

Watermarking in the spatial domain

1. Introduction

Watermarking techniques aim inserting imperceptible information in a host signal or image for allowing implementation of copyright protection, copy protection or fingerprinting-like operations. The purpose of the laboratory is to implement watermarking techniques operating on the spatial domain. The techniques that will be considered are:

- the LSB method
- the spread spectrum-based watermarking method

LSB –based watermarking is one the simplest techniques which, for a given length watermark, works in substitution mode and replaces the LSB or more than one less significant bits with the bits of the watermark (the concept is described in Fig.1).

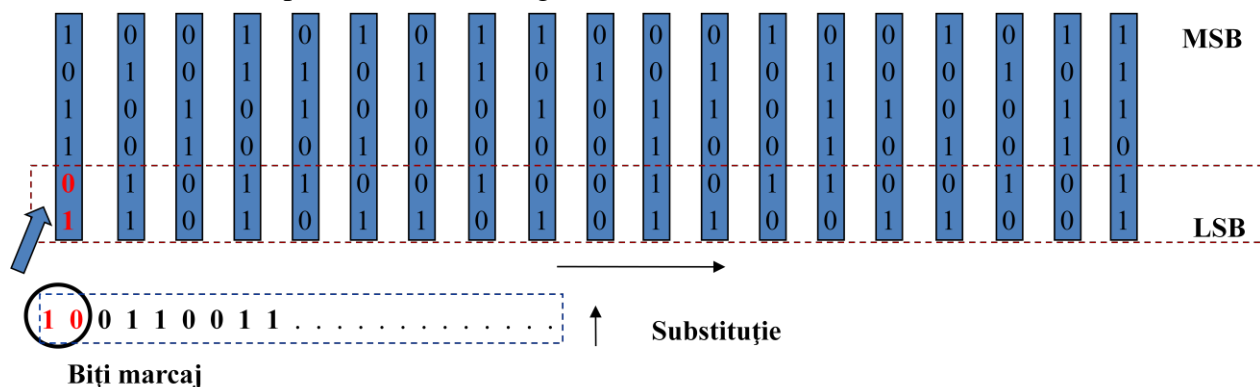


Fig.1 Watermarking method working on the least significant 2 bits

The most important advantages of the method are its simplicity and its robustness with respect to cropping-like attacks. The main disadvantage is that the method is not robust to substitution attacks (it is extremely simple to replace the watermarked bits with other watermarks).

Spread spectrum techniques are addressing this issue and are using a secret key that governs the watermarking process using an additive law. The concept is illustrated in Fig.2 [1] and it can be described mathematically as follows:

$$U(x, y) = I(x, y) + \text{alfa} * m(i) * PN(x, y) \quad (1)$$

In eq. (1) $m(i)$ represents a bit of the message that is to be inserted, represented in bipolar convention, alfa is a weighting factor that controls the amplitude of the watermarking signal and $PN(x, y)$ a pseudo-noise sequence that has the same length as the number of pixels in the input image.

The PN sequence is uniquely identified by the encryption key K and by an algorithm that generates it using K as a seed. For inserting multiple bits, the PN sequence is rotated with one position at the time. Since a rotated PN sequence is also a PN sequence and based on the excellent autocorrelation properties of PN sequences, this insures that multiple watermarks will not be influencing each other.

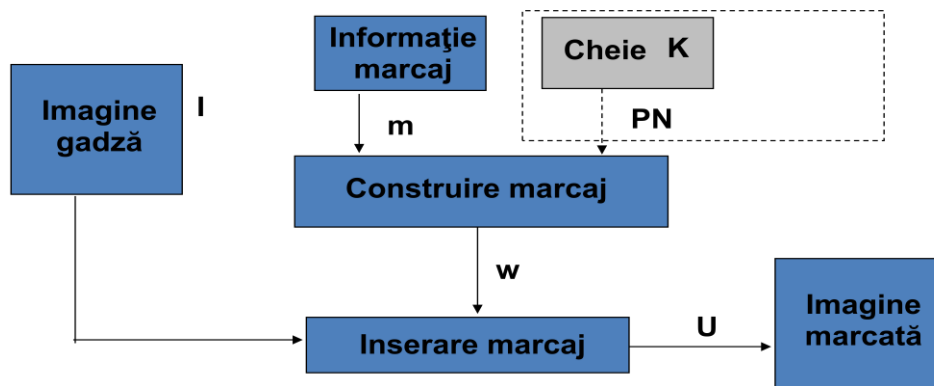


Fig.2 Spread spectrum watermarking

2. LSB-based watermarking in VisualDSP ++

The Visual DSP project **Marcare_LSB**, placed on the discipline's website, includes all the necessary function for embedding a watermark using the LSB technique. The declaration section is as follows:

```

unsigned char imIn[16384];
unsigned char imOut[16384];
static unsigned char mesaj[11] = "Laborator 5";
static unsigned char mesaj_binar[88];
  
```

and it defines the following variables : *imIn* variable to hold the input image, *imOut* –variable to store and allow visualization of the processed image, *mesaj* –the actual message (m in Fig.2) to be embedded , *mesaj_binar* – the binary expression of the previous message
The last variable - *masca_binara_pixeli*- is used to define the positions where the watermark will be placed (Example: 0xFC = 111111100 in binary)

An auxiliary function is included in the project:

```
void conversie_binar( char * caracter, char * destinatie)
```

for allowing the textual message to be converted onto a binary format. Watermark insertion is done as follows:

```

k=-1;
for (i=0; i<latime; i++)
for (j=0; j<inaltime; j++){
  adresa=(++k)%87;
  tmp=imIn[i+j*latime] & masca_binara_pixeli + mesaj_binar[adresa]*2 +
  mesaj_binar[adresa+1];
  imOut[i+j*latime]=tmp;
}
}
  
```

3. Exercises

- 1) Surf through the C code, identify the role of each statement and insert explanatory comments into the code.

- 2) Using the Visual DSP's debugging features establish which is the binary message that will be embedded onto the host image
- 3) Choose and input image, run the application and analyze the effects on the output image
- 4) Modify the application in order to insert the watermark on the most significant two bits of the input image. In the perceptual transparency property verified?
- 5) Modify the application in order to insert the watermark on the least significant 4 bits of the input image. Which is the result on the processed image?

4. Spread-spectrum watermarking in VisualDSP ++

5. The Visual DSP project **Marcare_spectru_imprastiat**, placed on the discipline's website, includes all the necessary function for embedding a watermark using spread-spectrum techniques. Three supplementary variables are defined:

```
#define alfa 24
int secventaPN [16384]={
#include "PN.dat"
};
int secventaPN_deplasata[16384];
```

The PN sequence was generated using Matlab as it is included in the PN.dat file that automatically initializes the *secventaPN* variable.

The constant *alfa* has a predefined value of 24 and, in order to insert more than one bit, *the secventaPN_deplasata* was defined. The actual rotation of the PN sequence is carried out using the C function: *void Deplasare_ciclica(int * src, int * dest, int nr_el)*

Watermark insertion is done using the following C loop:

```
for (k=0;k<88;k++)
{
Deplasare_ciclica( secventaPN , secventaPN_deplasata, 16384);
for (i=0; i<latime;i++)
for (j=0; j<inaltime;j++)
{
if (mesaj_binar[k]==0)
tmp=-1;
else
tmp=1;
val= imIn[i+j*latime]+tmp*secventaPN_deplasata[i+j*latime]*alfa;
if (val <=0)
val=0;
else
if (val>255)
val=255;
else
;
imOut[i+j*latime]=val;
}
}
}
```

5. Exercises

- 1) Surf through the C code, identify the role of each statement and insert explanatory comments into the code.

- 2) Run the application and analyze the effects on the processed image.
- 3) What happens if *alfa* is set to have a higher value ?
- 4) Modify *alfa* to 2.0. How this change impacts on the robustness of the watermark?
- 5) State the modifications that should be brought to the application for allowing insertion of the watermark using a perceptual mask.

Bibliografie

[1] Tehnologii de prelucrare a informației – lecture notes 2023-2024, <http://ares.utcluj.ro/tpi>