University of Florence
School of Engineering

Master degree program in
Computer Engineering

Disparity coherent stereo video watermarking

Master Thesis of

Benedetta Barbetti, Michaela Servi

December 2015

Supervisor:

Prof. Alessandro Piva

Advisors:

Prof. Carlo Colombo
Dott. Pasquale Ferrara
Dott. Francesca Uccheddu

Academic Year 2014/2015

Abstract

Nowdays stereoscopic videos play an important role in many applications: from medical diagnosis and endoscopic surgery to fault detection in manufactory industry, army and arts, in people tracking and mobile robotics navigation as, naturally, in the film industry with 3D movie release.

The huge increase of distribution system of this content leads to the increase of concerns over content copyright protection: in this thesis a disparity-coherent watermarking technique has been presented to protect stereoscopic video contents.

The algorithm belongs to view-based methods and operates in both frequency and spatial domain in a diparity-coherent way, namely, a physical point of the captured scene always carries the same watermark sample regardless of where it appears in the left and right views.

This kind of techniques has been proved to yield less visual discomfort and to be robust against view synthesis attacks, as shown by the experiments conducted on the implemented algorithm.

Contents

List of Figures

List of Tables

Introduction

In the last few years the stereoscopic technique has become a great part of image and video processing.

In medical diagnosis and endoscopic surgery [?] [?] as in fault detection in manufactory industry, army and arts, multiview imaging is considered as a key enabler for professional added value services.

Nowdays stereoscopic techniques are also used in people tracking [?] and mobile robotics navigation [?] for economic reasons and to improve performances.

Finally the worldwide success of 3D movie releases and 3D video games [?] and the deployment of 3D televisions made the nonprofessional user aware about a new type of multimedia entertainment experience.

The increasing production and distribution of these contents leads to the concerns over copyright protection.

Digital watermarking can be considered as the most flexible property right protection technology, since it adds some information (a mark, i.e. copyright information) in the original content without altering its visual quality so that such a marked content can be further distributed/consumed by another user without any restriction; still, the legitimate/illegitimate usage can be determined at any moment by detecting the mark. In same case the watermarking protection mechanism, instead of restricting the media

copy/distribution/consumption, provides means for tracking the source of the content illegitimate usage.

The purpose of this thesis is to provide a new watermarking system for copyright protection of stereoscopic videos.

The method operates in the frequency and in the spatial domain by embedding a pseudo-random sequence of real numbers in a selected set of DFT coefficients of the left image; then the reference watermark is distorted according to the depth information prior to insertion and spatially added to the right image.

This new algorithm is robust against view synthesis and lossy compression.

In Chapter 1 the stereoscopic video context is presented, specifically the devices used to capture the scene and to display it, and the stereoscopic vision background.

In Chapter 2 an overview of the digital watermarking process is presented.

Chapter 3 and 4 present a new correlation-based detection for spatial disparity-coherent watermaking technique and a new disparity-coherent watermarking techinique which works in the frequency domain, respectively.

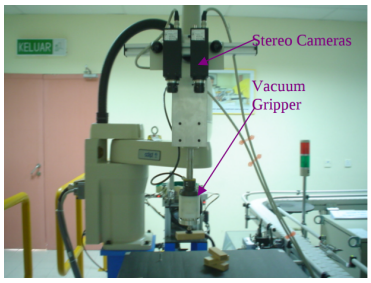
Finally in Chapter 5 the experimental results conducted on the new algorithms are presented.

Chapter 1

Stereoscopic Video

Depth information is essential for an accurate image analysis or for enhancing the realism.
In a wide variety of image processing applications, explicit depth information is required in addition to general image informations, such as intensities, color, densities.
Examples of such applications are found in 3D vision (robot vision, photogrammetry, remote sensing systems), in medical imaging (computer tomography, magnetic resonance imaging, microsurgery), in remote handling of objects (random bin picking), in space exploration (mobile robotics navigation) or 3D movies and videogames (Figures 1.1 and 1.2).
In remote sensing the terrain's elevation needs to be accurately determined for map production, in remote handling an operator needs to have precise knowledge of the threedimensional organization of the area to avoid collisions and misplacements.

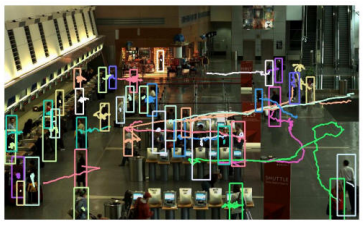
Depth in real world scenes can be explicitly measured by a number of range sensing devices such as by laser range sensors, by structured light or



(a) In bin picking applications [?] stereo vision helps to reconstruct the 3D environment and detect the part of the object to be robotically picked

(b) Surgical robot *Da vinci* is provided with a stereoscopic camera that allows a tridimensional view of the operative filed.

Figure 1.1: Stereoscopy in medical and industrial field



(a) In people tracking application stereo vision improves segmentation thanks to depth information and it's less sensible to light changes.

(b) In mobile robotics navigation stereo vision has became the first choice technology because it provids a lot of quality data for low costs.

Figure 1.2: Stereoscopy application's fields

by ultrasound. However it's usually undesirable to have separate systems for acquiring the intensity and the depth information because of the relative low resolution of the range sensing devices and because it's not an easy task to fuse information from different type of sensors; for these reasons and for a non-negligible economic factor stereoscopic vision has becoming the technology of choice in these type of applications.

(a) Stereo video frames, left and right.

(b) Overlap of the two frames.

(c) 3D view with specific glasses

(d) Polarized glasses for 3D view

Figure 1.3: Stereoscopy in 3D video games

1.1 3D capturing devices

For stereoscopic shooting, two synchronized cameras must be used [**?**]. The distance between the center of the lenses of the two cameras is called the interaxial, and the cameras' convergence, is called the angulation. These two parameters can be modified according to the expected content peculiarities.

The two cameras must be correctly aligned, identically calibrated (i.e.

Figure 1.4: Interaxial separation between lenses

brightness, color, etc...) and perfectly synchronized (frame-rate and scan-wise).

To hold and align the cameras, a stereo-rig is used; the rigs can be of two main types:

- the side-by-side rig, where the cameras are placed side by side (Figure 1.5a). This kind of 3D-rig is mostly useful for large landscape shots since it allows large interaxials; however, it doesn't allow small interaxials because of the physical size of the cameras;

(a) Syde-by-syde rig

(b) Beamsplitter rig

(c) Monoblock camera

Figure 1.5: Professional technologies for 3D TV

- the beamsplitter rig (Figure 1.5b), where one camera films through a semi-transparent mirror, and the other films the reflection in the mirror. These rigs allow small and medium interaxials, useful for most shots, but not the very large interaxials (because the equipment would be too large and heavy).

Monoblock cameras have been designed as well, where the two cameras are presented in a fixed block and are perfectly aligned, which avoids cameras desynchronization (Figure 1.5c).

A second category of 3D shooting devices is presented in Figure 1.6. These electronic devices are less expensive and are targeting the user-created stereoscopic picture/movie distribution.

An other important category of 3D image capture devices it's the one em-

(a) Mobile phones

(b) Digital cameras

(c) Camcorders

Figure 1.6: Digital personal stereo acquisition systems

ployed in the robotics and automation field, Figure 1.7, [?] [?]. They are usually impressively precise, cost-efficient and fast.

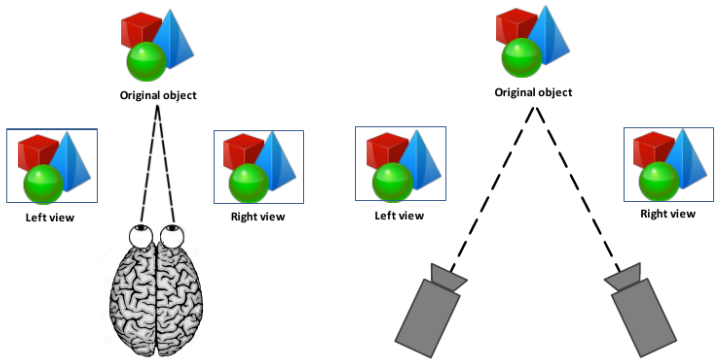
(a) The Bumblebee series from Point Grey

(b) Human wearable stereo device

Figure 1.7: Industrial and robotic stereo cameras

1.2 Stereo vision

In image processing stereo vision is the process of extracting 3D information from multiple 2D views of a scene [?].
The 3D information can be obtained from a pair of images, also known as a stereo pair, by estimating the relative depth of points in the scene.
From the anatomic point of view, the human brain calculates the depth in a visual scene mainly by processing the information brought by the images seen by the left and the right eyes. These left and right images are slightly different because the eyes have biologically different emplacements.
Consequently, the straightforward way of achieving stereoscopic digital imaging is to emulate the Human Visual System (HSV) by setting-up (under controlled geometric positions), two traditional 2D cameras.



(a) Binocular human visual system (b) Stereoscopic system

Figure 1.8: Binocular human vision vs. stereoscopic content acquisition.

1.2.1 Background

In order to be able to perceive depth using recorded images, a stereoscopic camera is required, which consists of two cameras that capture two different,

horizontally shifted perspective viewpoints; with two (or more) cameras we can infer depth, by means of triangulation, if we are able to find corresponding points in the two images (Figure 1.9).

The camera setup should be geometrically calibrated such that the two

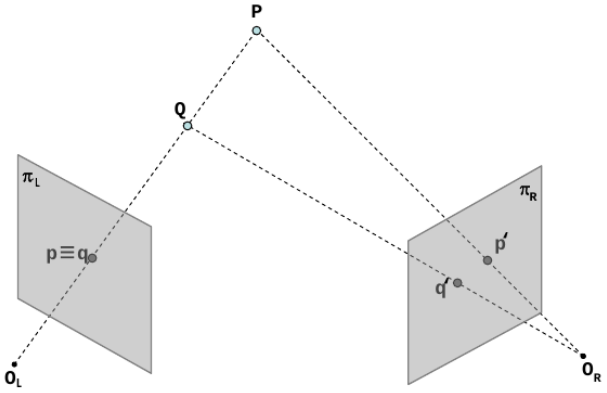


Figure 1.9: Triangulation: with two cameras the depth of P is estimated if corrispondent points are find in both images

cameras capture the same part of the real world scene.
Calibration of a stereo camera system involves the estimation of the intrinsic and extrinsic parameters of the model [?]: intrinsic parameters embody the characteristics of the optical system and its geometric relationship with the image sensor, extrinsic parameters relate the location and orientation of the second camera with respect to the first one in the 3D space (Figure 1.10).

These parameters can be used to rectify a stereo pair of images to make them appear as the two image planes are parallel (Figure 1.11); once the images are rectified, epipolar geometry it's used to find corresponding points and compute the disparity map [?].

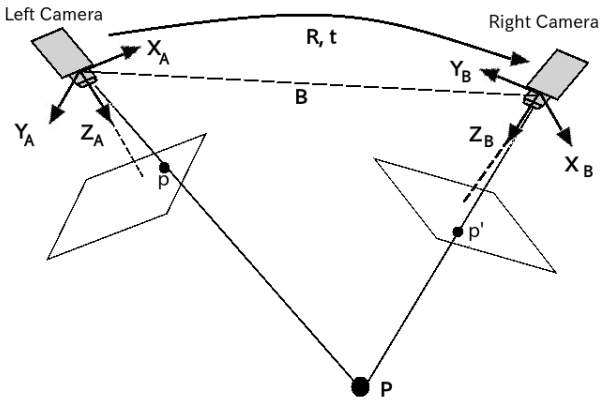


Figure 1.10: Stereo camera model

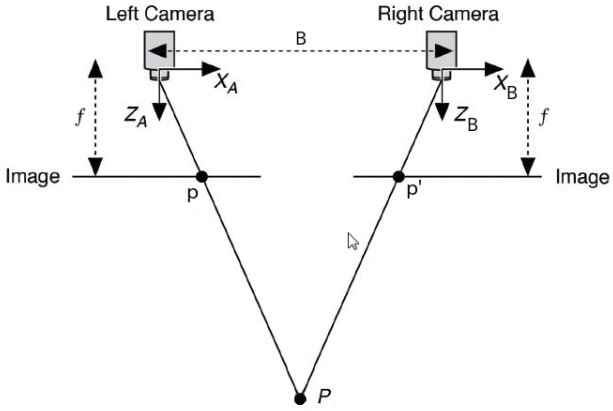


Figure 1.11: Rectified stereo cameras

1.2.2 Disparity map computation

With the stereo rig in standard form and by considering similar triangles in Figure 1.13 (PO_LO_R and Ppp') the following relations hold:

$$\frac{B}{Z} = \frac{(B + x_L) - x_R}{Z - f} \tag{1.1}$$

so

$$Z = \frac{B \cdot f}{x_L - x_R} = \frac{B \cdot f}{d} \tag{1.2}$$

where $d = x_L - x_R$ it's called *disparity*.

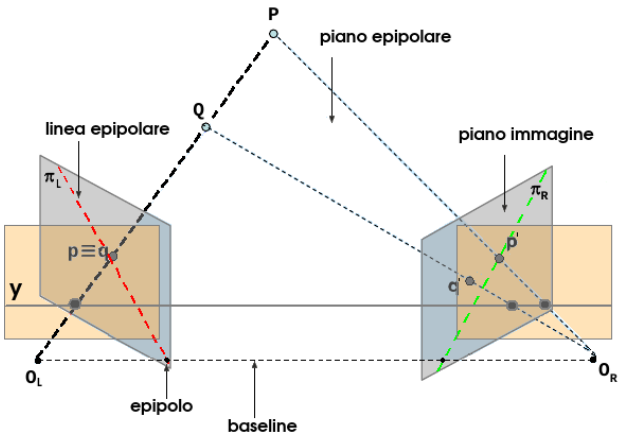


Figure 1.12: Rectified images: corresponding points (p, p'), projection of the same 3D point (P) are constrained on the same image horizontal line, the epipolar line.

Disparity is, therefore, the difference between the x coordinates of two cor-

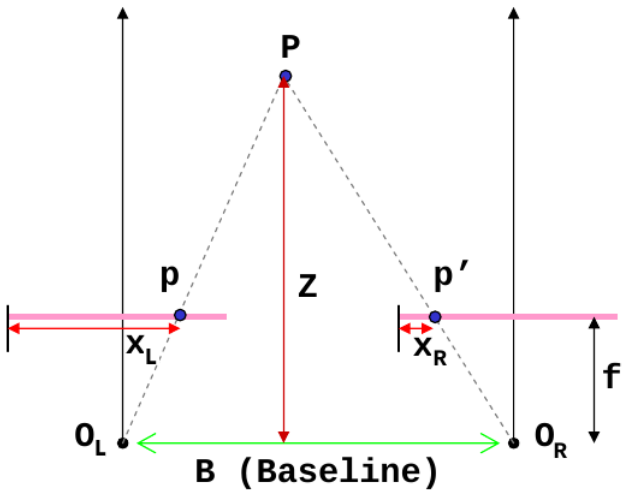
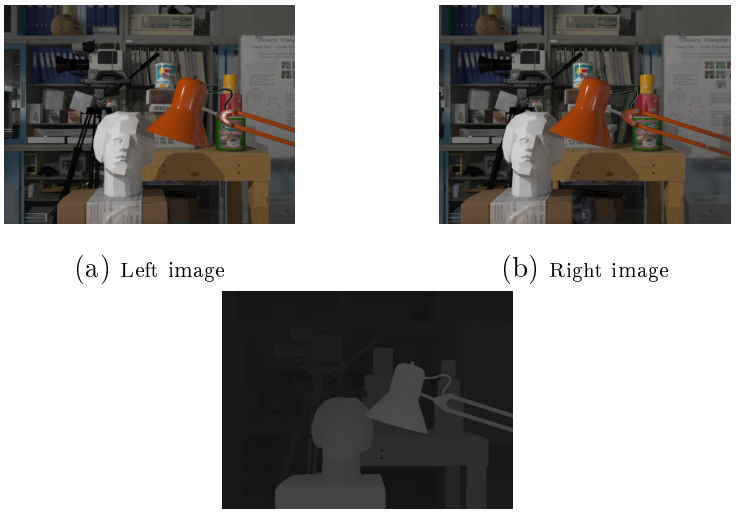


Figure 1.13: Geometry of standard form

responding points and it is usually encoded with greyscale image (Figure 1.14c), where points closer to the cameras are brighter and correspond to a higher disparity.

In order to compute the disparity map is necessary to find corresponding



(a) Left image (b) Right image

(c) Disparity map

Figure 1.14: Stereo pair and disparity map

points; stereo correspondance is though a challenging task that has to manage with perspective distortions, uniform and ambiguous regions, repetitive patterns, occlusions and discontinuities(Figure 1.15).

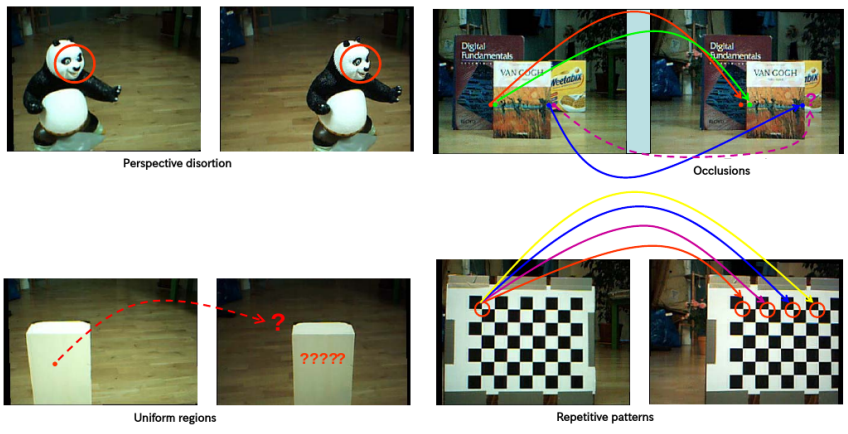


Figure 1.15: Stereo matching general problems

In general, stereo matching algorithms can be categorized into two major classes:

- local methods

- global methods.

Local stereo algorithms estimate the correspondence using a local support region or a window. Local algorithms generally rely on an approximation of the smoothness constraint assuming that all pixels within the matching region have the same disparity. However, this assumption is not valid for highly curved surfaces or around disparity discontinuities.

A naive approach consists of comparing each pixel or window in the left image with every pixel or window on the same epipolar line in right image and picking position with minimum match cost (e.g., SSD, SAD, normalized correlation).

Global stereo methods consider stereo matching as a labeling problem where

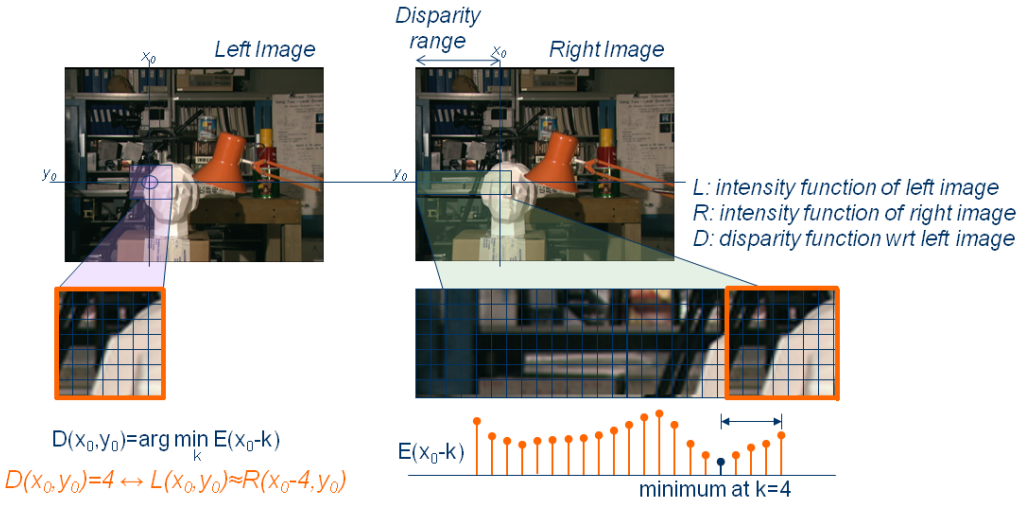


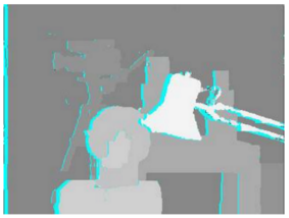
Figure 1.16: Local stereo matching window based

the pixels of the reference image are nodes and the estimated disparities

are labels. An energy functional embeds the matching assumptions by its data, smoothness, and occlusion terms and propagates them along the scan line or through the whole image. The labeling problem is solved by energy functional minimization, using dynamic programming, graph cuts, or belief propagation.
Even if this class of algorithms is significantly slow, the results, especially when textures and discontinuities are present, are much accurate.

In this thesis the Kolmogorov and Zabih's Graph Cuts Stereo Matching Algorithm, [?], has been used, because there were no time constraints requirements and the quality of the computed disparities has been considered satisfying with regard to the ground trouth.

(a) Left image



(b) Graph cuts' disparity map

(c) Ground truth disparity map

Figure 1.17: Results of the Kolmogorov and Zabih's graph cuts algorithm on the Tsukuba pair

In this algorithm the correspondence problem is addressed by constructing a problem representation and an energy function that takes into account the *uniqueness* of a configuration.

A configuration f is any map $f : \mathcal{A} \to \{0, 1\}$, where $\mathcal{A}$ is the set of pair of pixels (p, q), (p pixel of left image and q pixel of right image), which may potentially correspond. If $a = (p, q)$ is an assigment, then $f(a) = 1$ means that p and q correspond under the configuration f.

A configuration is *unique* if for all pixels p (resp. q), there is at most one active assignment involving p (resp. q): for instance, considering p, if $f(p, q1) = f(p, q2) = 1$, then $q1 = q2$. A pixel that correspond to no pixel in the other image is labeled as occluded.

The energy of a configuration f is defined as:

$$E(f) = E_{data}(f) + E_{occlusion}(f) + E_{smoothness}(f) + E_{uniqueness}(f) \qquad (1.3)$$

where each term promotes a desired property of the configuration: the data term measures how well matched pairs fit, the occlusion term minimizes the number of occluded pixels, the smoothness term penalizes the nonregularity of the configuration, and the last term enforces the uniqueness.

Since this energy funtion is not graph-representable, his minimization can be approximated by an iterated constrained minimization, given by so-called expansion moves [**?**]]; given this changes, the energy assumes a new expression, $E_{f,\alpha}$.

The minimal cut of a graph that represents the energy $E_{f,\alpha}$ is then found. The problem is NP-hard, so a local minimum is computed.

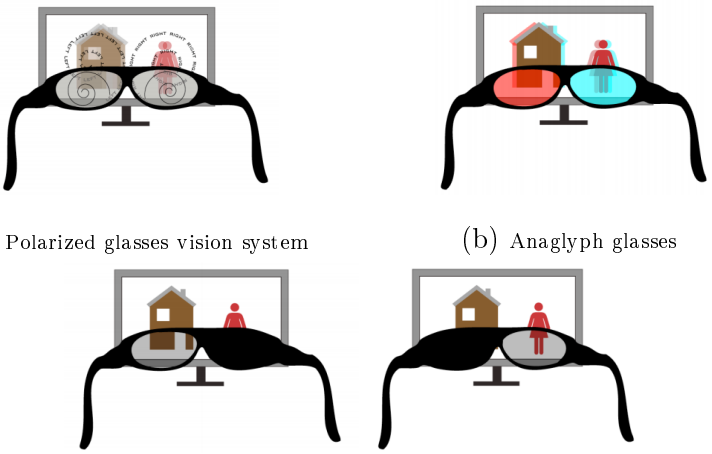
1.3 3D video displays

The basic technique of stereo displays is to present offset images that are displayed separately to the left and right eye. Both of these 2D offset images are then combined in the brain to give the perception of 3D depth.

For stereoscopic 3D displays the viewer needs to wear special glasses which separate the views of the stereoscopic image for the left and the right eye. These 3D glasses can be active or passive [?] [?].

On the one hand, active glasses are controlled by a timing signal that allows to alternatively darken one eye glass, and then the other, in synchronization with the refresh rate of the screen. Hence presenting the image intended for the left eye while blocking the right eye's view, then presenting the right-eye image while blocking the left eye, and repeating the process at a high speed which gives the perception of a single 3D image(Figure 1.18c). This technology generally uses liquid crystal shutter glasses(Figure 1.19a).

On the other hand, passive glasses are polarization-based systems and con-



(a) Polarized glasses vision system

(b) Anaglyph glasses

(c) LCD shutter glasses glasses vision system

Figure 1.18: 3D-TV visual systems

tain a pair of opposite polarizing filters; each of them passes light with similar polarization and blocks the opposite polarized light (Figure 1.19b). In circular polarization each image is circularly polarised by the display and shown together, the left eye is polarised clockwise and the right eye is polarised anticlockwise. The glasses also have a circular polarising filter for each eye, the left lens filters or blocks out the right eye image and the right lens filters or blocks out the left eye image (Figure 1.18a).

In linear polarization passive 3D TV screens sport a filter with alternating horizontal and vertical stripes, separated by a black, picture-blanking bars. When used with glasses which have corresponding polarising lenses, alternate frames are presented to each eye to create a 3D image.

The color anaglyph-based systems are a particular case of the passive glasses and use a color filter for each eye, typically red and cyan, Figure 1.19c . The anaglyph 3D image contains two images encoded using the same color filter, thus ensuring that each image reaches only one eye (Figure 1.18b).

(a) LCD shutter glasses

(b) Polarized glasses

(c) Anaglyph glasses

Figure 1.19: Passive and active glasses for 3D viewer technologies

Chapter 2

Stereo video watermarking

2.1 Watermaking

Digital watermarking consists in imperceptiby and persistently associating some extra information with some original content.
The basic watermarking workflow is presented in Figure 2.1.

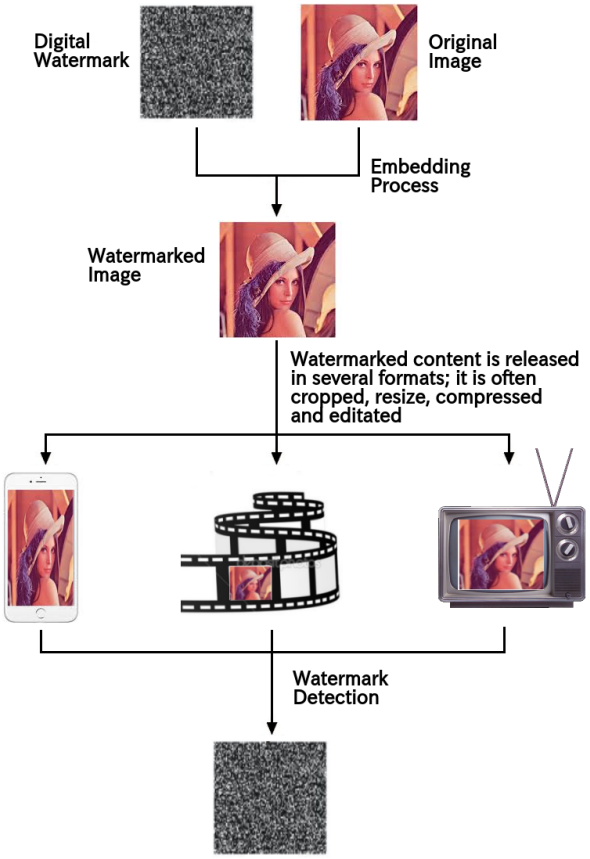


Figure 2.1: Watermarking workflow

2.1.1 Properties

Three parameters are required to evaluate watermarking technique performances:

- perceptual impact, that is the misure of how much the watermark affects the quality of the host data;
- robustness, i.e.,the capability of the hidden data to survive host signal manipulation including compression, signal processing, geometric manipulations;
- data payload, that is the amount of data of information bits that it is able to convey.

These requirements are though inversely proportional (Figure 2.2): the more information is embedded, the more the watermark is visible and viceversa; the more robusteness is encreased, the more the watermark is visible and viceversa.

Finally, a watermarking technique can be:

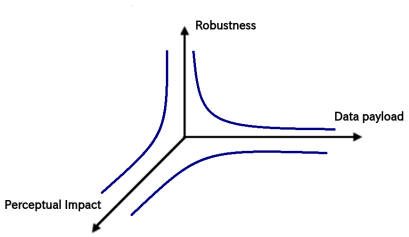


Figure 2.2: Watermark properties trade-off

- non-blind/blind, if at the decoder side the original content is available or not, respectively;

- private/public if only authorized users can recover it or if anyone to read the watermark, respectively;

- detectable/readable, if it is only possible to decide whether a given watermark is embedded in the content or if the bits hidden in the content can be read without knowing them in advance, respectively.

2.1.2 Embedding domains

Host features modified during embedding can belong to

- spatial domain: the watermark is embedded by directly modifying the pixel values;

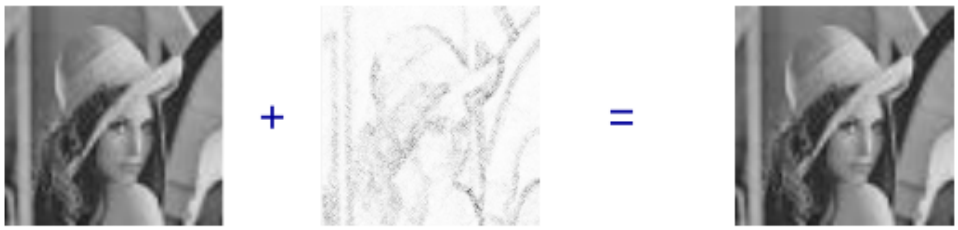


Figure 2.3: Spatial domain watermark insertion

- frequency domain: the image is transformed through a mathematical transformation, some coefficients are modified and finally the inverse transform is carried out;

- hybrid techniques: a block wise transform is applied, the image is divided into blocks and for each block a mathematical transformation is computed, some coefficients are modified and the inverse transform is done.

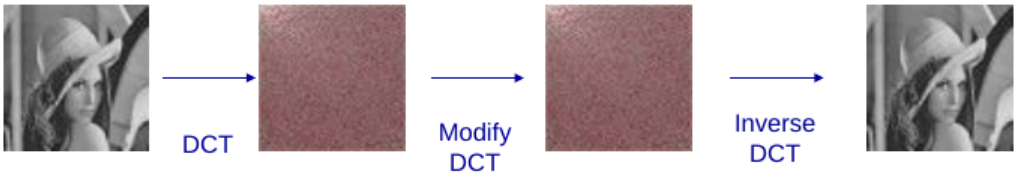


Figure 2.4: Frequency domain watermark insertion

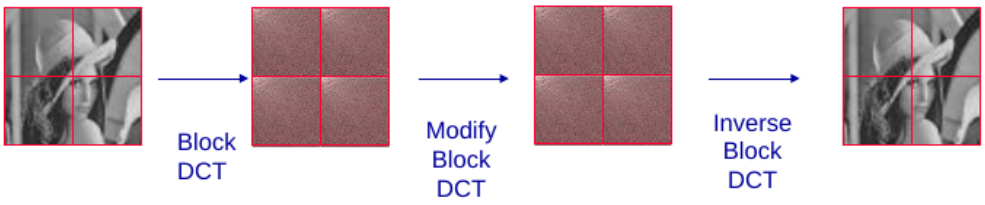


Figure 2.5: Hybrid technique

2.1.3 Embedding techniques

The most straightforward ways to add a watermark in a given content have been proved to be Spread Spectrum (SS) approach and Side Information (SI). As in spread spectrum communications, the former approach considers the

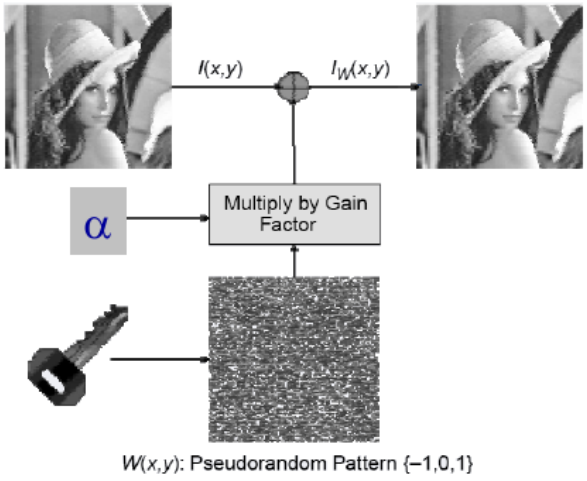


Figure 2.6: Spread spectrum technique

original content as a signal and the watermark as a noise that is spread over very many frequency bins so that the energy in any one bin is very small and certainly undetectable [?] [?].
The latter takes advantage of the fact that the original content is known at the embedder side (but unknown at the detector): this way the watermark can be modulated according to the original and the quantity of inserted data can be maximed [?, ?, ?, ?].

Sometimes hybrid watermarking methods combining spread spectrum and side information concepts can be applied; they try to benefit from both the robustness and transparency of the spread spectrum methods and the increased data payload of the side information methods [?] [?].

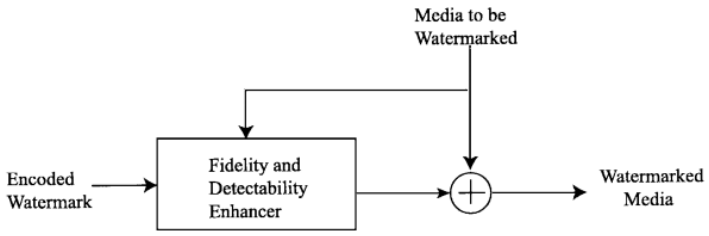


Figure 2.7: Side information technique scheme

2.2 Stereoscopic video watermarking

In the literature, stereoscopic video watermarking has been initially approached as a direct extension of still image watermarking, i.e. by considering the right and the left views as two independent images. This way, the stereo data can be straightforward exploited with basic 2D methods. However, such straightforward application does not consider the peculiarities of the stereoscopic video content, therefore a second modality considers derived

representations from the stereo pair, as a disparity map.
A new approach, however, has been recently introduced in stereoscopic view-based methods: disparity-coherent watermarking [?].

2.2.1 Embedding domain

In stereoscopic video context the studies can be structured in two other categories in addition to spacial and frequency domain:

- view-based methods [?, ?, ?, ?, ?, ?];

- disparity-based methods [?]

according to the reference image in which the mark is actually inserted.
In Figure ?? the workflows of both methods are presented.

The predilection direction in the literature is represented by the view-based watermarking approaches, which are currently deployed for stereoscopic still images.
In this context disparity-coherent watermarking has been introduced, [?], to provide superior robustness against virtual view synthesis, as well as to improve perceived fidelity.
Disparity-coherence refers to the fact that a physical point of the captured scene should carry the same watermark sample regardless of where it appears in the left/right view.
The advantages of producing disparity-coherent watermarks are two: first it produces pairs of stereoscopic views that are more in line with what would naturally occur in reality and thereby yields less visual discomfort, second disparity-coherent watermarks are expected to exhibit superior robustness against view synthesis, [?].

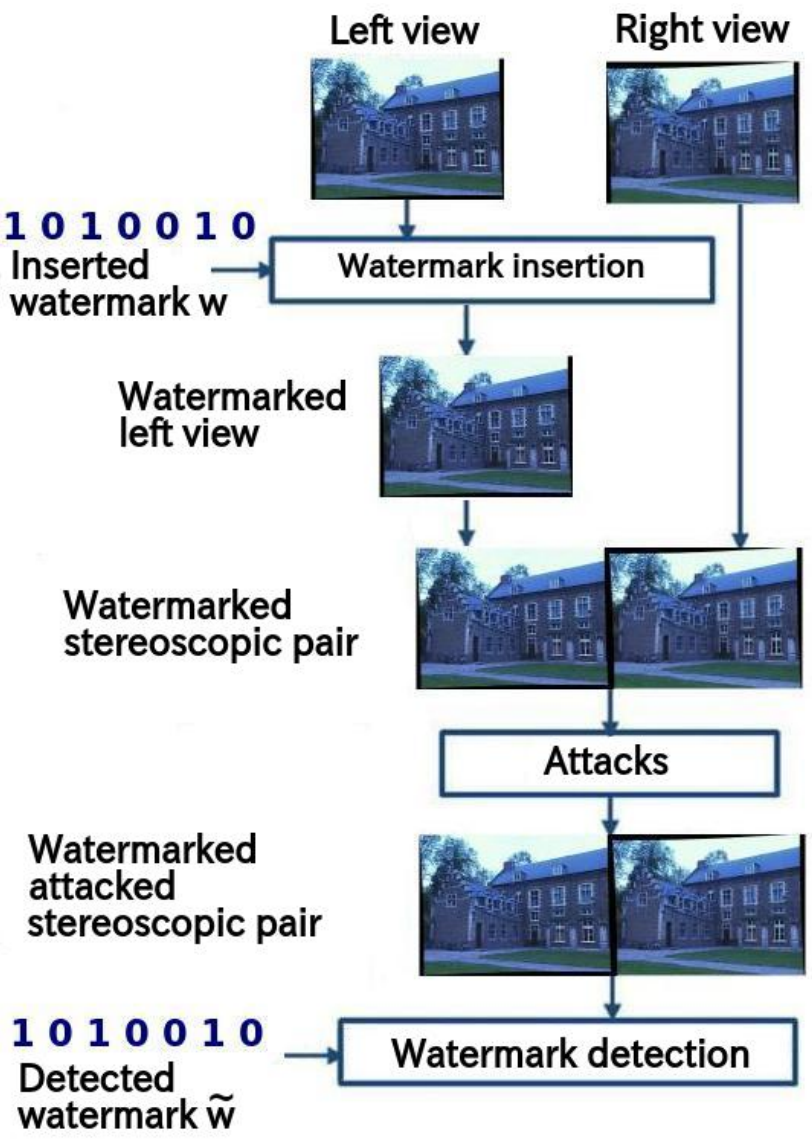


Figure 2.8: View-based watermarking workflow

View synthesis consists in generating a virtual view in-between views that are available, e.g.the left and right views in stereo video.

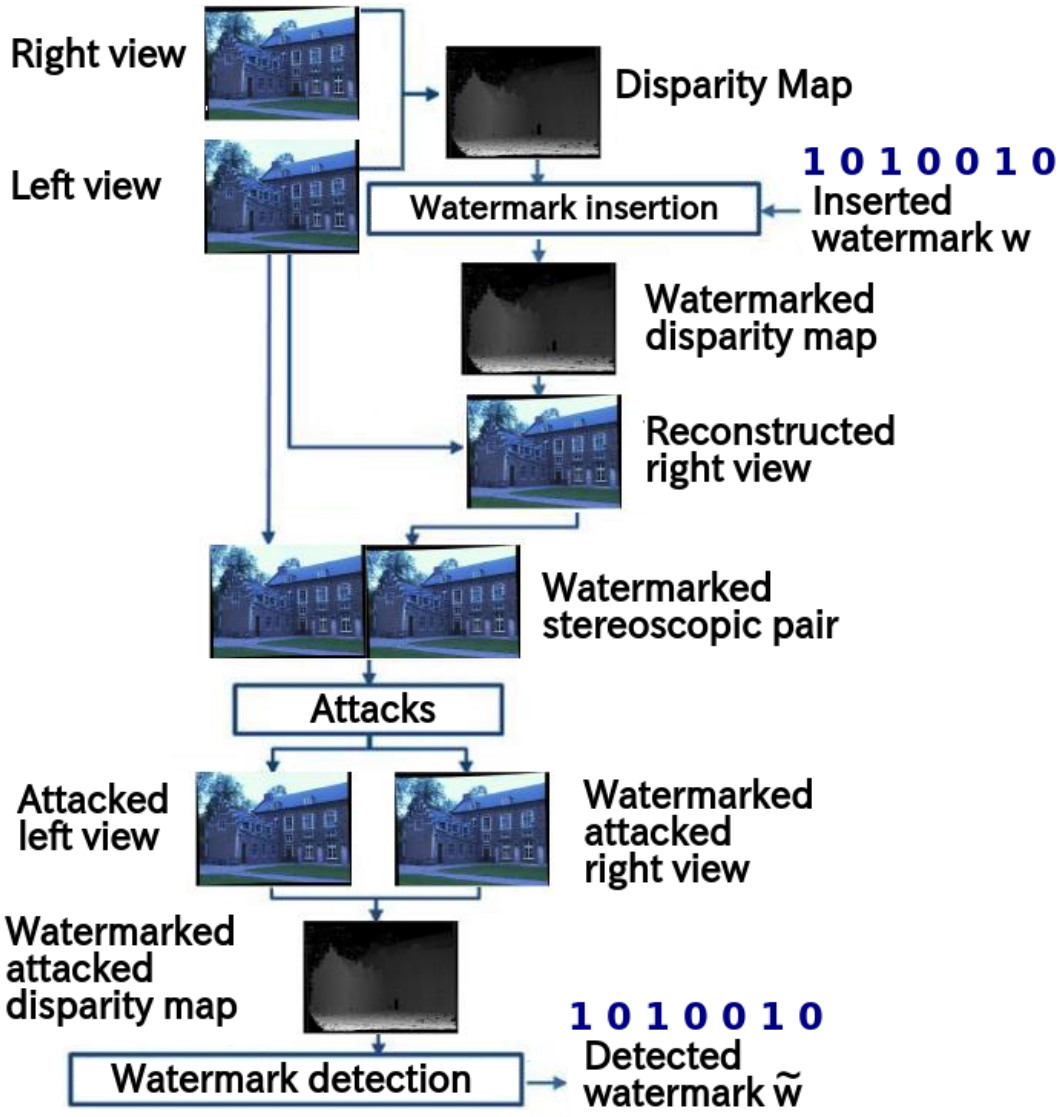


Figure 2.9: Disparity-based watermarking workflow

2.2.2 Perception evaluation

Perceptual impact can be defined as the imperceptibility of the embedded additional information in the watermarked content. This may signify either that the user is not disturbed by the artefacts induced by the watermark in the host document or that the user cannot identify any difference between

the marked and the unmarked document.
The visual quality of the watermarked content in images and 2D video is usually objectively evaluated by five objective measures, namely, the PSNR, IF, NCC, SC, and SSIM [**?**].

In this thesis the measures in [**?**] have been used to evaluate the quality of the watermarking technique in terms of human perception.
In Chaminda et al.'s study a Reduced-Reference (RR) quality metric for color plus depth 3D video compression and transmission is proposed, using the extracted edge information of color plus depth map 3D video.
The work is motivated by the fact that the edges/contours of the depth map can represent different depth levels and this can be considered for measuring structural degradations. Since depth map boundaries are also coincident with the corresponding color image object boundaries, edge information of the color image and of the depth map is compared to obtain a quality index (structural degradation) for the corresponding color image sequence.
In order to quantify structural comparison, luminance comparison and contrast comparison parameters for the depth map and corresponding watermarked views, a modified version of the commonly used SSIM metric is adopted:

$$Q_{Depth}(x,y) = [l(x,y)]^{\alpha} \cdot [c(x,y)]^{\beta} \cdot [S_{Depth}(x',y')]^{\gamma} \tag{2.1}$$

where $l(x,y)$ and $c(x,y)$ are luminance and contrast comparisons performed on original depth maps and the ones computed after watermarking, respectively, and $S_{Depth}(x',y')$ is the structural comparison between the gradient/edge maps of original and post-watermarking computed depth map images.

Then the overall depth map quality is calculated as

$$MQ_{Depth}(X,Y) = \frac{1}{M}\sum_{j=1}^{M} Q_{Depth}(x_j, y_j). \tag{2.2}$$

The SSIM-based quality index for the color image can be described as follows:

$$Q_{View}(x,y) = [l(x,y)]^{\alpha} \cdot [c(x,y)]^{\beta} \cdot [S_{View}(x',y')]^{\gamma} \tag{2.3}$$

where $l(x,y)$ and $c(x,y)$ are luminance and contrast comparisons performed on original and watermarked views, respectively, and $S_{View}(x',y')$ is the structural comparison between the gradient/edge maps of the gradient maps of the corresponding original depth map and the watermarked views.
Hence, the overall color image quality is calculated as

$$MQ_{View}(X,Y) = \frac{1}{M}\sum_{j=1}^{M} Q_{View}(x_j, y_j). \tag{2.4}$$

As in [**?**], the Sobel operator has been selected to obtain edge information (i.e., the binary edge mask) due to its simplicity and efficiency.

Finally the PSNR measure has been used to evaluate the quality of the watermarked videos and the quality of the compressed videos.
PSNR is the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation; many signals have a very wide dynamic range, therefore PSNR is usually expressed in terms of the logarithmic decibel scale (dB).
For color images with three RGB values per pixel, the definition of PSNR is the following:

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \tag{2.5}$$

$$MSE = \frac{1}{3 * MN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\Big(\mathbb{R} + \mathbb{G} + \mathbb{B}\Big) \tag{2.6}$$

with

$$\mathbb{R} = [R_I(i,j) - R_{\tilde{I}}(i,j)]^2 \tag{2.7}$$

$$\mathbb{G} = [G_I(i,j) - G_{\tilde{I}}(i,j)]^2 \tag{2.8}$$

$$\mathbb{B} = [B_I(i,j) - B_{\tilde{I}}(i,j)]^2 \tag{2.9}$$

where I and $\tilde{I}$ are the MxN reference image and the noisy approximation, respectively, and MAX_I is the maximum possible pixel value of the image (when the pixels are represented using 8 bits per sample, this is 255).
For video sequence, the average value of all frames' PNSR value is computed. Typical values for the PSNR in video compression and watermarking are between 30 and 50 dB.

2.2.3 Robustness

The robustness refers to the ability of detecting the watermark after applying some signal modifications and malicious attacks on the marked content, such as spatial filtering, additive noise, geometric transformations, lossy compression and, in stereoscopic context, view synthesis.

Spatial filtering

Linear filtering (such as blurring) and non-linear filtering (such as sharpening) are included in some image processing software: this operations remove from a signal some unwanted component or feature (Figure 2.10).

Additive Noise

The additive noise can be added to the content when applying some usual processing or when transmitting the signal over a communication channel during the broadcast (Figure 2.11).

(a) Original image

(b) Blurred image

Figure 2.10: Spatial filtering: blurring

(a) Original image

(b) Noised image

Figure 2.11: Additive noise

Geometric distortions

The geometric distortions include rotations, translations, spatial scaling, cropping and changes in aspect ratio (Figure 2.12) they commonly occur during format changes.

Lossy compression

In video analysis, lossy compression is a common operation as it helps reduce resource usage, such as data storage space or transmission capacity.
This process brings to a degradation of the image due to the compression

(a) Original image

(b) Cropped image

(c) Rotated image

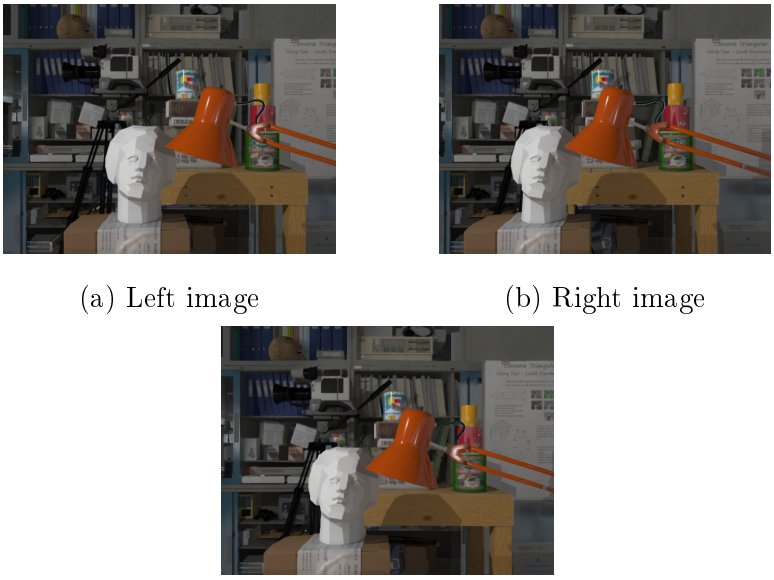
Figure 2.12: Geometric transformations

ratio, thus affects the embedded watermark, as it removes the redundancy exploited in watermarking schemes.
To prevent this problem a solution can be to improve the strenght of the embedded watermark.

View synthesis

Since in stereoscopic video context it is rather common practice to generate intermediate virtual views to adjust depth perception and since such view synthesis introduces non-rigid local geometric distortion that are not properly tackled by state-of-the art resynchronization mechanisms, stereo video watermarking strategies have to achieve robustness to synthetic view synthe-

sis (Figure 2.13).



(a) Left image (b) Right image

(c) Synthetized view

Figure 2.13: View synthesis

In this thesis a disparity-coherent watermarking algorithm has been implemented. It works in the frequency and spatial domain: a pseudo-random sequence of real numbers is embedded in a selected set of DFT coefficients of the left image then the reference watermark is spatially inserted in a disparity-coherent way in the right view.

It has shown good results in quality measure tests and roubustness test against view synthesis and compression.

An optimum criterion to verify if a given mark is present in an image is derived based on statistical decision theory, [?], allowing a robust watermark detection without resorting to the original uncorrupted image.

Chapter 3

Spatial disparity-coherent watermarking

As said in the previous Chapter, a number of works focused on how to incorporate depth information into the perceptual shaping process of the embedded watermark.
This process allows to achive disparity-coherence and makes sure that a physical point of the captured scene carries the same watermark sample regardless of where it appears in the left and right view.
This process brings two advantages: it produces stereoscopic views more in line with reality, therefore yields less visual discomfort; and it is expected to have superior robustness against view synthesis.

3.1 Prior work

A prior work that is based on the disparity-coherent technique is the one carried on by Doerr et al in "Blind Detection for Disparity-Coherent Stereo

Video Watermarking" [?].

The watermark strategy assumes that the key-seeded reference watermark pattern $w_K \tilde{} N(0,1)$ is embedded spatially in the left view and subsequently transferred to the right one.

The watermark embedding and detection operations for the left view are therefore given by the conventional spread-spectrum equations:

$$f_l^w = f_l + \alpha w_K$$

$$\rho(f_l + \epsilon\alpha w_K, w_K) = \frac{1}{wh}\sum_{x,y}(f_l(x,y) + \epsilon\alpha w_K(x,y))w_K(x,y) \approx \epsilon\alpha$$

where the superscript w indicates watermarked quantities, the subscript l (resp. r) denotes quantities related to the left (resp. right) view, $\alpha > 0$ is the embedding strength, and w is normally distributed with zero mean and unit variance.

The embedding strenght used in [?] to keep the embedding distortion imperceptible is $\alpha = 3$.

For the right view, the watermarking equation is the same, except that the watermark pattern w_K is warped according to the depth information prior to insertion.

$$\forall(x,y) \in [1:w][1:h] f_r^w(x,y) = f_r(x,y) + \alpha w_K(x + d(x,y), y) = f_r + \alpha w_K^d(x,y)$$

The watermark detection on the right view relies on the computation of a horizontal cross-correlation array.

$$\rho(f_r + \epsilon\alpha w_K^d, w_K^s) \approx \epsilon\alpha D_s$$

$$\rho = \epsilon\alpha[D_{smin}, .., D_0, .., S_{smax}]$$

where D_s is the proportion of pixels whose disparity value is exactly equal to s.

The correlation array is then mapped into a scalar value in order to compare it with a threshold and to decide whether the tested content contains the watermark. Authors proposed three possible mapping functions:

$$\rho_{max} = \max_s \rho[s]$$

$$\sum_s \rho[s]$$

$$\sum_{|rho[s]|>\tau_\rho} |\rho[s]|$$

3.2 Gaussian-noise disparity-coherent watermarking

Based on the described technique, we propose a new spatial watermarking technique.

For the spatial watermark it's been taken under consideration the insertion of a Gaussian-noise reference watermark in an additive way.

As in Doerr et al, the left view is processed in the conventional way, with spred-spectrum equations (riferimento all'equazione); the watermark is then warped according to the disparity value and inserted in the right view (rif all'eq), taking under consideration that the occluded zones shoudn't be processed.

The added pattern and the reference images have the same size, so it should be noted that the warping process will generate a loss of marked pixel, due to the baseline's lenght.

Since the disparity map and the occclusion map are usually not available,

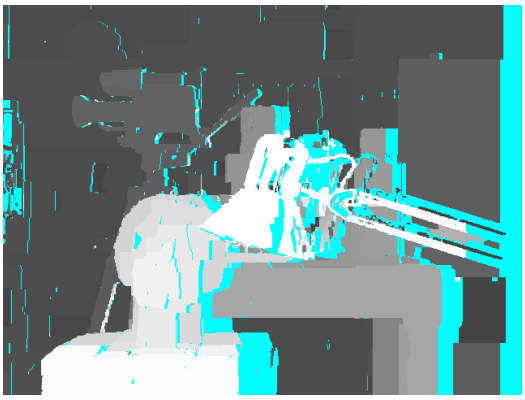


Figure 3.1: Disparity left-to-right computed with KZ

it needs to be estimated through the KZ algorithm, before the warping process.

The embedding strenght is $\alpha = 1$; it should be noted that this baseline watermarking framework could be enriched with conventional add-ons, e.g. perceptually modulate the embedding strength to better accommodate for the human visual system or canceling host interference for improved detection statistics.

In the detection process, it has been used a conventional correlation-based detector for the left view (ref to eq).

On the other hand to detect the watermark in the right view two differet correlation-based strategies are proposed: in the first strategy we computed the correlation value between the non-distorted watermark and the right view warped according to the right-to-left disparity, this way the priviously warped watermark is restored, even if there will be discontinuities due the occluded zones. In formula:

$$\rho((f_r + \epsilon\alpha w_K^*)^*, w_K) = \frac{1}{wh}\sum_{x,y}(f_r(x,y) + \epsilon\alpha w_K^*(x,y))^* w_K(x,y) \approx \epsilon\alpha$$

where the superscript * indicates the warped mark/image.

The second strategy is again a simple correlation-based detector, but the correlation value is computed between the right view and the warped watermark instead of the original one, based on the fact that the right view should contain this, rather than the reference pattern and that the reciever can compute the disparity map thats needed to warp the mark and perform the detection.

$$\rho(f_r + \epsilon\alpha w_K^*, w_K^*) = \frac{1}{wh}\sum_{x,y}(f_r(x,y) + \epsilon\alpha w_K^*(x,y))w_K^*(x,y) \approx \epsilon\alpha$$

To illustrates the performance of the binary classifier system as its discrimination threshold is varied it has been drawn the corresponding ROC curve.
The ROC curve is a representation of the sensitivity as a function of fall-out. The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.
The true-positive rate is also known as sensitivity and the false-positive rate is also known as the fall-out.

As said before the disparity-coherent watermarking have the ability to detect the embedded watermark in synthetized views: to performe the detection on a random right view, that might be synthetized, the detector will need to calculate the disparity map between the analyzed view and the recieved left, and warp it accordingly, to recompose the original watermark.
There is then a tight bond between the watermarking process and the evaluation of the disparity maps; with the graph-cuts algorithm it's possible to compute accurate maps and to know the occluded zones.

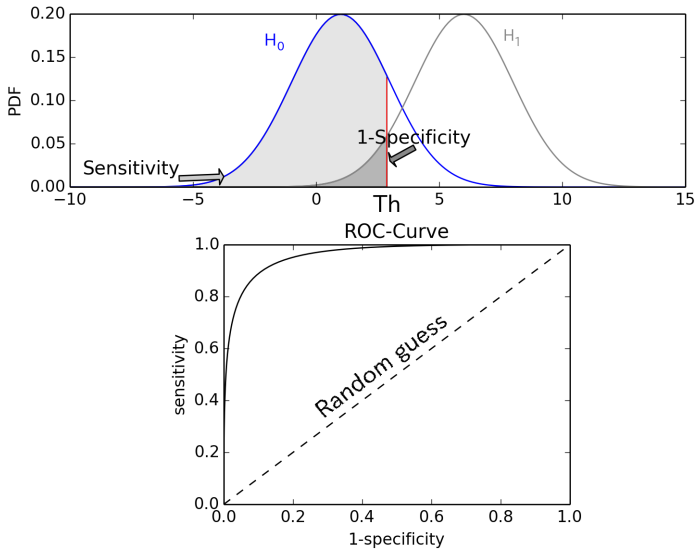


Figure 3.2: Top: Probability density functions for two distributions. Bottom: corresponding ROC-curve.

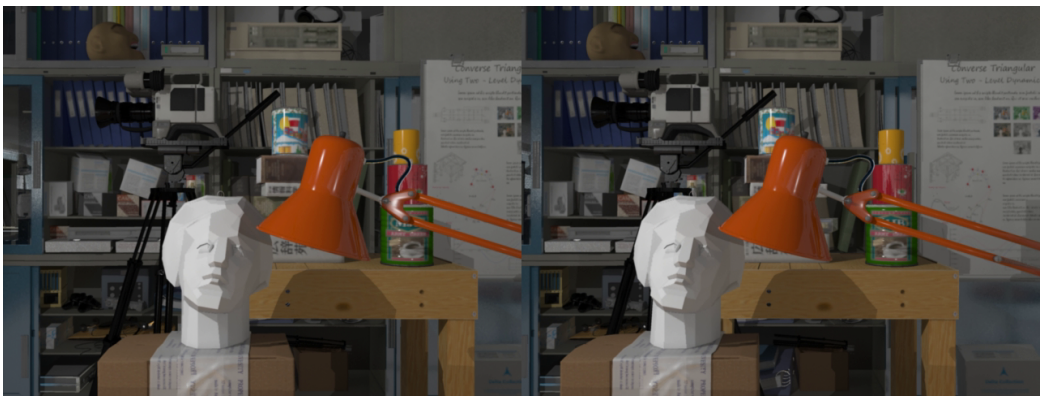


Figure 3.3: Stereo image marked with spatial algorithm with power equal to 1.

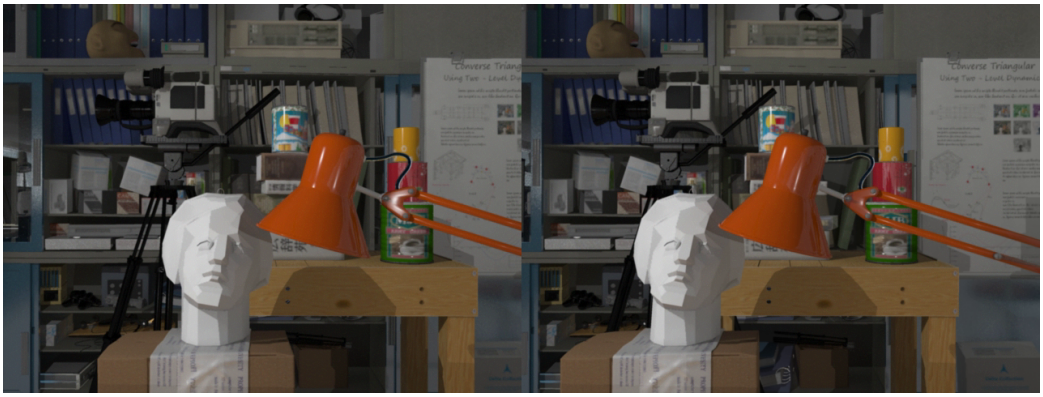


Figure 3.4: Stereo image marked with spatial algorithm with power equal to 1

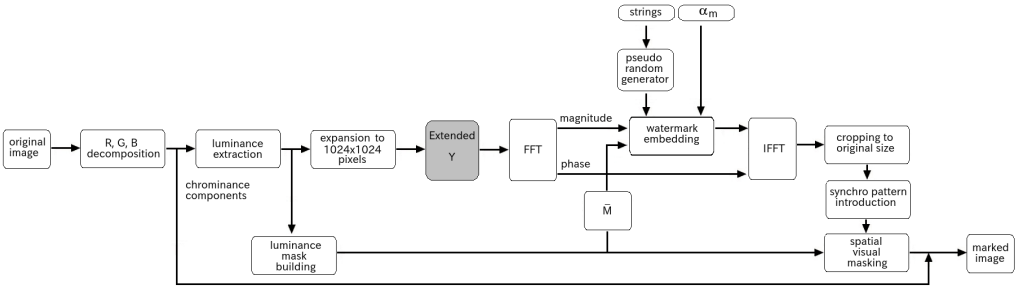
Chapter 4

Frequency disparity-coherent watermarking

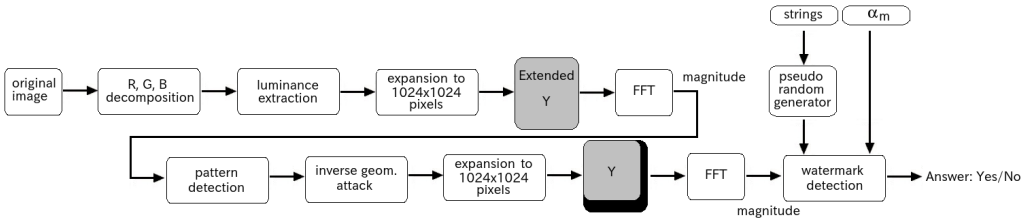
Now we proposed a variant of the described watermarking process, which works in the frequency domain.

4.1 Watermark in Fourier domain

The strategy is based on the technique presented by Piva et al in "Improving DFT Watermarking robustness through optimum detection and synchronisation" [?], where a watermarking algorithm for digital images operating in the frequency domain is presented: the method embeds a pseudo-random sequence of real numbers in a selected set of DFT coefficients of the image. Moreover, a synchronisation pattern is embedded into the watermarked image, to cope with geometrical attacks, like resizing and rotation. After embedding, the watermark is adapted to the image by exploiting the masking characteristics of the Human Visual System, thus ensuring the watermark invisibility.



(a) Watermark casting process



(b) Watermark detection process

Figure 4.1: Piva et. al watermarking workflow

For our stereo watermarking task this process has been simplified and cut to the basic frequency watermaking; the implemented steps are described below.

4.1.1 Watermark embedding

In [?] the watermark is embedded in a subset of DFT coefficients of the luminance Y.

Since a traslation of the scene will only change the phase values of the DFT, leaving unaltered the magnitude values, the watermak only concernes the latter, to achieve robustness against image traslation.

To garantee a blind detection system the number and position of the coefficient are fixed a priori: based on the size of the image to watermark, the

coefficient are choosen in the medium frequencies of the spectrum to achieve a compromise between robustness and invisibillity.
The watermark embedding rule is the following:

$$y_i^{'} = y_i + \alpha m_i y_i \tag{4.1}$$

where $y_i^{'}$ represents the watermarked DFT magnitude coefficient, y_i the corresponding original, m_i is a sample of the watermark sequence, and α is the watermark energy.
The inverted DFT is then applied to obtain the watermarked luminance $Y^{'}$.

4.1.2 Watermark detection

To determine if a given image luminance Y either embedds or not the reference watermark in [?] a threshold-based detection is used.
The luminance of the received image is extracted and its DFT trasform is computed; from the obtained magnitude matrix the right coefficents can be selected since their positions are known as said above.
Knowing the seed (in the shape of two strings, one numeric one alphanumeric) the watermark can be reproduced.

To verify if the selected coefficients have been altered by means of the watermark it is used a statistical decision theory: two hypotheses are defined, the image contains the reference watermark (hypotheses H_1) or the image does not contain this mark (hypotheses H_0). Relying on Bayes theory of hypothesis testing, the optimum criterion to test H1 versus H0 is minimum Bayes risk; the test function results to be the likelihood ratio function L that has to be compared to a threshold:

- if $L > \lambda$, the watermark m^* is present;
- if $L < \lambda$, the watermark m^* is absent.

To choose a proper threshold, it has been chosen to fix a constraint on the maximum false positive probability and the optimum decoder is designed refferring to the Neyman-Pearson criterion, as:

$$L(y) = \sum_{i=0}^{N-1}[-\beta ln(1+\alpha_m m_i^*)] + \sum_{i=0}^{N-1}[-(\frac{y_i}{\alpha_i(1+\alpha_m m_i^*)}))^{\beta_i} + (\frac{y_i}{\alpha_i})^{\beta_i}]$$

and

$$\lambda = 3.3\sqrt{2\sum_{i=0}^{N-1}[\frac{[(1+\alpha_m m_i^*)^{\beta_i}]}{(1+\alpha_m m_i^*)^{\beta_i}}] + \sum_{i=0}^{N-1}\{\frac{[(1+\alpha_m m_i^*)^{\beta_i}-1]}{(1+\alpha_m m_i^*)^{\beta_i}}\} - \sum_{i=0}^{N-1}[\beta_i ln(1+\alpha_m m_i^*)]}$$

In () $m^* = \{m_i^*\} i = 0, 1, ...N-1$ is the watermark, α_m the mean watermark energy, α_i and β_i are statistic parameters describing the probability density function shape of the magnitude of the watermarked DFT coefficients y_i.

The values of this parameters are choosen by means of Maximum Likelihood criterion, based on the the fact that the coefficients belonging to small subregions of the spectrum are characterised by the same statistic parameters and follows a Weibull distribution, modeled as:

$$f(y_i) = \frac{\beta}{\alpha}(\frac{y_i}{\alpha})^{\beta-1}\exp\{-(\frac{y_i}{\alpha})^{\beta}\}$$

In summary, the detection process can be decomposed in the following steps:

- generation of the watermark m^*;
- estimation of the parameters α, β into the regions composing the watermarked area of the spectrum;

- computation of $L(y)$ and λ ;
- comparison between $L(y)$ and λ ;
- decision.

The decoder can detect the watermark presence also in highly degraded images. In particular, the system is robust to sequences of different attacks, such as rotation, resizing, and JPEG compression, or such as cropping, resizing and median filtering.

4.2 Stereo watermarking embedding

For the stereo-marking process a 512x512 subset of pixels of the image has been considered, in particular we focused in marking the part of the scene which is common to both the left and right view.

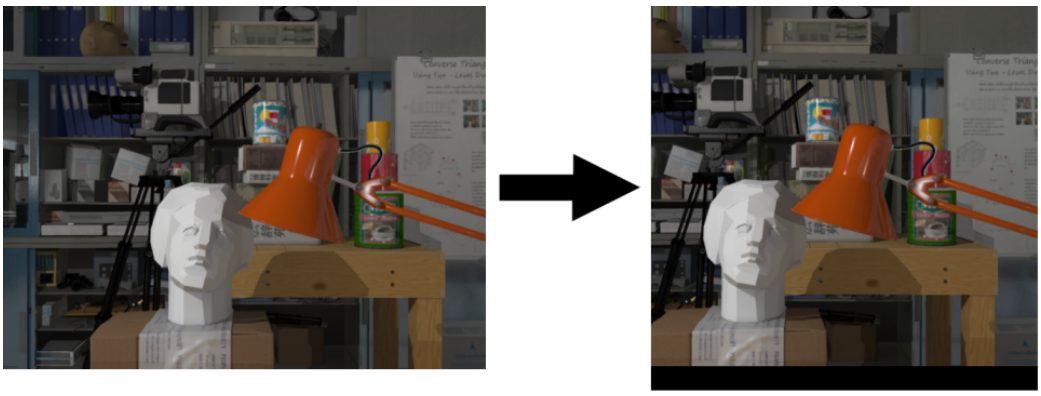


Figure 4.2: Cropping of the original image

The left view has been processed with the algorithm described in 4.1.1 (Figure 4.3).

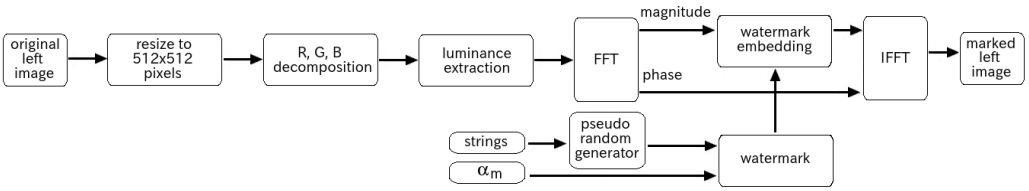


Figure 4.3: DFT watermark casting workflow of the left image

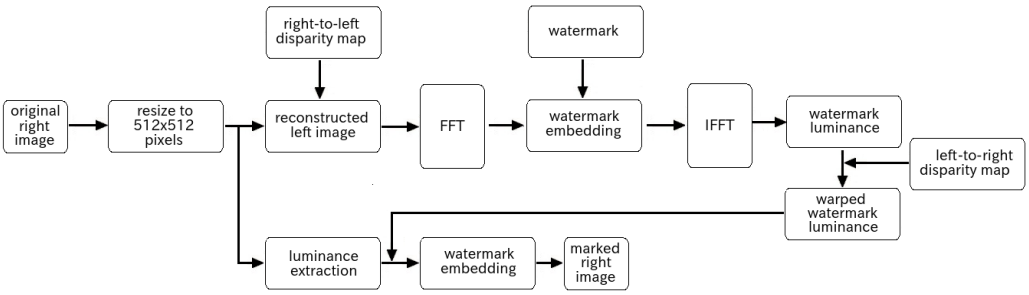


Figure 4.4: Disparity-coherent watermark casting workflow of the right view

In order to mark the left and right view with the same watermark (Figure 4.4), but in a disparity-coherent way a study on the left marking process has been conducted. The NXM left image l can be written as a function of its DFT transform L:

$$l = \frac{1}{MN} \sum \sum (|L(u,v)|) \exp\{\phi(u,v)\} \exp\{+j2\pi(\frac{ux}{M}\frac{vy}{N})\}$$

The marking process alters the DFT coefficients according to the Equation in 4.1 which can be written by:

$$l_w = \frac{1}{MN} \sum \sum (|L(u,v)| + \alpha |L(u,v)||w|) \exp\{j(\phi_L + \phi_w)\} \exp\{+j2\pi(\frac{ux}{M}\frac{vy}{N})\}$$

the signal alteration is therefore given by:

$$\alpha |L||w| exp\{j(\phi_L + \phi w)\}$$

where $|w|$ is the magnitude of the watermark, ϕ_L is the phase of the left view and ϕ_w the phase of the watermark which takes value in $\{0,\pi\}$.

To watermark the right view the pattern is created ad-hoc: the watermak's signal is generated using the phase of the left image and the phase of the reference watermark and the coefficients of the right view.
This way the right view will be marked with its coefficient, but with the correct phase, and the corresponding pixel in the left and right view will present the same alteration, not to cause visual distorsions. To obtain the same additive multiplicative alteration on the right view coefficients we created the watermark ad-hoc with the following formula:

$$\alpha|R^*||w|exp\{j(\phi_L + \phi w)\}$$

the subscript * indicates that the right image has been warped according to he right-to -left disparity to have the same phase of the left image, and generte the correct mark. The created mark is the brought in then spatial domain and warped according to the left-to right disparity previous to spatial insertion in the right view.

The complete formula can then be written as:

$$l_w = l + \frac{1}{MN}\sum\sum(\alpha|L(u,v)||w|\exp\{j(\phi_L + \phi_w)\})\exp\{+j2\pi(\frac{ux}{M}\frac{vy}{N})\}$$

$$r_w = r + \frac{1}{MN}\sum\sum(\alpha|R(u,v)^*||w|\exp\{j(\phi_L + \phi_w)\})^{**}\exp\{+j2\pi(\frac{ux}{M}\frac{vy}{N})\}$$

The subscript ** indicates the warping according to the left-to-right disparity.

4.3 Stereo detection algorithm

The detection of the watermark is performed with the detector implemented by Piva et al.

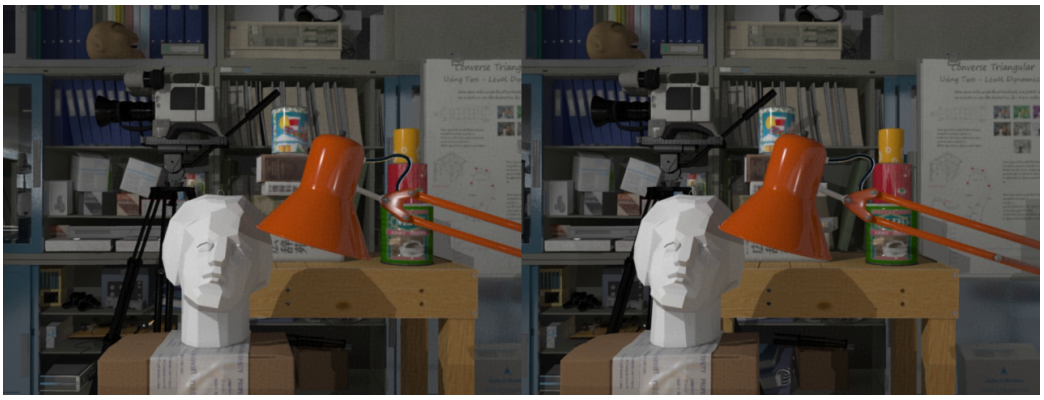


Figure 4.5: stereo image marked with DFT algorithm with power equal to 0.3

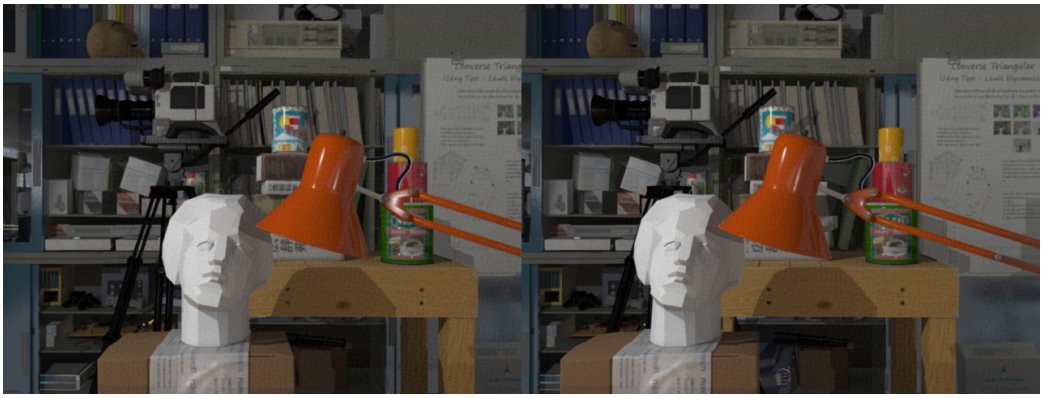


Figure 4.6: Stereo image marked with DFT algorithm with power equal to 0.5

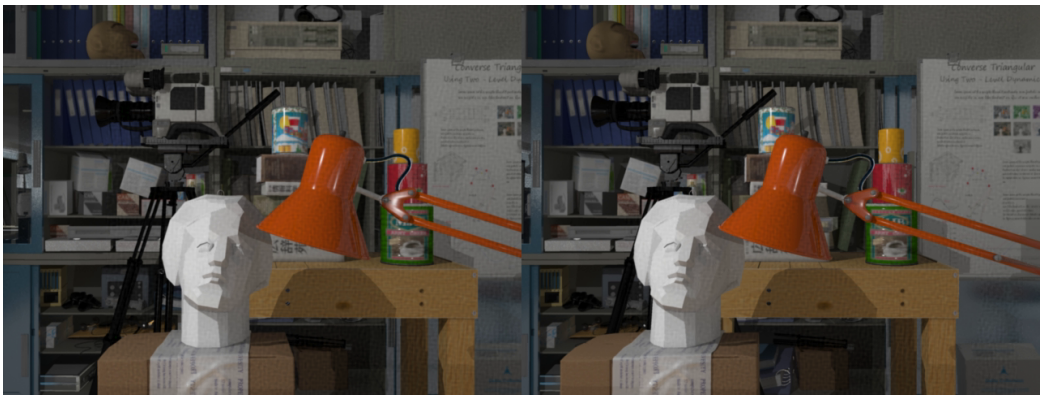


Figure 4.7: Stereo image marked with DFT algorithm with power equal to 0.6

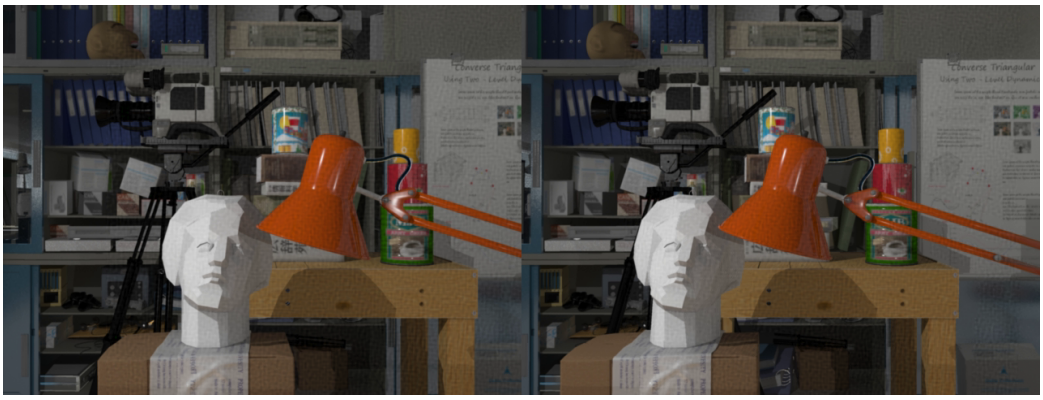


Figure 4.8: Stereo image marked with DFT algorithm with power equal to 0.7

As for the embedding process, the algorithm is applied to the left view without changes, meanwhile, some adaptations are needed for the right view detection. The detection algorithm workflow for left and right view is shown in Figure 4.9 and 4.10, respectively.

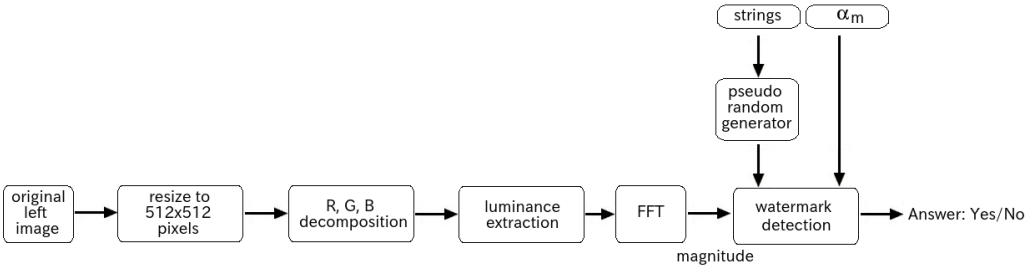


Figure 4.9: Watermark detection process for left image

First the detection algorithm computes the right-to-left disparity, then the right view is warped accordingly to ricreate the phase of the inserted watermark; to mantain the correctt phase the occluded zones are filled with the pixels of the recieved left view (taking under consideration that this little amount of image's pixel would not influence the detection).
The created image is then processed by the threshold-based detection algo-

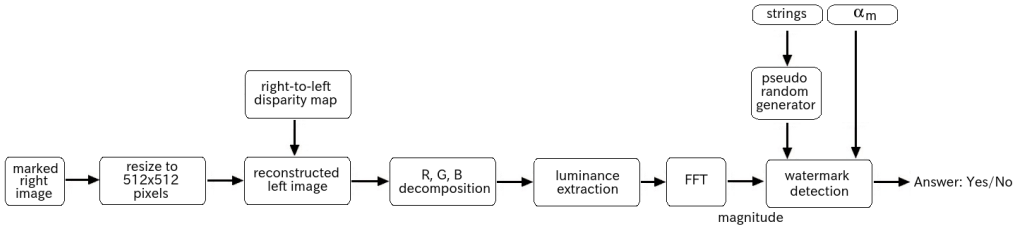


Figure 4.10: Watermark detection process for right image

rithm.

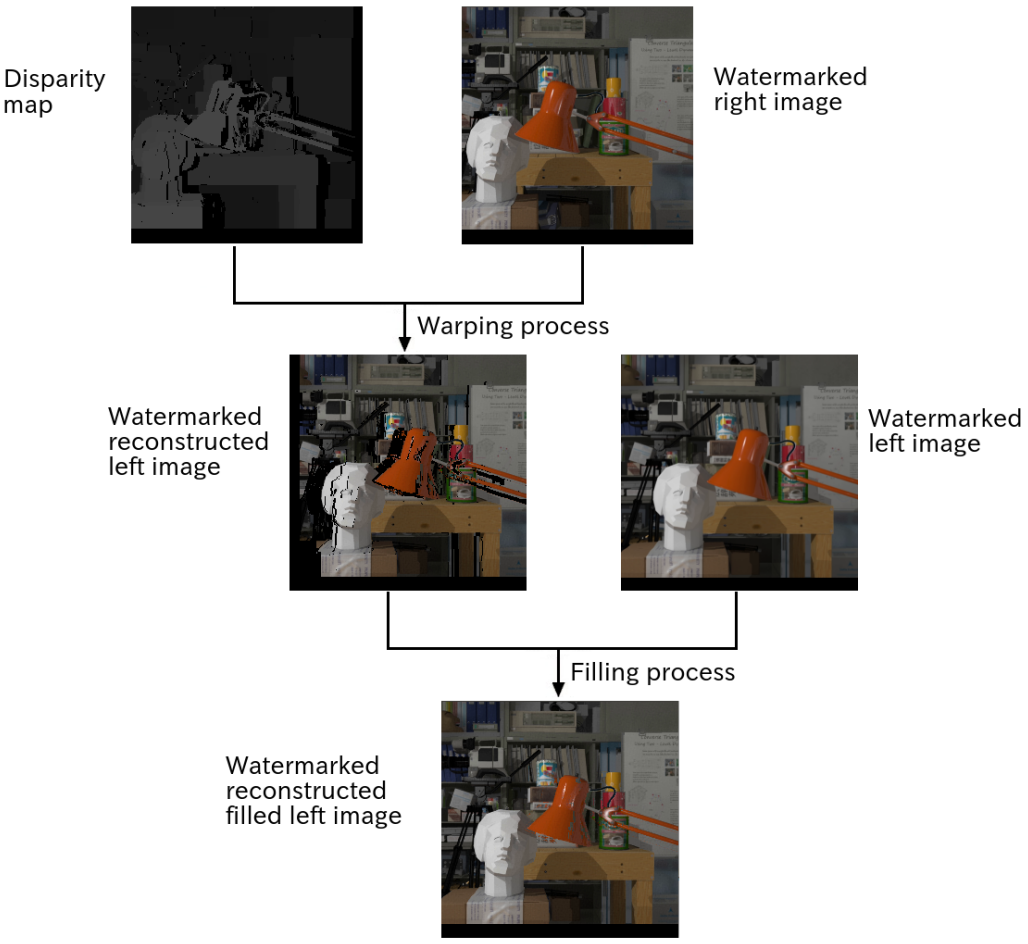


Figure 4.11: Detection workflow

Chapter 5

Experimental Results

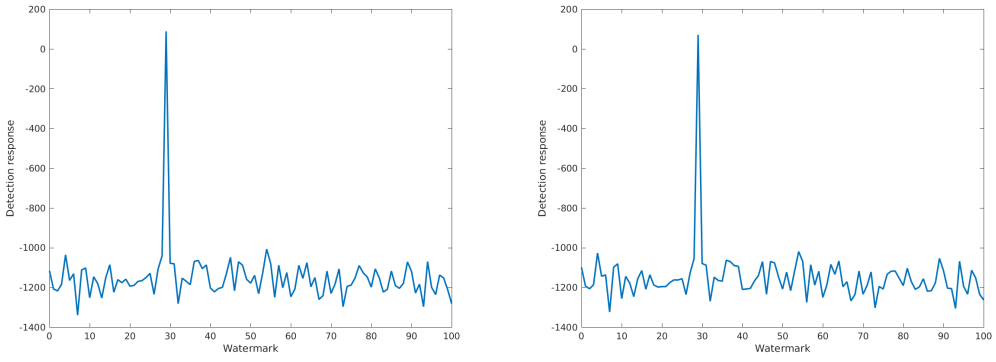
The proposed method has been tested to verify, besides the uniqueness of the watermark, its validity in terms of robustness and perceptual impact.
As said before, robustness is the ability of the watermark to cope with the degradation of the image due to compression, view synthesis etc.
Another important feature of a good watermarking method is perceptual transparency, such that human eye could not distinguish the dissimilarities between the watermarked image and the original one.
In this chapter will be presented the results carried out to test the algorithm performances.

The marking process has been applied to a 1 minute stereo-video sequence created starting from the left and right view of the New Tsukuba dataset [**?**], with GOP of 60 frames and 30 fps.
Its been chosen to mark every 60 frames, i.e. only the I frame of each GOP.
The frames of the reference video has been marked with different power and new marked videos has been created with different levels of compression.

The compressed videos are made with the `ffmpeg` library [**?**], changing the Constant Rate Factor (CRF), the default quality setting for the x264 encoder. The value can be set in a range between 0 and 51, where lower values would result in better quality (at the expense of higher file sizes).

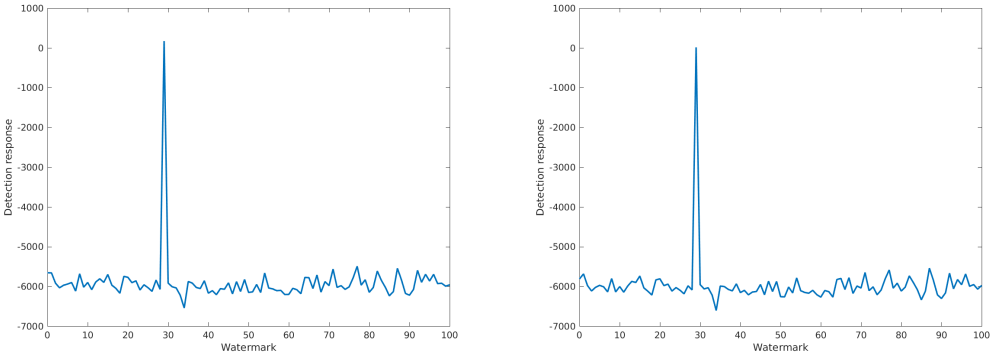
5.1 Uniqueness of the watermark

The first experiment we present aims to demonstrate the uniqueness of the watermark. We can say a mark is unique if the detector will present a significantly higher score only in correspondence of the reference watermark. Figures 5.1a-5.2b show the response of the watermark detector to 100 randomly generated watermarks of which only one matches the reference watermark.



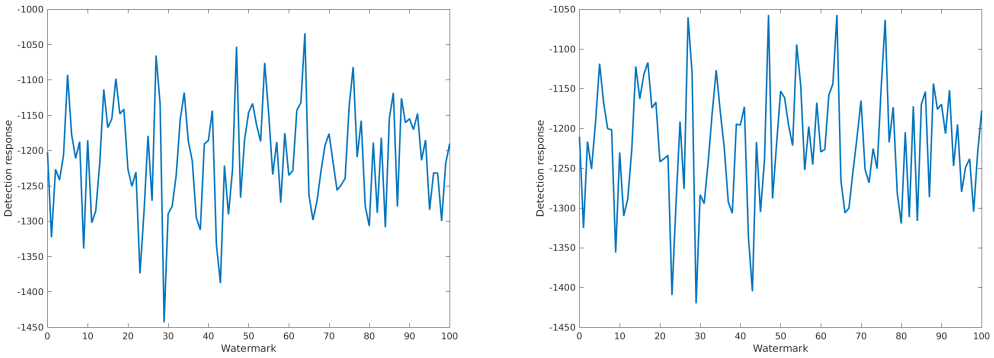
(a) detection on the left view (b) detection on the right view

Figure 5.1: Detector response on the left and right views marked with power equal to 0.3



(a) detection on the left view (b) detection on the right view

Figure 5.2: Detector response on the left and right views marked with power equal to 0.6



(a) detection on the left view (b) detection on the right view

Figure 5.3: Detector response on the left and right views where the images hasn't been marked

It can be noted that the response due to the correct watermark is very much stronger with respect to the response to incorrect watermarks, suggesting that the algotithm has very low false positive (and false negative) response rate. This holds either for the left detection (figure 5.1a and 5.2a) and for the right detection(figure 5.1b and 5.2b).
Figures 5.3a and 5.3b prove that when the image under consideration doesn't contain the reference watermark the score array doesn't present any spikes.

5.2 Robustness against compression attack

In video analysis, compression is useful because it helps reduce resource usage, such as data storage space or transmission capacity.
This process is considered an attack since it brings to a degradation of the image due to the compression ratio, thus a degradation of the watermark.
This problem can be addressed improving the strenght of the embedded watermark, so it can resist the image degradation, but it is necessary to mantain an acceptable trade-off between robustness and the perceptual impact of the watermark.

Figures 5.4-5.8 show the degradation of the image due to compression and watermark power.

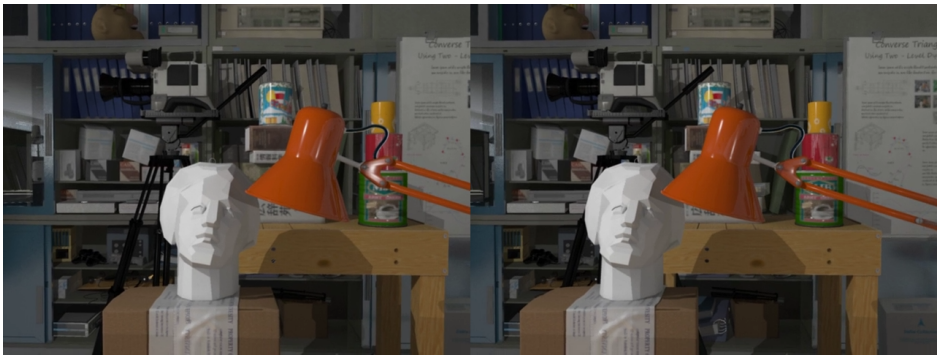


Figure 5.4: Stereo image from video marked with power 0.3 and compressed with crf equal to 1

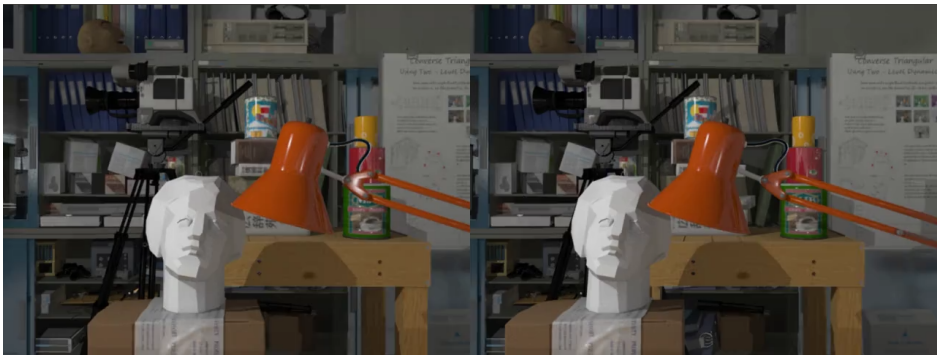


Figure 5.5: Stereo image from video marked with power 0.3 and compressed with crf equal to 25

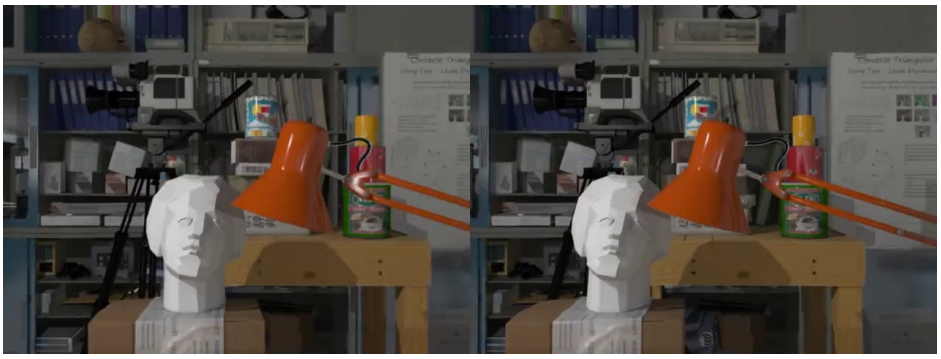


Figure 5.6: stereo image from video marked with power 0.3 and compressed with crf equal to 30

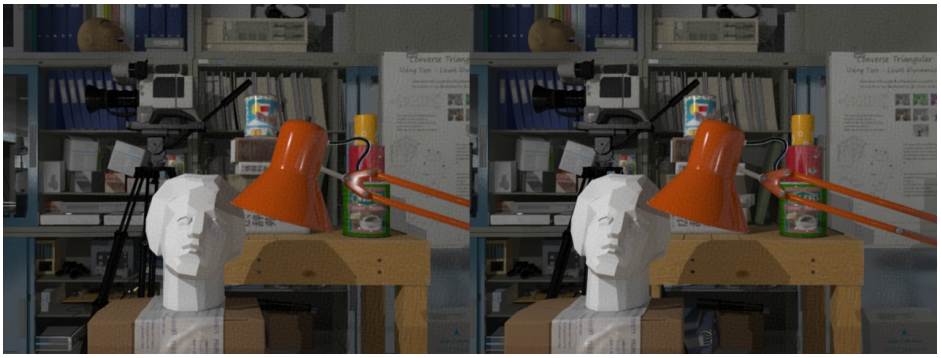


Figure 5.7: Stereo image from video marked with power 0.6 and compressed with crf equal to 1

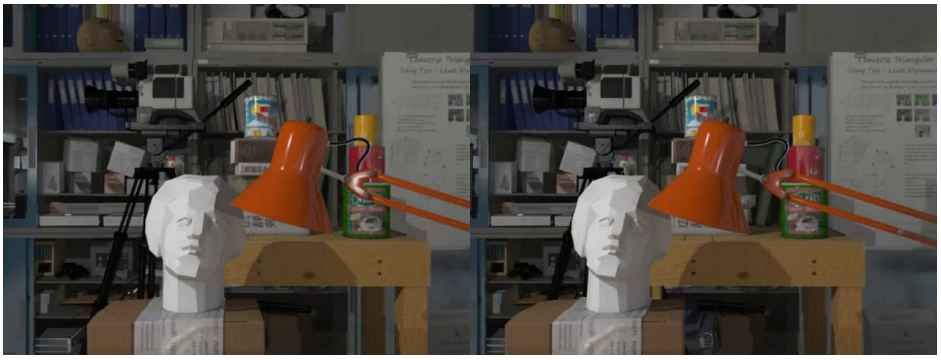


Figure 5.8: Stereo image from video marked with power 0.6 and compressed with crf equal to 25

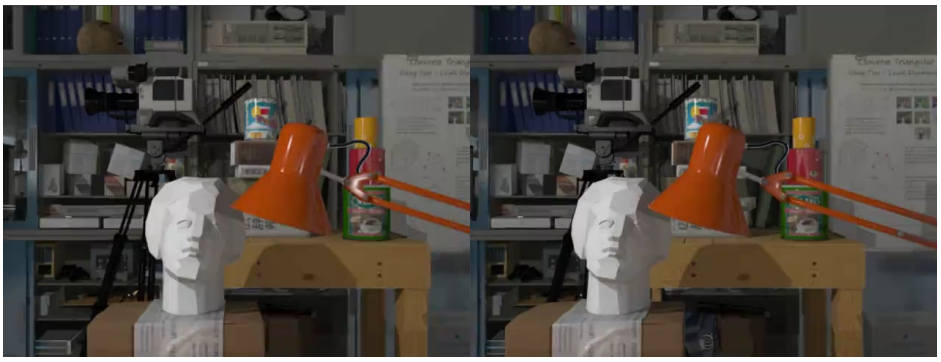


Figure 5.9: stereo image from video marked with power 0.6 and compressed with crf equal to 30

In figures 5.4-5.6 the images has been marked with power equal to 0.3, we can see that this way the watermark does not affect the image and how the frames are degradated from the compression.

Yet in figures 5.7-5.9 it can be noted that when the power is equal to 0.6 the mark is perceptible and it become less visible the more the image is compressed.

5.2.1 Spatial watermarking robustness

In spatial domain watermarking systems, the watermark is embedded directly in the spatial domain (pixel domain).
Many of the spatial watermarking techniques provide simple and effective schemes for embedding an invisible watermark into an image, but are less robust to common attacks such as lossy compression.

The evaluation of this detection system has been studied through the ROC curve, which show the performance of a binary classifier system as its discrimination threshold is varied. The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The best possible prediction method would yield a point in the upper left corner or coordinate (0,1) of the ROC space, representing 100% sensitivity (no false negatives) and 100% specificity (no false positives). The (0,1) point is also called a perfect classification. A completely random guess would give a point along a diagonal line from the left bottom to the top right corners.

The results are in figures 5.10-5.17.

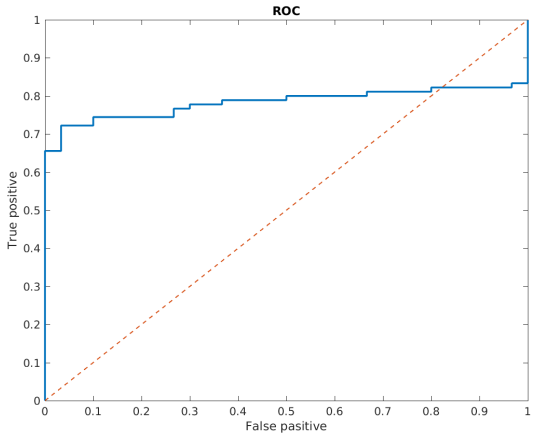


Figure 5.10: ROC curve of a spatial marked image with power equal to 1 and not compressed

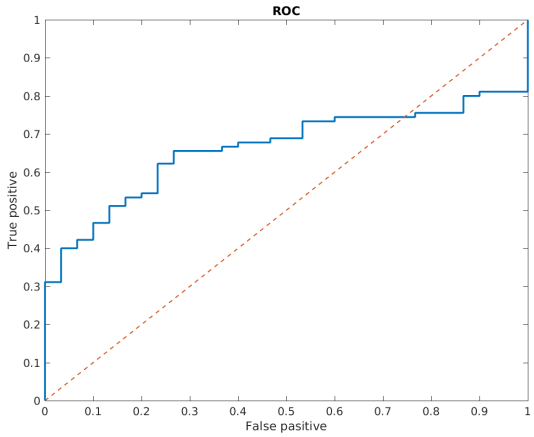


Figure 5.11: ROC curve of a spatial marked image with power equal to 1 and compressed with crf 15

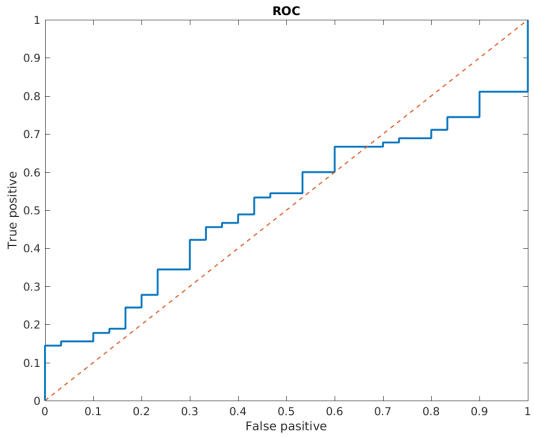


Figure 5.12: ROC curve of a spatial marked image with power equal to 1 and compressed with crf 25

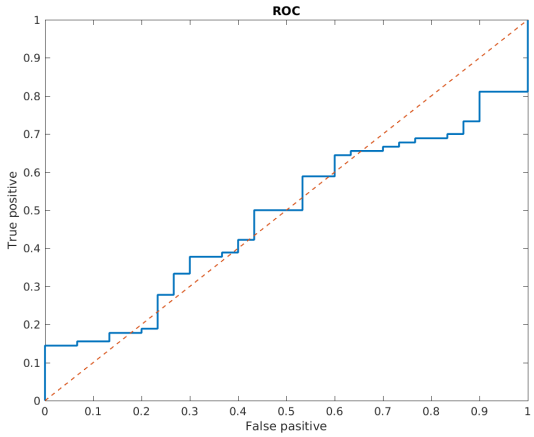


Figure 5.13: ROC curve of a spatial marked image with power equal to 1 and compressed with crf 30

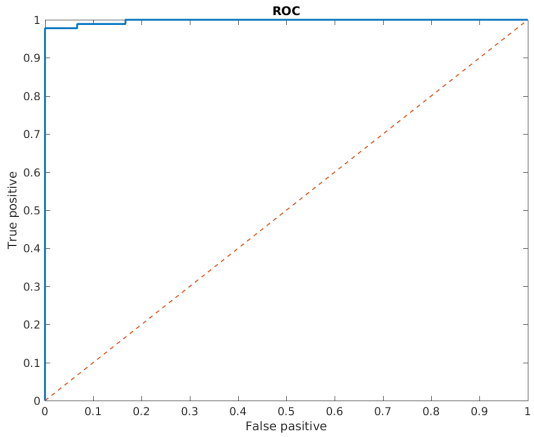


Figure 5.14: ROC curve of a spatial marked image with power equal to 3 and not compressed

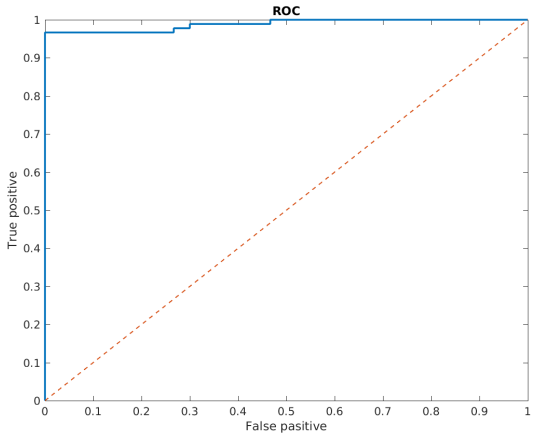


Figure 5.15: ROC curve of a spatial marked image with power equal to 3 and compressed with crf 15

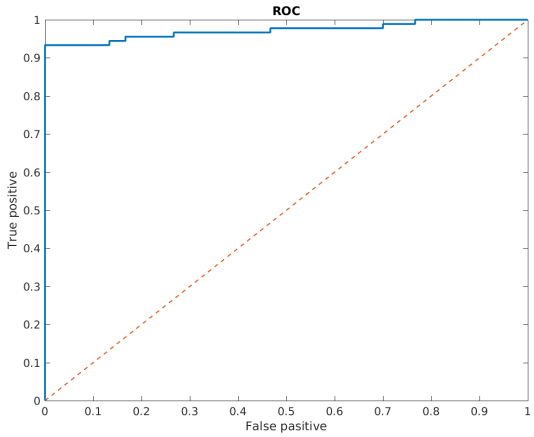


Figure 5.16: ROC curve of a spatial marked image with power equal to 3 and compressed with crf 25

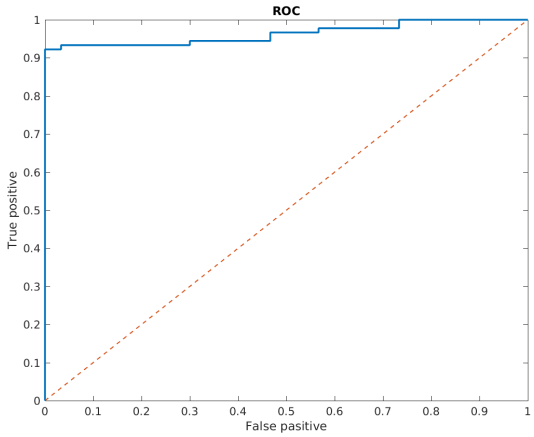


Figure 5.17: ROC curve of a spatial marked image with power equal to 3 and compressed with crf 30

The ROC functions above reveal that the more the image is compressed the more the classification is a random guess, but when the mark is added with power equal to 3 the compression doesn't affect the detection process.

5.2.2 DFT watermaking robustess

In transform domain watermarking systems, watermark insertion is done by transforming the image into the frequency domain using a discrete Fourier transform (DFT), full-image DCT, block-wise DCT, wavelet, Hadamard, Fourier-Mellin, or other transforms.
It is often claimed that embedding in the transform domain is advantageous in terms of visibility and security.

Two studies are presented in this section: the first one concernes the power of the watermark needed in order to achieve robustness against different levels of compression; the latter focus on youtube, and tries to find the right power to achive robustness in a downloaded video.

Each test has been made with both the ground truth and graph-cuts disparities.

Tables 5.1-5.2 shows how the algorithm manage to find the watermark in a compressed video, in particular it is shown if the mark is detected in the left/right view or both images. The first table shows the results when the algorithm is used with the ground truth disparity, the second when using graph cuts.

power	compression level	both	left	right
0.3	1	30	0	0
0.3	15	30	0	0
0.3	25	10	5	0
0.3	30	1	1	0
0.6	1	30	0	0
0.6	15	30	0	0
0.6	25	28	0	0
0.6	30	16	2	0

Table 5.1: detection table when ground truth disparity is used

power	compression level	both	left	right
0.3	1	30	0	0
0.3	15	29	1	0
0.3	25	11	1	0
0.3	30	2	1	0
0.5	1	30	0	0
0.5	15	30	0	0
0.5	25	24	2	0
0.5	30	9	2	0
0.6	1	30	0	0
0.6	15	30	0	0
0.6	25	26	1	1
0.6	30	15	4	0

Table 5.2: detection table when graph cuts disparity is used

One can notice that, at a global level, detection statistics gradually degrade with the compression ratio. The embedded watermark becomes hardly detectable at the crudest compression levels even with if embedded with a strong power.

From the results emerges that when the watermark power is grater than or equal to 0.5 the detection supports a compression level of 25 with good statistics.

Figures 5.18-5.21 show how the uploading and the subsequential down-

load of a non compressed video on youtube degradates the image.

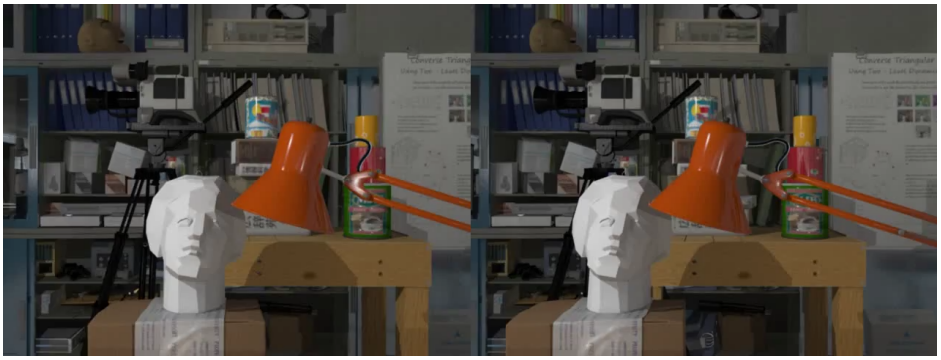


Figure 5.18: Stereo image from video uploaded with power equal to 0.3

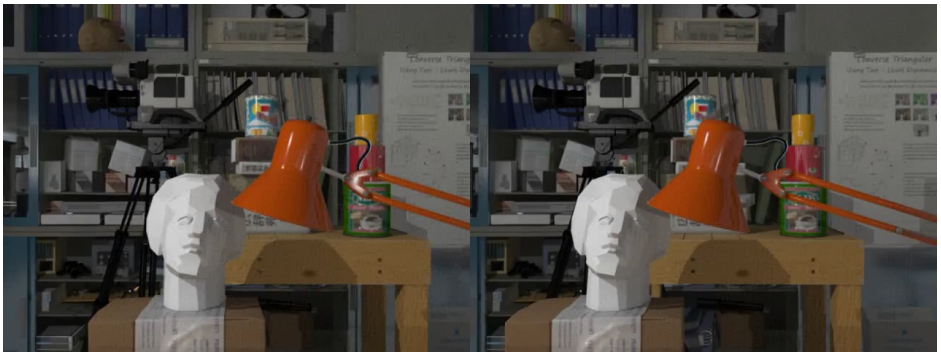


Figure 5.19: Stereo image from video uploaded with power equal to 0.6

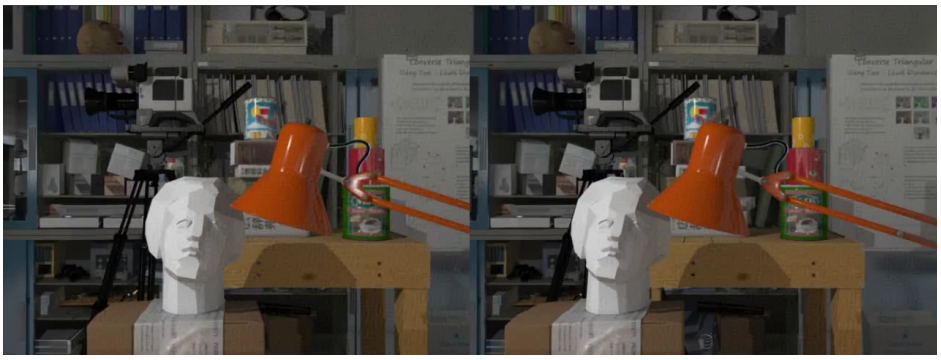


Figure 5.20: Stereo image from video uploaded with power equal to 0.7

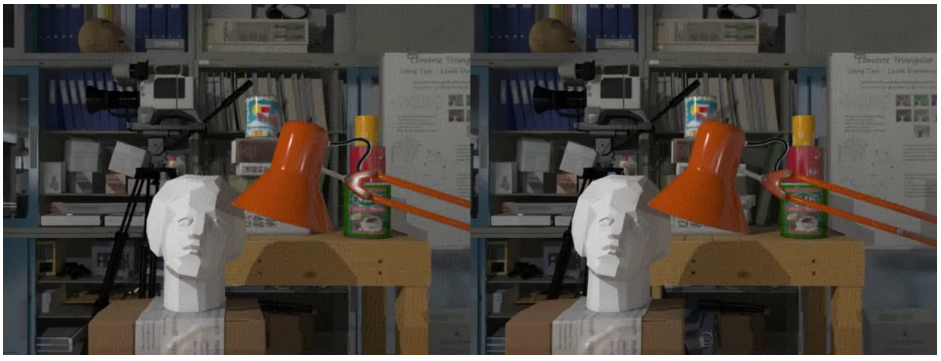


Figure 5.21: Stereo image from video uploaded with power equal to 0.8

From Figures 5.18-5.21 it can be noticed that as a consequence of the image degradation the watermark become less perceptible even when inserted with a high power.

Tables 5.3-5.4 show how a video uploaded on youtube and subsequentially downloaded can preserve the watermark, respectively when the watermark is inserted with the ground truth disparity and with graph cuts.

power	both	left	right
0.3	1	0	0
0.6	9	1	0
0.7	12	2	0
0.8	16	0	1

Table 5.3: Detection statistic for a downloaded video marked with ground truth disparity

power	both	left	right
0.3	1	0	0
0.5	6	1	0
0.6	11	0	0

Table 5.4: Detection statistic for a downloaded video marked with graph cuts disparity

To validate this results the average PSNR value has been computed between original and compressed stereoscopic videos at diffent compression levels.

PSNR is the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation; it is usually expressed in terms of the logarithmic decibel scale (dB) and typical

values for the PSNR in video compression and watermarking are between 30 and 50 dB. The results of this study are shown in Table 5.5: PSNR value decreases with the increment of the compression level. This implicates a decrement of the true positive rate after compression due to the image degradation as shown in Tables 5.1-5.2 and 5.3-5.4.

Compression Level(CRF)	PSNR(dB)
15	46.0194
25	40.4861
YT	38.2039
30	37.5587

Table 5.5: Average PSNR values between original video and compressed videos at different compression levels. The acronym YT stands for YouTube compression level, whose value is between 25 and 30 as the PSNR results show.

5.3 Robustness to View Synthesis

In a second batch of experiments, we analyzed the impact of virtual view synthesis on the detection performances of our watermarking system.

View synthesis can be viewed as the interpolation of a virtual view from two reference views. The reference views are essentially warped to the virtual viewpoint based on depth information and the intrinsic and extrinsic camera parameters and merged together.

The view synthesis process using depth itself may destroy the watermark, as a result, to achieve robustness against this process, it is important to design the watermarks inserted in the left and right views so that they nicely overlap after the warping operation and thereby complement each other rather than

cancel one another. The same 3D point should always carry the same watermark sample wherever it is projected in a view; hence we used a disparity-coherent technique.

To conduct these experiments, we generated a number of intermediate synthetic views (figures 5.22-5.24), equally spaced apart between the left (reference) view and the right one, using the code in [**?**].

Figure 5.22: Synthetized view at distance 1/4 of the baseline from the left image

Figure 5.23: Synthetized view at distance 1/2 of the baseline from the left image

The Table 5.6 contains the results for the frequency marking: the first column is the distance between the left view and the synthetized one, in

Figure 5.24: Synthetized view at distance 3/4 of the baseline from the left image

terms of fraction of the baseline, then its show in how many synthetized images the mark is detected.

position	both	left	right
1/2	30	0	0
1/4	30	0	0
3/4	29	1	0

Table 5.6: Detection in the syntetized views

The same study is proposed for the spatial marking: from a video marked with additive gaussian noise have been generated three synthetized views for each pair of marked frames, respectively one in the middle and the other two at 1/4 and 3/4 of distance from the left.
The ROC curves in figures 5.25-5.30 show the results for the different intermediate synthetic views.

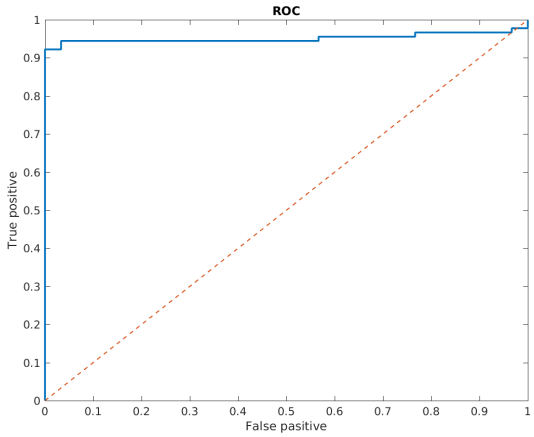


Figure 5.25: ROC curve of a synthetic view created at distance equal to baseline/4 marked with power equal to 1

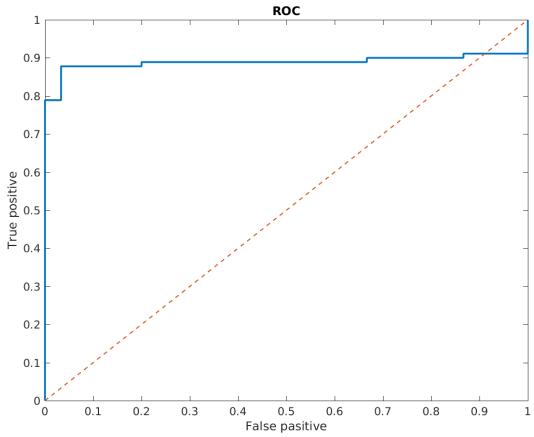


Figure 5.26: ROC curve of a synthetic view created at distance equal to baseline/2 marked with power equal to 1

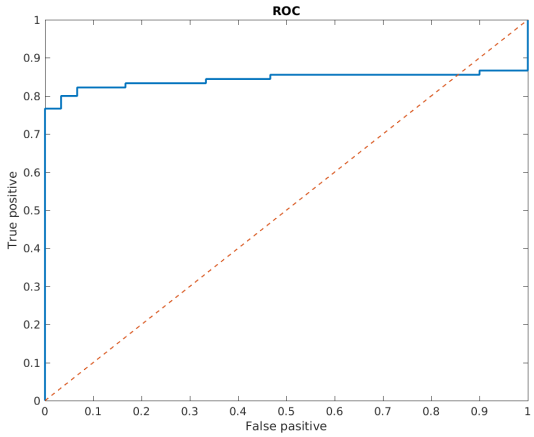


Figure 5.27: ROC curve of a synthetic view created at distance equal to baseline*3/4 marked with power equal to 1

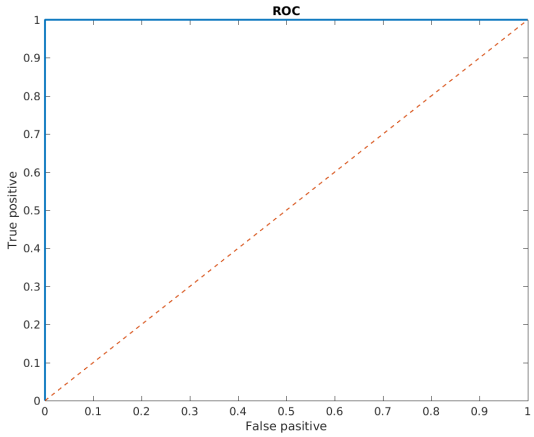


Figure 5.28: ROC curve of a synthetic view created at distance equal to baseline/4 marked with power equal to 3

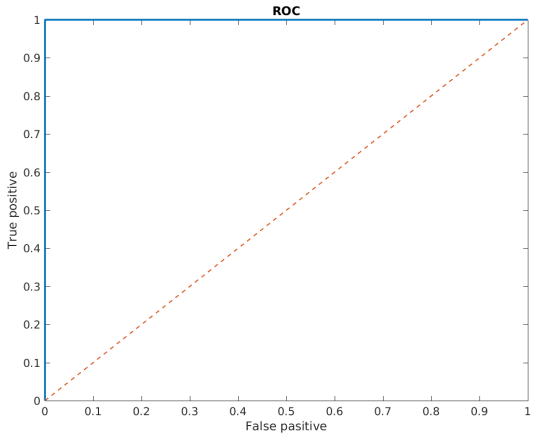


Figure 5.29: ROC curve of a synthetic view created at distance equal to baseline/2 marked with power equal to 3

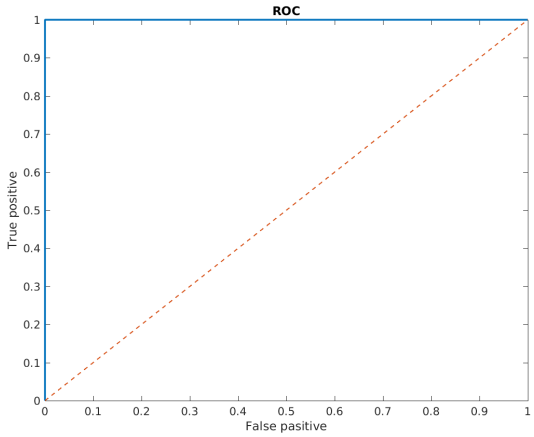


Figure 5.30: ROC curve of a synthetic view created at distance equal to baseline*3/4 marked with power equal to 3

It can be noted that the detection statistics are very high in the synthetized views, either for the spatial and frequency technique; we can therefore expect the synthetized views to behave like the other against compression.

Often with stereo watermarking, view synthesis can be a problem since it introduces non-rigid local geometric distortion that are not properly tackled by state-of-the art resynchronization mechanisms. Local geometric deformations destroys the synchronization necessary for the detection process to be succesful.

With the proposed method the detection of the watermark in the right view works by warping it according to the disparity; this way the resynchronization is an internal step of the detection process and it doesn't need side information to resynchronize the watermark, since the disparity can be estimated anytime from the received views.
Therefore we can say that this strategy manages to achieve complete robustness to view synthesis.

5.4 Perceptual impact

As said in Chapter 2 the perceptual impact, i.e. the imperceptivity of the watermark to the human eye, has been measure with the metrics proposed by Chaminda et al.
Based on the fact that the edges/contours of the depth map can be considered for measuring structural degradation, it is proposed a metric which is a modified version of the SSIM metric. This measure is supposed to be computed on the degradated view and the corresponding disparity map, in our case we measured either the degradataion on the left and right view.

Figure 5.31 gives graphical illustration of contour extraction in order to compute the RR metrics on the left view, in this case the image has been marked with power equal to 0.3 and it can be noted that this kind of watermarking doesn't affect the contours of the scene.

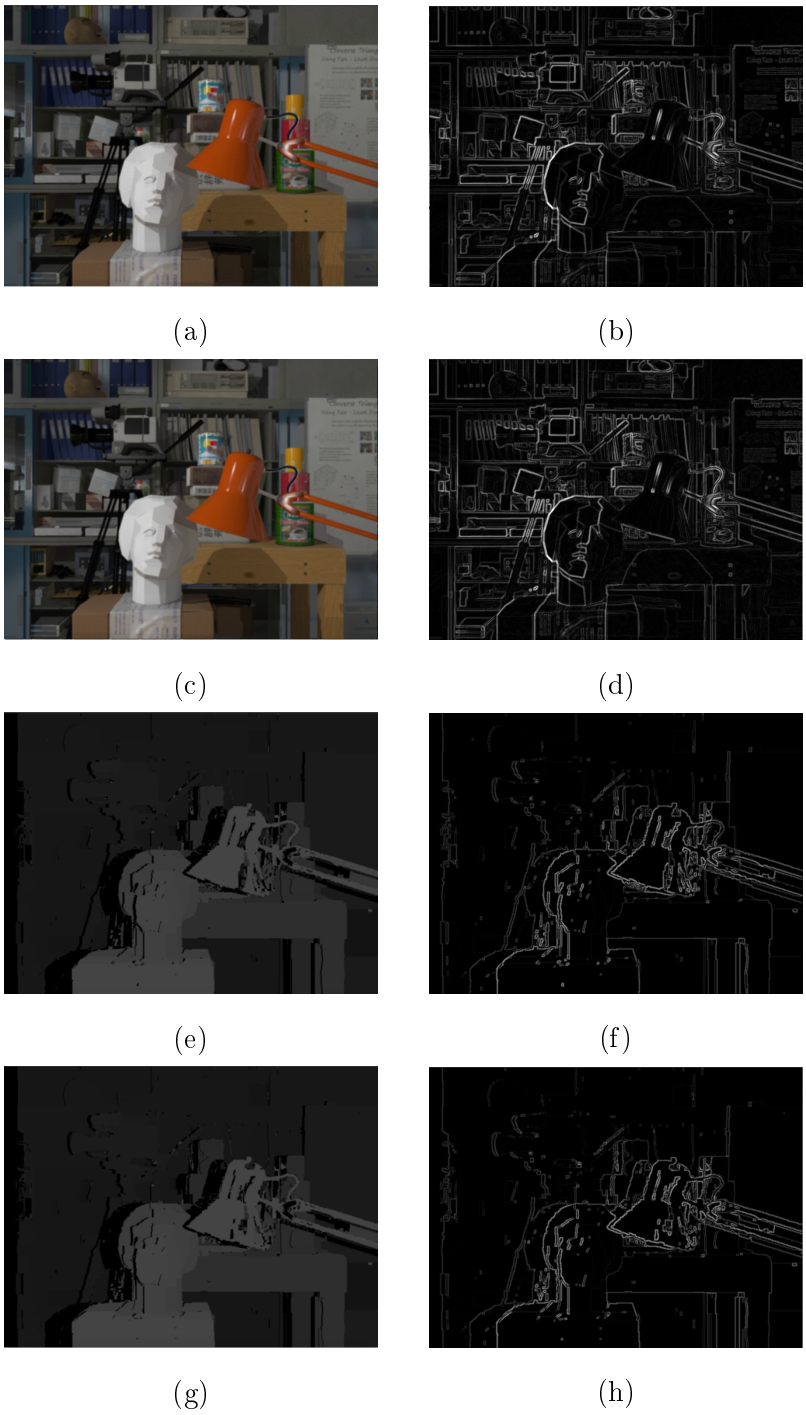
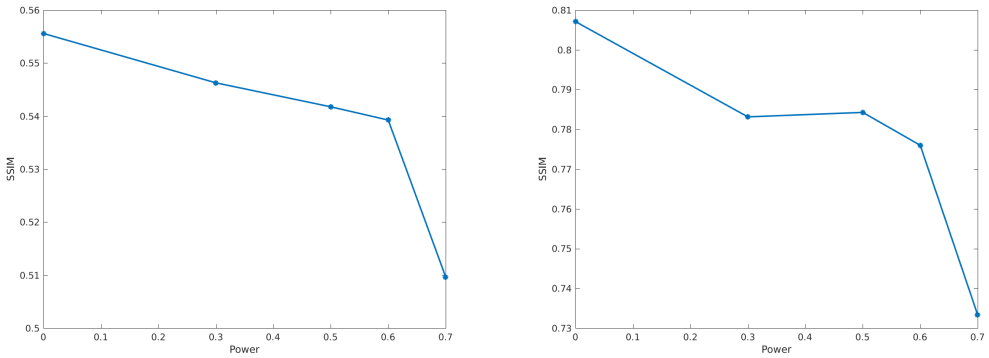


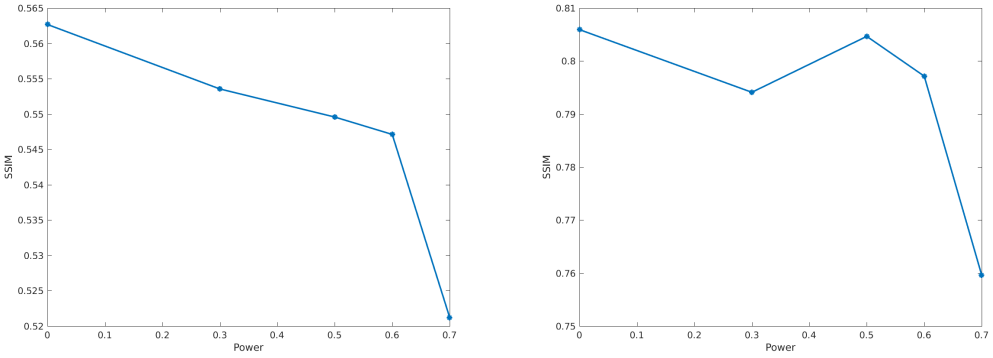
Figure 5.31: (a) Reference left image. (b) Extracted edge information from reference left image. (c) Watermarked left image. (d) Extracted edge information from the watermarked left image. (e) Left disparity map obtain from the non watermarked stereo pair. (f) Extracted edge information from the left disparity map obtain from the non watermarked stereo pair. (g) Left disparity map obtain from the watermarked stereo pair.(h) Extracted edge information from the left disparity map obtain from the watermarked stereo pair.

For different power of watermarking both the MQ_{depth} and MQ_{color} metrics are calculated, and in figures 5.32-5.33 its shown the value of the quality measure with respect to the power of the embedded watermark.



(a) Color quality metrics on the left view (b) Color quality metrics on the right view

Figure 5.32: Color Quality metrics



(a) Depth quality metrics on the left view (b) Depth quality metrics on the right view

Figure 5.33: Depth quality metrics

In figures 5.32-5.33 we can see how the quality metrics decrease when increasing the power of the watermark: it can be noted that the more the power of the watermark is high the more the quality of the perception is low, however with power lower the 0.6 the degradation mantain an acceptable level with respect to the value generated from the non marked image (power equal to 0).

Another study has been conducted to evaluate the visual impact of the watermark: the average PSNR value has been computed between the I frames of the original stereoscopic video and the watermarked stereoscopic videos with different values of the power(that is 30 pairs of stereoscopic frames)
. As said before typical values for the PSNR in video watermarking are between 30 and 50 dB. The results of this study are shown in Table 5.7: with a power value of 0.3 PSNR reaches a maximum value of 44.438, which indicates a good quality of the watermarked stereo pair.
As expected the PSNR value decrease with the increment of the watermaking power, indicating a degratation of the watermarked videos.
. This analysis is in line with the one conducted with the quality metrics in [?].

Power	PSNR(dB)
0.3	46.0071
0.5	45.9505
0.6	45.9291

Table 5.7

5.5 Remarks

In this chapter the experiments conducted to test the proposed watermarking technique have been presented.

First we proved the uniqueness of the watermark needed in order to have a reliable marking process.

The created stereo video sequence was than marked and compressed under different compression rates, to test the visibility of the watermark as well as its robustness against lossy compression.

Regarding the spatial technique results show that when the watermark is added with power equal to 1, it is preserved with acceptable statistic at a compression rate of 15; on the other hand when inserted with higher power (3 in this analysis), the detection statistics are optimal even for the crudest compression.

The frequency watermarking proved to be robust up to a compression rate of 25, and that the degradation introduced by web uploading tends to erase the mark, so in order to cope with this kind of attack a higher embedding power is needed.

Another important feature to test was the robustness against view synthesis, which was confirm by the results, for both the spatial and frequency techniques.

The quality of the watermarked video has been studied with RR quality metrics, and it emerges that the stereo frame degradation mantains a good value when marking with power lower than 0.6.

Finally the average PSNR value has been computed to analyse compression and visual impact of the watermark; the study supports the previous results.

Chapter 6

Conclusions

In this thesis a blind disparity-coherent watermarking algorithm has been implemented. While prior works only inserted the mark in the spatial domain, in this case both the frequency and spatial domain are considered.
The marking process can be summerized in two steps: (i) a pseudo-random sequence of real numbers is embedded in a selected set of DFT coefficients of the left image, (ii) the reference watermark is spatially inserted in a disparity-coherent way in the right view.
A new detection process is then proposed, which is also based on the disparity map: (i) the detection on the left view is performed according to [**?**], where the decision criterion is derived based on statistical decision theory, (ii) the detection on the right view is performed by first warping it according to the right-to-left disparity, this way the watermark is resynchronized to its initial shape.

The method has been tested against compression attack and web uploading. It emerged that the watermark can resist until a compression with crf equal to 25 with good detection statistics, besides the fact that the mark become less visible the more the compression rate increases, so it can be

embedded with a higher power.
The web uploading although doesn't prevent the mark with good statistics, this is why an important future development would be to hide the mark with a mask, so it can be inserted with higher power.

The method has also proved to be robust against view synthesis, thanks to the fact that the detection process resynchronize the watermark on the right view before performing the detection.

To evaluate the quality of the watermarked video sequence PSNR value and new measures based on SSIM value have been used. The experimental results shows that the video quality degradates inversely with the power of the watermark but it mantain a good measure when marking with power lower than 0.6; PSNR reaches a maximum of 44.438 with a power value of 0.3.

Future works could provide a visual mask to improve the quality of the waterkmarked video sequence, and add a synchronization pattern to make the watermak robust against geometrical attacks.

Further investigations are also needed to better comprehend the sensitivity of the human eye to noise addition in the left and right view.

Appendix A

Libraries and codes

The watermarking algorithm has been implemented in C++ programming language. The following libraries and codes has been used:

- `rectify-quasi-euclidean_20140626` []: to compute disparity range;
- `kz2_r1.0` [?], to compute disparity map;
- `ffmpeg-2.7.2` [?], to compress video sequence;
- `libconfig` [?], to set and save the watermark parameters.

Matlab code `viewSynthCode` [?] has been used to compute intermediate frames for view synthesis experiments.

Bibliography

[1] Sun Microsystems. Javasoft ships java 1.0, 1 1996.