of this method is using learning vector quantization (LVQ) to enlarge a capacity of hidden data. Neural network is used in order to create a uniform codebook and to raise a vector quantization power. This method allows hide multimedia content.

In next section will be described a production of codebook by neural network. Section 3 details the steps in data embedding and the recovering procedure. Section 4 and 5 consider experimental results and conclusions.

II. THE DATA HIDING TECHNIQUE

The scheme of data embedding is shown at fig.1. The

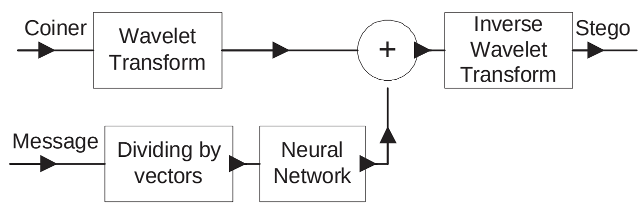

Fig.1. Scheme of data embedding

key moment of this scheme is embedding using vector quantizer by neural network. The image- coiner is transformed using the Wavelet Transform (WT). One level of WT is applied for increasing the volume of embedded data, the more levels of WT is recommended in order to raise a robust to JPEG2000 – compress.

The preparing of message consists of three steps; two of this is realized by neural network (NN). Foremost the image- message is replaced by its vector representation with length eight (block Dividing by Vectors) as shown fig.1

The block NN decide of task to create of codebook in process of training and vector quantizer of message. It is realized by Kohonen network. Kohonen's Self Organizing Feature Maps have used. They provide a way of representing multidimensional data in much lower dimensional spaces. This process reduce the dimensionally of vectors and essentially compress a data known as vector quantization.

After NN's training is produced a vector quantizer of message and received numbers are embedded to wavelet coefficients by additive algorithm. The modified coefficients subject to inverse WT and we have the stegoimage as a result.

III. CODEBOOK BY NEURAL NETWORK

It is proposed a neural network to coding image by a vector quantizer. As a learning vector quantization we used Kohonen network [4]. It consists of two neural units' layers (fig.2) and permits to divide the input space in a number of disjoint subspaces.

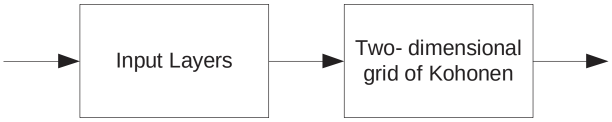

Fig.2. The structure of neural network

The output layer is two- dimensional grid of array and should give the cluster label of the input pattern. The input data, which are near to each other in the input space, must be mapped on output units, which are also new to each other. The competitive learning is used for Kohonen network's training. According to this approach it is defined a winner neuron with k- number at the beginning. The Euclidean distance measure is used for winning neuron's determination:

$$Dn = \min \left| X - W_j \right|$$
$$D_n = \min_j \sqrt{(x_1 - W_{ij})^2 + (x_2 - W_{ij})^2 + \ldots (x_n - W_{ij})^2} \quad (1)$$

where $X=\{x_1, x_2,\ldots, x_n\}$ is an input data, $W_j= \{W_{1j}, W_{2j},\ldots, W_{nj}\}$ are weight coefficients of j- output neuron, n is a dimension of an input image.

Next, the weights of the winning unit are adapted using the following learning rule

$$W_k(t+1) = W_k(t) + \gamma * h(k, j) * (X(t) - W_k(t)). \quad (2)$$

Here h(k,j) is a decreasing Gaussian function of the grid- distance between units k and j:

$$h(k, j) = \exp(-\frac{(j-k)^2}{2\sigma^2}. \quad (3)$$

The parameter σ is decreased during learning of neural network. The number of input units is eight; the size of output array is 16x16.

The graphical representing of codebook is shown in fig.3. Prior to network training, each unit's weight must be initialized. It is formed random sample of 8-dimensional vectors to learn network. Each number of sample are initialized so what $0<w<1$. After learning the weights of units have pointed a center of a cluster. These weight coefficients keep to so-called codebook. The columns of this dimension are a numbers or clusters, the rows of it are coordinates of clusters center.

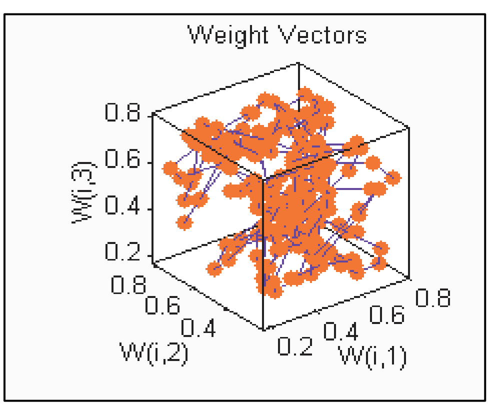

Fig.3. Visualization of codebook

IV. EMBEDDING

The embedding algorithm should satisfy requirements of JPEG2000 Standard in order to ensure a robust of stegoimage to JPEG2000- compress[5]. JPEG2000 uses a discrete wavelet transform (DWT) to decompose image into its high and low subbands. In standard multiple stages of the DWT are performed. The number of stages performed is implementation dependent and consists from 0 to 32 levels. For natural images, usually between 4 to 8 stages are used. One of compress methods is to cast off insignificant coefficients.

The embedding is made in frequency region. The coiner is decomposed using wavelet transform (WT). The wavelet coefficients (WTC) are divided into four parts: an approximate image LL and three detail images LH, HL and HH. Every next level of decomposition reduces the LL-area in four times. The values of detail coefficients are about zero and they don't save information after inverse transform. Therefore the hide information is embedded to LL-coefficients.

The preparing of hidden image consists of replacement of image vector with dimension 8 by the number of cluster.

Then these vectors enter on neural network. Each vector is replaced for value of weight coefficient of its cluster.

We have the number sequence with values from 1 to 256 as a result. The transparency of stegoimage might be violated at addition this numbers to WTC. This problem has decided thus. Each number V_i was divided into two parts: high and low:

$$\begin{aligned} High(Vi) &= [Vi/16], \\ Low(Vi) &= Vi - [Vi/16]*16. \end{aligned} \quad (4)$$

In such a way each vector representative by two numbers, each taken separately has the value less then 16.

The additive algorithm takes place at embedding:

$$f'(m,n) = f(m,n) + \beta * w_i, \quad (5)$$

where f(m,n), f′(m,n)- WTC before and after embedding accordingly, w_i- embedding number, β-coefficient for increase of energy signal and PSNR after recovering. We decrease distortions of hidden image due to this coefficient.

The modified coefficients are inverse wavelet transformed to produce a stegoimage.

The hidden image is recovered following essentially an inverse sequence of operations. Some moments of this action demand of explanation only. After calculation the difference between the stego and coiner we have received the hidden image as sequence of vector number. We should have a codebook to reconstruct of image. We replace each vector number by 8-dimentional sequence according to codebook. In that way we take initial image.

V. EXPERIMENTAL RESULTS

This method of data embedding was implemented with using a packet MathLab [6].

The image 256x256 was chose as a coiner. The tests were realized at the quantity of decompose levels from 1 to 5. The more number of decompose levels the less capacity of embedded data, but the more the robust to JPEG2000-compress. The ration between the embedded data and a coiner size is 1:4 if the number of decomposed levels is one; 1:64 if the number of decompose levels is four.

The coefficient β is from 2 to 8, this value is quite enough for the ensuring of available level of PSNR.

Fig.4 points basic and transformed images. The recovered image has a visible distortion, but the image of a sportsman is quite identifiable. The vector quantization introduces the most of these distortions. After vector quantization of transmitted image the PSNR averages 36-40 dB. The PSNR of recovered signal is reduced on 2-5 dB yet. This significance is acceptable [7].

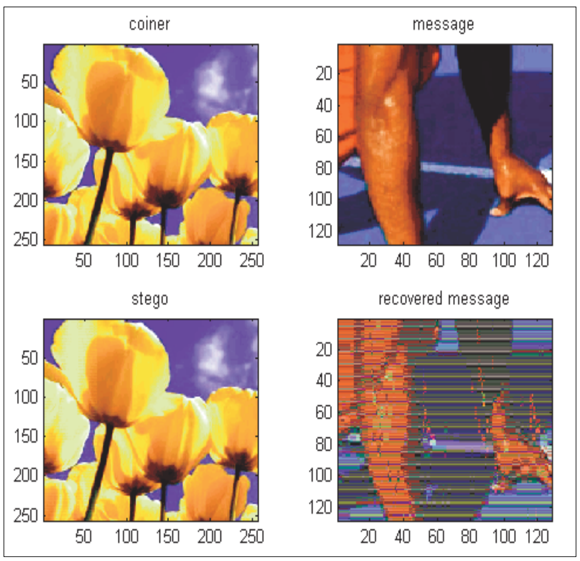

Fig.4. Visualization of experimental results

VI. CONCLUSIONS

In this paper a method of data embedding was be presented. This method allows sending one image within other image. The stegoimage has a transparency and robust to JPEG2000- compress. We used a learning vector quantizer built on neural network. It consists of two neural units' layers: input layers with number of neurons eight and output layer. It is a map of Kohonen with size 16x16 neurons.

The acquired codebook is fixed and may be used for quantization of any image. The codebook volume is 2 kB.

The recovering a hidden image produces with a coiner. For that an image-coiner, a codebook and a stegoimage are transmitted separately – on different channels or in due time. The size of hidden image conflicts with robust to compress and it is aggregates from 1 to 25 percent depending on decomposition levels.

This method can be used to transmit of hidden information and to embed a digital watermarking for copyright protection. In last case the author should has a registered logotype.

The neural network application has some of advantages. *First* the amount of codebook vectors is fixed and it's equal to 256. One codebook can be used for quantization of all images due to this property. It isn't necessary to build a codebook for each image. The building of fixed codebook by standard mathematics means has a large calculated complexity. *Second* the period of quantization process with using neural network less then one without neural network. This difference is about one number exponent.

REFERENCES

[1] C.C. Chang and W.C. Wu. "A Steganografic Method for Hiding Secret Data Using Side Match Vector Quantization". http:\\ ietisy.oxfordjournals.ogr\cgi\ content\refs\E88-D\9\2159. Copyright 2005 The Institute of Electronics, Information and Communication Engineers.

[2] H. Heijmans and L. Kamstra "Reversible Data Embedding Based on the Haar Wavelet Decomposition". *Proceeding of VIIth Digital Image Computing: Techniques and Applications*, 2003, Sydney.

[3] L. Gorbashko. "A Text Data Hiding and Recovering without Host Image" *Proceeding of 8th International Conference of Pattern Recognition and Information Proceeding (PRIP'05)*, 18-20 May 2005 Minsk, Belarus, pp.205-207.

[4] T. Kohonen. "Self-organized Formation of Topologically Correct Feature Maps" *Biological Cybernetics.*, 1982, No.43, pp.59-69.

[5] Michael D. Adams. "The JPEG-2000 Still Image Compression Standard" http:\\ www.ece.uvic.ca \~madams.

[6] В.С. Медведев и В.Г. Потемкин "Нейронные сети. MatLab 6" Москва, 2002.

[7] В.А. Грибунин "Цифровая стеганография" Москва, 2002.