A WATERMARKING SYSTEM FOR TEACHING STUDENTS TO RESPECT INTELLECTUAL PROPERTY RIGHTS

Maria Chroni, Angelos Fylakis and Stavros D. Nikolopoulos
Department of Computer Science, University of Ioannina, GR-45110 Ioannina, Greece

Keywords: Watermarking Techniques, Intellectual Property Rights, Teaching IP, Educational Tools, Visual Representations, Learning Process, Color Images, 2D-representations of Permutations, Algorithms.

Abstract: Our work proposes a watermarking system for supporting the teaching process for educating students to respect intellectual property rights. In particular, we propose an educational tool, which we named WaterIP, that can be efficiently used by students to enable them to perceive how to protect their own ideas and how to respect others' intellectual property. Our system uses an efficient technique for watermarking images by exploiting certain properties of a specific 2D representation of permutations, it has a friendly graphical user interface and shows interesting performance figures. The system provides students with two main working levels: (I) the student creates a secret key, i.e., the watermark, and embeds it into his own image, and (II) he makes the marks of a watermarked image visible in order to later extract the watermark from it using only, for pedagogical reasons, a ruler and a pencil. The watermarking method behind WaterIP can be applied to all educational levels beginning from early childhood up to preliminary and high schools. We demonstrate the educational effectiveness of our WaterIP system by presenting ways of how it can be applied in class and show that WaterIP helps to understand what intellectual property rights really stand for.

1 INTRODUCTION

Digital watermarking (or, simply, watermarking) is a technique for protecting the intellectual property of a digital object; the idea is simple: a unique identifier, which is called *watermark*, is embedded into a digital object which may be used to verify its authenticity or the identity of its owners

The watermarking problem can be described as the problem of embedding a watermark w into an object I and, thus, producing a new object I_w, such that w can be reliably located and extracted from I_w even after I_w has been subjected to transformations (Collberg and Nagra, 2010); for example, compression in case the object is an image.

2 THEORETICAL TOOLS

In this section we present representations of the two main objects of our watermarking system: the permutations and the digital color images. In particular, we propose a 2D representation of permutations and give a 3D representation of color images.

2.1 Permutations

Informally, a permutation of a set of objects S is an arrangement of those objects into a particular order, while in a formal (mathematical) way a permutation of a set of objects S is defined as a bijection from S to itself (i.e., a map $S \to S$ for which every element of S occurs exactly once as image value).

Permutations may be represented in many ways. The most straightforward is simply a rearrangement of the elements of the set S, where S is the set $N_6 = \{1,2,\dots,6\}$; in this way we think of the permutation $\pi = (2,5,3,1,6,4)$ as a rearrangement of the elements of the set N_6 such that "1 goes to 2", "2 goes to 5", "3 goes to 3", "4 goes to 1", and so on. (Sedgewick and Flajolet, 1996; Golumbic, 1980).

2D Representation of Permutations. In this representation, the elements of the permutation $\pi = (\pi_1, \pi_2, \dots, \pi_n)$ are mapped in specific cells of an $n \times n$ matrix A as follows:

- integer $i \quad \longrightarrow \quad$ entry $A(\pi_i^{-1}, i)$

or, equivalently,

Chroni M., Fylakis A. and Nikolopoulos S..
A WATERMARKING SYSTEM FOR TEACHING STUDENTS TO RESPECT INTELLECTUAL PROPERTY RIGHTS.
DOI: 10.5220/0003920103360339
In *Proceedings of the 4th International Conference on Computer Supported Education* (CSEDU-2012), pages 336-339
ISBN: 978-989-8565-06-8

- the cell at row i and column π_i is labeled by the number π_i, for each $i = 1, 2, \ldots, n$.

In our 2DM representation, a permutation π over the set $N_n = \{1, 2, \ldots, n\}$ is represented by an $n \times n$ matrix A as follows:

- the cell at row i and column π_i is marked by a specific symbol, for each $i = 1, 2, \ldots, n$.

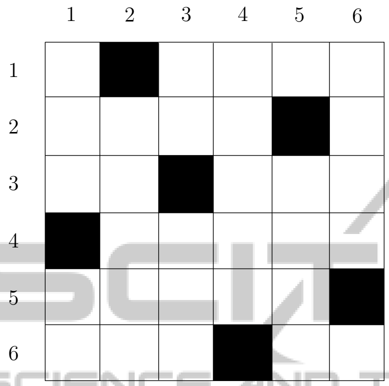

Figure 1: A 2DM representation of $\pi = (2, 5, 3, 1, 6, 4)$.

We next present an algorithm which extracts the permutation π from its 2DM representation matrix. More precisely, let π be a permutation over N_n and let A be the 2DM representation matrix of π ; given the matrix A, we can easily extract π from A in linear time (in the size of matrix A) by the following algorithm:

```
Algorithm Extract_π_from_2DM
Input: the 2DM representation matrix $A$ of $\pi$;
Output: the permutation $\pi$;
1. For each column $i$ of matrix $A$, $1 \le i \le n$, do:
       find the marked cell $j$ in the column $i$,
       and set $i$ in cell $A(j,i)$ and 0 in all other cells;
2. For each row $i$ of matrix $A$, $1 \le i \le n$, do:
       find the cell $j$ in the row $i$ with value not
       equal to 0, and set $\pi_i \leftarrow A(i,j)$;
3. Return the permutation $\pi$;
```

2.2 Color Images

A digital image is a numeric representation of a 2-dimensional image; it has a finite set of values, called *picture elements* or *pixels*, that represent the brightness of a given color at any specific point in the image. (Gonzalez and Woods, 2007).

The range of colors can be represented on the Cartesian 3-dimensional system. The axes x, y and z are used for the red green and blue color respectively.

In our system, since a color is a triple of integers (x,y,z), a digital image I of resolution $N \times M$ (i.e., it contains N rows and M columns of pixels) is stored in a three-dimensional matrix Img of size $N \times M \times 3$ as follows:

if the pixel $I(i,j)$ of the image I has (x,y,z) color, then $Img(i,j,1) = x$, $Img(i,j,2) = y$, and $Img(i,j,3) = z$.

3 OUR WATERMARKING SYSTEM

In this section we describe the modules and the main functions of the proposed watermarking system. Our system, which we named WaterIP, provides to a student two main working levels:

- **Embed Level.** Through a friendly graphical user interface, the student creates a secret key (i.e., the watermark w) and selects a picture I in which he wants to embed the watermark; in our system the watermark w is a permutation π over the set N_6 and it is embedded into the original picture I, using the 2DM representation, resulting the watermarked picture I_w.
- **Mark Level.** The student, in order to prove that he is the owner of the picture I_w, inputs the watermarked picture I_w into the system which makes the marks visible to the student so that he will be able to easily extract the watermark w (i.e., his secret key) just by looking at the marks; in particular, the system returns the marked picture I_m to the student.

We should mention that our system uses a permutation π over the set N_6 for the watermark w. The set N_6 was selected in purpose; we preferred to use a fixed size rather than giving the size as a choice to the user. That choice was made in order to make the system simple as it is designed to serve educational needs. Also 6 is not a great number so it is relatively easy to memorize by a student, nor it is a very small number and that makes the permutation more presentable.

3.1 Embed Watermark into Image

We next describe an algorithm which embeds a permutation π into an image I;

The algorithm takes as input a permutation π and an image I, in which the user wants to embed the watermark $w = \pi$, and produces the watermarked image I_w; it works as follows:

Step 1. The algorithm first computes the 2DM representation of the permutation $\pi = (\pi_1, \pi_2, \ldots, \pi_6)$, that is, it computes the 6×6 array A (see, Subsection 2.1); the entry (i, π_i) of the array A contains the symbol "*", $1 \le i \le 6$.

Step 2. Next, the algorithm takes the input image I and places on it an imaginary 6×6 grid which covers the whole images. Then it scans the image and goes to each grid-cell $C_{ij}(I)$ (there are always 36 grid-cells in any image) and locates the central pixel p_{ij}^0 of the grid-cell $C_{ij}(I)$, $1 \le i, j \le 6$.

Then, it computes the difference between the brightness of the central pixel p_{ij}^0 and the average brightness of the eight pixels $p_{ij}^1, p_{ij}^2, \ldots, p_{ij}^8$ around it, and stores this value in the variable $\text{dif}(p_{ij}^0)$

Finally, it computes the maximum absolute value of all 36 differences $\text{dif}(p_{ij}^0)$, $1 \le i, j \le 6$, and stores it in the variable $\text{Maxdif}(I)$.

Step 3. The algorithm goes again to each central pixel p_{ij}^0 of each grid-cell C_{ij} and if the corresponding entry $A(i, j)$ contains the symbol "*", then it increases its brightness k_{ij}^0 by the value e_{ij}^0 so that it surpasses the image's maximum difference $\text{Maxdif}(I)$ by a constant c; that is, $k_{ij}^0 + e_{ij}^0 = \text{Maxdif}(I) + c$.

In our system we use $c = 9$, and thus the brightness k_{ij}^0 of the central pixel of each grid-cell C_{ij} is increased by e_{ij}^0, where

$$e_{ij}^0 = \text{Maxdif}(I) - k_{ij}^0 + 9 \tag{1}$$

$1 \le i, j \le 6$.

Let I_w be the resulting image after increasing the brightness of the 6 central pixels, with respect to π, of the image I. Hereafter, we call these 6 central pixels as *2DM-pixels*; recall that, p_{ij}^0 is a 2DM-pixel if $A(i, \pi_i)$ = "*", or, equivalently, the cell (i, π_i) of the matrix A is marked.

3.2 Show Marks on Image

Next we describe our system's algorithm which is responsible for making the marks of a watermarked image I_w visible to the student so that he will be able to extract, in a specific way, the watermark w by hand.

The algorithm takes as input a watermarked image I_w in which the student wants to make the markings visible, and produces the marked image I_m; the steps of the algorithm are the following:

Step 1. The algorithm places again the same imaginary 6×6 grid on image I_w and locates the central pixel p_{ij}^0 (i.e., the 2DM-pixel) of each grid-cell $C_{ij}(I)$, $1 \le i, j \le 6$; there are 36 central pixels in total. Then, it finds the 6 central pixels $p_1^0, p_2^0, \ldots, p_6^0$, among the 36, with the maximum brightness using a known sorting algorithm.

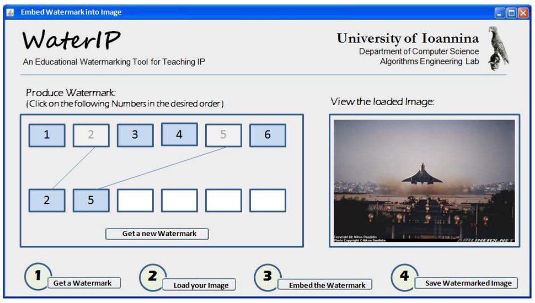
Embed Watermark into Image
WaterIP
An Educational Watermarking Tool for Teaching IP
University of Ioannina
Department of Computer Science
Algorithms Engineering Lab
Produce Watermark:
(Click on the following Numbers in the desired order)
1
2
3
4
5
6
2
5
Get a new Watermark
View the loaded Image:
1 Get a Watermark
2 Load your Image
3 Embed the Watermark
4 Save Watermarked Image

Figure 2: Part of the WaterIP interface.

Step 2. In this step, the algorithm takes the 6 grid-cell $C_1, C_2, \ldots, C_6$ of the image I_w which correspond to 6 central pixels $p_1^0, p_2^0, \ldots, p_6^0$, and places a red film over the whole area of each grid-cell C_i, $1 \le i \le 6$.

To this end, the algorithm takes each pixel $p_{i,j}$ of each grid-cell C_i, $1 \le i \le 6$, and modifies the brightness $k_{i,j} = (x_{i,j}, y_{i,j}, z_{i,j})$ of the pixel $p_{i,j}$ as follows:

$$k'_{i,j} = (255, y_{i,j}, z_{i,j}) \tag{2}$$

where, j takes $N/6 \times M/6$ different values. The resulting image is the marked image I_m

Step 3. Finally, the algorithm returns the marked image I_m.

4 INSIDE THE CLASSROOM

Our work proposes a supporting educational tool for teaching, inside the classroom, the value of intellectual property. As mentioned before we believe that the best way to gain people's respect towards intellectual property rights is to start from the roots, and that is the early education. Students should be taught, as part of the ethical education, to respect intellectual property at schools. (Watkins et al., 2007).

We consider that our tool can be applied in all educational levels, as part of various lesson plans, beginning from early childhood; uploading an image or photo the teacher can demonstrate the system to preschool learners and set them ethical dilemmas concerning the proper use of images or photos. Teachers of preliminary and high schools can go on more complex meanings and explain learners the concepts of watermarking, embedding, extracting, permutation, image analysis, etc.

We next use an imaginary scenario where two classmates claim the ownership of the same image, Alex who is really the owner of the image and Bob who claims that he is the owner, and show how our WaterIP system helps both to understand what intellectual property rights really stand for.Then we recommend a lesson plan of how our WaterIP system can be efficiently applied in groups inside a classroom during a course.

4.1 Two Classmates

Let us first present the scenario of the two classmates, Alex and Bob, where both claim the ownership of an image.

The main idea of the scenario is, that Alex first paints a picture at a computer or just takes a photo and uploads it to the computer. Afterwards he runs our system and places the numbers between 1 and 6 in a random order that he memorizes and keeps secret. That order actually forms the permutation which also is the watermark that Alex wants to embed in his digital image. He selects from a menu which image he wants to use and then runs the algorithm and places the watermark into it. Then he gets the watermarked image. That watermark will later be his proof that the picture was really made by him and that he deserves to be rewarded for it and not someone else; of course we shall not forget that in order to do that he should have memorized the permutation in order to claim the property of the image. Afterwards Alex uploads the watermarked image in his personal student web-page or in the lesson's webpage, making it public.

In Figure 2 we show a form, part of the system's friendly user interface, in which Alex produces the watermark that he wants to embed in his image. What he does here is that he selects the numbers from the upper set by clicking on them and places them linearly in the second set of numbers which forms the watermark (Button 1) and then selects and views an image (Button 2); the figure depicts an intermediate stage where Alex has already load an image and has selected the fist two numbers, i.e., 2 and 5, of his watermark. After the creation of his watermark, he is able to embed it in the image (Button 3) and save the watermarked image (Button 4).

Then Bob may later download the picture and insist to the teacher that he painted it. Now the teacher should pretend that he is going to reward that student for that picture but also asks him if he can prove that it is really his image. Of course Alex that really painted that image would not want another one to be awarded with a grade for a picture that he painted. So Alex also gets into the discussion by telling his teacher that there is a watermark in that picture and also which is the value of this watermark. This way he can prove that it is his property whereas Bob won't be able to do that. Then Alex uses the WaterIP program again, inserts the picture that is being claimed by both sides and gets the marked picture. Last Alex applies the extracting algorithm, gets printed the marked image, and using a ruler for his help gets the permutation. That proves to the teacher that the picture was indeed painted by Alex as the watermark he was told previously matches with the one that was just extracted from the picture. Last the teacher rewards Alex for his picture.

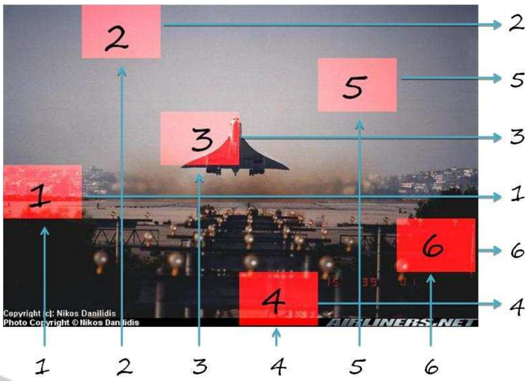
2
5
3
1
6
4
2
5
3
1
6
4
1
2
3
4
5
6
Copyright (c): Nikos Danilidis
Photo Copyright © Nikos Danilidis
AIRLINERS.NET

Figure 3: The marked image I_m.

Figure 3 shows the most important part of the whole process. After Alex runs the marking algorithm he has a picture with red films over the marked areas and, using a ruler and a pencil, he follows the extracting procedure:

Procedure Water_Extract

1. Scan the picture with the ruler from left to right and write the numbers from 1 to 6 in ascending order inside the red films;
2. Get the watermark by scanning the picture from top to bottom and read every number that comes across;

It is worth mentioning that after the completion of the above process a student has experienced, he can then easier understand what intellectual property really stands for.

REFERENCES

Collberg, C. and Nagra, J. (2010). *Surreptitious Software*. Addison-Wesley.

Golumbic, M. (1980). *Algorithmic Graph Theory and Perfect Graphs*. Academic Press, Inc., New York.

Gonzalez, R. C. and Woods, R. E. (2007). *Digital Image Processing*. Prentice-Hall, 3rd edition.

Sedgewick, R. and Flajolet, P. (1996). *An Introduction to the Analysis of Algorithms*. Addison-Wesley.

Watkins, C., Carnell, E., and Lodge, C. (2007). *Effective Learning in Classrooms*. Sage Publications.