H.264 Video Watermarking: Applications, Principles, Deadlocks, and Future
Marc Chaumont

To cite this version:

HAL Id: lirmm-00577950
https://hal-lirmm.ccsd.cnrs.fr/lirmm-00577950
Submitted on 17 Mar 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
H.264 video watermarking: applications, principles, deadlocks, and future

Marc Chaumont

July 17, 2010
Preamble

Outline

1. Preamble

2. H.264

3. Watermarking
   - Definitions
   - Video watermarking
   - Security of video watermarking
   - A practical example: the traitor tracing (active fingerprinting)

4. Conclusion & Perspectives
Slides may be downloaded at [http://www.lirmm.fr/~chaumont/Publications.html](http://www.lirmm.fr/~chaumont/Publications.html)
e-mail: marc.chaumont@lirmm.fr
Where video compression is hidden in every days life?

A word of video compression

- Camera (Video surveillance, Smart Phone, ...),
- Streaming (YouTube, Television, ...),
- Storing (DVD, Blue-Ray, Hard-Disk, ...),
- Editing (Cinema, advertisement, entertainment).

→ Lots of people use videos.
There is a need for good compression algorithms

There is more and more video contents but:

- network bandwitch is limited,
- storing devices memories are limited.

Example:

**SDTV (images 720x576, 25 fps, 90 min. of movie):**

- Rate without compression: 237 Mbits/s,
- ADSL in France $\approx 30€/month$ for 20 Mbits/s.
- Storing capacity without compression: 1,22 Tera-bits,
- Storing capacity for a DVD: 4,7 GB.
Standardization - codecs evolution

Figure: 25 years of video compression standards
And where is the money?

People making money with video contents:

- Producers,
- Distributors (Hollywood...),
- Cinema operators,
- Technology providers (Internet providers, Reading and Recording devices constructors).

Those lobbys are dynamic in standardization committees and/or watermarking.
A part of this money is used for protection

The problem for right owners is the pirates...

Scientists should find solutions in order to dissuade users from pirating
Many solutions for security

Possible solutions:

- **cryptography**
  - flaws due to reverse engineering.

- **securize all the channel** from reader to displayer
  - Example: Blue-Ray reader + HD TV + HDMI wire
  - Blue-Ray DVDs have already been pirated.

- **spy the network and collect pirate IPs**
  - it may dissuade casual user.

- **propose cheap movies renting**
  - it may dissuade casual user.

- ...

- **watermarking**
  - ...
What about watermarking?

Applications using watermarking:

<table>
<thead>
<tr>
<th>Related to security</th>
<th>Related to media enhancement</th>
</tr>
</thead>
<tbody>
<tr>
<td>copyright identification</td>
<td>broadcast monitoring</td>
</tr>
<tr>
<td>traitor tracing (active fingerprinting)</td>
<td>device control</td>
</tr>
<tr>
<td>authentication</td>
<td>enrichment (functionalities and/or meta-datas)</td>
</tr>
<tr>
<td>copy control</td>
<td>with forward compatibility</td>
</tr>
<tr>
<td></td>
<td>improve compression performances</td>
</tr>
<tr>
<td></td>
<td>improve error recovery &amp; correction</td>
</tr>
</tbody>
</table>

In most of these applications, the watermarking should be robust.
Outline

1 Preamble

2 H.264

3 Watermarking
   • Definitions
   • Video watermarking
   • Security of video watermarking
   • A practical example: the traitor tracing (active fingerprinting)

4 Conclusion & Perspectives
What is H.264/AVC?

H.264 or MPEG-4 Part 10:

- **State-of-the-art** video coding standard,
- First version approved in 2003,
- Normalized by ITU-T and ISO/IEC organizations,
- **Up to 50% in bit rate savings** compared to MPEG-2 and MPEG4 Part 2 simple profile.


Visual example...

H.264 100Kbs

MPEG2 100Kbs
General coding scheme

- Macroblock 16x16

INTRA or INTER prediction

+ - residue

Integer Transformation

Quantization

Entropy coding

0010101001 bitstream

control data

motion data
General coding scheme

- **Macroblock**: 16x16
- **INTRA or INTER prediction**
- **Integer Transformation**
- **Quantization**
- **Entropy coding**

Data flow:
- Residue
- Control data
- Motion data

Result: 0010101001 bitstream
Intra prediction modes

0 (vertical) 1 (horizontal) 2 (DC) 3 (diagonal down-left) 4 (diagonal down-right)

5 (vertical-right) 6 (horizontal-down) 7 (vertical-left) 8 (horizontal-up)
General coding scheme

Macroblock (16x16)

INTRA or INTER prediction

+ - residue

Integer Transformation

Quantization

Entropy coding

0010101001 bitstream

control data

motion data
Inter prediction - Motion Estimation

**Figure:** Motion estimation in temporal direction.
Inter prediction - Motion Vectors

Figure: Motion vectors.
General coding scheme

Macroblock 16x16

INTRA or INTER prediction

+ - residue

Integer Transformation

Quantization

Entropy coding

0010101001 bitstream

control data

motion data
General coding scheme

H.264 video watermarking - Marc Chaumont - IPTA’2010

16x16 Macroblock

INTRA or INTER prediction

+ - residue

Integer Transformation

Quantization

Entropy coding

0010101001 bitstream

control data

motion data
General coding scheme

- Macroblock 16x16
- INTRA or INTER prediction
- Integer Transformation
- Quantization
- Entropy coding
- 0010101001 bitstream
- Residue
- Control data
- Motion data
Outline

1. Preamble

2. H.264

3. Watermarking
   - Definitions
   - Video watermarking
   - Security of video watermarking
   - A practical example: the traitor tracing (active fingerprinting)

4. Conclusion & Perspectives
Outline

1. Preamble
2. H.264
3. Watermarking
   - Definitions
     - Video watermarking
     - Security of video watermarking
     - A practical example: the traitor tracing (active fingerprinting)
4. Conclusion & Perspectives
Recall: applications

Applications using watermarking:

<table>
<thead>
<tr>
<th>Related to security</th>
<th>Related to media enhancement</th>
</tr>
</thead>
<tbody>
<tr>
<td>copyright identification</td>
<td>broadcast monitoring</td>
</tr>
<tr>
<td>traitor tracing (active fingerprinting)</td>
<td>device control</td>
</tr>
<tr>
<td>authentication</td>
<td>enrichment (functionalities and/or meta-datas)</td>
</tr>
<tr>
<td>copy control</td>
<td>with forward compatibility</td>
</tr>
<tr>
<td></td>
<td>improve compression performances</td>
</tr>
<tr>
<td></td>
<td>improve error recovery &amp; correction</td>
</tr>
</tbody>
</table>

In most of these applications, the watermarking should be robust.
What is robust watermarking?

The robust watermarking is the art of modifying a media (image, sound, video, ...) such that:

- it contains a **message** most of the time in relation with the media,
- degradation is most of the time **imperceptible**,
- the hidden **message is not lost** when media degradation occurs (attacks).
Robustness illustration

original

watermarked

Robustness illustration(1): detection = Ok

Robustness illustration(2): detection = Ok

Robustness illustration(3): detection = Ok

watermarked

luminosity upscaling

Robustness illustration(4): detection = Ok

watermarked

luminosity downscaling

Robustness illustration(5): detection = Ok

watermarked

sharp amplification

Desynchronization attack detection = NOT Ok

watermarked

rotated, cropped, and resized

## Few non-malicious attacks for a video

### Non-malicious attacks:

| Photometric | Noise addition, DA/AD conversion  
|            | Gamma correction  
|            | Transcoding and video format conversion  
|            | Intra and inter-frames filtering  
|            | Chrominance resampling (4:4:4, 4:2:2, 4:2:0)  |
| Spatial Desynchronization | Changes display formats (4/3, 16/9, 2.11/1)  
|            | Changes resolution (NTSC, PAL, SECAM)  
|            | Positional jitter  
|            | Hand-held camera recording (curved-bilinear transform)  |
| Temporal Desynchronization | Changes of frame rate  
|            | Frame dropping / insertion  
|            | Frame decimation / duplication  |
| Video editing | Cut-and-splice and cut-insert-splice  
|            | Fade-and-dissolve and wipe-and-matte  
|            | Graphic overlay (subtitles, logo)  |

---

Robust watermarking families

The three major families (multi-bits):
Not informed : Spread Spectrum (DM-SS),
Informed : Quantized-based (QIM, SCS, P-QIM),
Informed : Trellis-based (DPTC).
Informed embedding scheme

![Diagram of informed embedding scheme]

**Figure:** Informed embedding.
Informed extracting scheme

Watermarked image \rightarrow \text{Noise} \rightarrow y \rightarrow z \rightarrow \text{Watermark extractor} \rightarrow \text{Watermark decoder} \rightarrow m_n \rightarrow \text{Output message}

Watermark key

\text{Figure: Blind extraction.}
Outline

1. Preamble
2. H.264
3. Watermarking
   - Definitions
   - Video watermarking
   - Security of video watermarking
   - A practical example: the traitor tracing (active fingerprinting)
4. Conclusion & Perspectives
Major approaches

Before compression
- images
  - SS [Cox et al., TIP’1997]
  - DPTC [Miller et al., TIP’2004]
  - P-QIM [Li and Cox, TIFS’2007]
  - ... 

Before quantization
- Luma modification:
  - [Golikeri et al., JEl’2008]
- Motion vectors modification:
  - [Zhang et al., SCGIP’2001]
- GOP structure modification:
  - [Linnartz and Talstra, ESRCS’1998]
  - ... 

Inside H.264 structure
- After quantization
  - During encoding process
    - [Mobasseri and Raikar, SPIE’2007]
    - [Zou and Bloom, SPIE’2009]
    - ...

During Entropy coding
- In an already H.264 encoded bitstream
  - [Shahid et al., EUSIPCO’2009]
  - [Noorkami and Mersereau, TIFS’2008]
  - ...

Temporal watermarking
- [Haitsma and Kalker,ICIP’2001]
- [Chen et al., IWDW’2009]

3D DFT
- [Deguillaume et al., SPIE’1999]

On-off keying (Extended BA)
- [Xie et al., MM&Sec2008]
- ...
Chosen schemes

Before compression
- Images
  - SS [Cox et al., TIP’1997]
  - DPTC [Miller et al., TIP’2004]
  - P-QIM [Li and Cox, TIFS’2007]
  - ...

- Sequence of images
  - Temporal watermarking
    [Haitsma and Kalker, ICIP’2001]
    [Chen et al., IWDW’2009]
  - 3D DFT
    [Deguillaume et al., SPIE’1999]
  - On-off keying (Extended BA)
    [Xie et al., MM&Sec2008]
  - ...

Inside H.264 structure

Before quantization
- Luma modification:
  [Golikeri et al., JEI’2008]
- Motion vectors modification:
  [Zhang et al., SCGIP’2001]
- GOP structure modification:
  [Linnartz and Talstra, ESRCS’1998]
  - ...

During encoding process
- [Mobasseri and Raikar, SPIE’2007]
  [Zou and Bloom, SPIE’2009]
  - ...

After quantization

During Entropy coding

In an already H.264 encoded bitstream

- [Hartung and Girod, Signal Processing 1998]
  [Gong and Lu, ISM’2008]
  - ...
Chosen schemes

Before compression

- images
  - SS [Cox et al., TIP’1997]
  - DPTC [Miller et al., TIP’2004]
  - P-QIM [Li and Cox, TIFS’2007]
  - ...

- sequence of images
  - Temporal watermarking [Haitsma and Kalker, ICIP’2001]
    [Chen et al., IWDW’2009]
  - 3D DFT [Deguillaume et al., SPIE’1999]
  - On-off keying (Extended BA) [Xie et al., MM&Sec2008]
  - ...

Inside H.264 structure

Before quantization

- Luma modification: [Golikeri et al., EJ’2008]
- Motion vectors modification: [Zhang et al., SCGIP’2001]
- GOP structure modification: [Linnartz and Talstra, ESRS’1998]
- ...

Inside H.264 structure

After quantization

- During encoding process
  - [Mobasser and Raikar, SPIE’2007]
  - [Zou and Bloom, SPIE’2009]
  - ...

During Entropy coding

- In an already H.264 encoded bitstream
  - [Shahid et al., EUSIPCO’2009]
  - [Noorkami and Mersereau, TIFS’2008]
  - ...

  [Hartung and Girod, Signal Processing 1998]
  [Gong and Lu, ISM’2008]
  ...

...
Before quantization

Macroblock 16x16

INTRA or INTER prediction

\[ +/ - \]

residue

Integer Transformation

Quantization

Entropy coding

\[ 0010101001 \]

bitstream

control data

motion data

[Goñikeri et al., Journal of Electronic Imaging 2007]

Golikeri et al. scheme

- Embed 1 bit in 1 macro-block,
- Use of a perceptual mask,
- Quantization-based watermarking (ST-SCS),
- Step size $\Delta$ and strength $\alpha$ parameters tune depending on the H.264 quantization.

<table>
<thead>
<tr>
<th>Property</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imperceptible</td>
<td>✓</td>
</tr>
<tr>
<td>Photometric robustness</td>
<td>✓ (should reduce payload)</td>
</tr>
<tr>
<td>Video bitrate weakly modified</td>
<td>✓ respect 'quiet well’ RD constraints</td>
</tr>
<tr>
<td>No Drift</td>
<td>✓</td>
</tr>
<tr>
<td>Real-time</td>
<td>✓</td>
</tr>
</tbody>
</table>

⚠️ Not robust to desynchronisation or temporal attacks.
After quantization

Shahid et al. scheme

- Modify LSB (1, 2 or 1&2) of non-zero ACs quantized coefficients whose magnitude are greater or equals to 2,
- Inter and Intra,
- RD optimization mode selection achieved on all prediction modes.

<table>
<thead>
<tr>
<th></th>
<th>✓</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imperceptible</td>
<td>✓</td>
</tr>
<tr>
<td>Photometric robustness</td>
<td>NO</td>
</tr>
<tr>
<td>Video bitrate weakly modified</td>
<td>✓</td>
</tr>
<tr>
<td>No Drift</td>
<td>✓</td>
</tr>
<tr>
<td>Real-time</td>
<td>✓</td>
</tr>
</tbody>
</table>

⚠️ Not robust to desynchronisation or temporal attacks.

Note: In [Noorkami and Mersereau, TIFS’2008], robust 0-bit watermarking, psychovisual masking, embedding in ACs coefficients and detection without knowing exact location of watermarked coefficients.
During entropy coding (codeword substitution)

Mobasseri and Raikar scheme

The algorithm creates "exceptions" in H.264 code space (portion of CAVLC) that only the decoder understands while keeping the bitstream syntax compliant.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Imperceptible</td>
<td>✓</td>
</tr>
<tr>
<td>Photometric robustness</td>
<td>NO</td>
</tr>
<tr>
<td>Video bitrate weakly modified</td>
<td>✓ (file size unchanged)</td>
</tr>
<tr>
<td>No Drift</td>
<td>✓</td>
</tr>
<tr>
<td>Real-time</td>
<td>✓</td>
</tr>
</tbody>
</table>

Watermark can be removed but cannot be forged or replaced.

⚠️ Not robust to desynchronisation or temporal attacks.
Brief conclusion

**Good news**

There are good solutions robust to photometric attacks **INSIDE H.264** (or a similar codec).

**Bad news**

Most of the solutions (all?) **INSIDE H.264** (or a similar codec) are **not robust** (or not enough robust) to **temporal and spatial desynchronizations**.
Major approaches

Before compression
- Images
  - SS [Cox et al., TIP’1997]
  - DPTC [Miller et al., TIP’2004]
  - P-QIM [Li and Cox, TIFS’2007]

Sequence of images
- Temporal watermarking
  - [Haitsma and Kalker, ICIP’2001]
  - [Chen et al., IWDW’2009]
- 3D DFT
  - [Deguillaume et al., SPIE’1999]
- On-off keying (Extended BA)
  - [Xie et al., MM&Sec2008]

Before quantization
- Luma modification:
  - [Golikeri et al., JEL’2008]
- Motion vectors modification:
  - [Zhang et al., SCGIP’2001]
- GOP structure modification:
  - [Linnartz and Talstra, ESRC’1998]

After quantization

During encoding process
- [Mobasseri and Raikar, SPIE’2007]
- [Zou and Bloom, SPIE’2009]

During Entropy coding
- [Shahid et al., EUSIPCO’2009]
- [Noorkami and Mersereau, TIFS’2008]

In an already H.264 encoded bitstream
- [Hartung and Girod, Signal Processing 1998]
- [Gong and Lu, ISM’2008]
Brief conclusion

Good news
There are good solutions robust to photometric attacks inside H.264 (or a similar codec).

Bad news
Most of the solutions (all?) inside H.264 (or a similar codec) are not robust (or not enough robust) to temporal and spatial desynchronizations.

→ What about security?
Outline

1. Preamble

2. H.264

3. Watermarking
   - Definitions
   - Video watermarking
   - Security of video watermarking
   - A practical example: the traitor tracing (active fingerprinting)

4. Conclusion & Perspectives
Definition

The classical framework of security:

Kerckhoffs’s framework

The embedding and extracting algorithms are known by the attacker and the attacker owns observations. The only secret parameter is the key.

Security attack

A security attack is an attack for which secrets parameters or secret informations are obtained.

Security subject addresses those technical points:

- Analysis and creation of secure algorithm,
- Analysis and creation of security attack.

## Security of few images schemes

Security addresses the problem of recovering secret parameters.

<table>
<thead>
<tr>
<th>Images</th>
<th>Proposed attacks</th>
</tr>
</thead>
</table>
Collusion attack

**Collusion type I occurs when:**

The **same watermark** is embedded into **different copies** of different data.

Collusion = estimate watermark from each watermarked data (e.g. average individual estimations)

Hypothesis: The watermark is often considered as noise addition. A simple estimation consequently consists in computing the difference between the watermarked data and a low-pass filtered version of it.

---

Collusion attack

Collusion type II occurs when:

**Different watermarks** are embedded into **different copies** of the same data.

Collusion = suppress watermarks thanks to a linear combination of the different watermarked data (e.g. average in order to produce unwatermarked data).

Hypothesis: Generally, averaging different watermarks converges toward zero.

---

Collusion attack

**Inter** video collusion (not specific to video):
Collusion with several videos

<table>
<thead>
<tr>
<th></th>
<th>Collusion type I</th>
<th>Collusion type II</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Copyright application</strong></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>(same watermark in ≠ videos)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Traitor tracing application</strong></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>(≠ watermarks in the same videos)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Intra** video collusion (specific to video):
collusion with just 1 video

<table>
<thead>
<tr>
<th></th>
<th>Collusion type I</th>
<th>Collusion type II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same watermark in ≠ frames of the video</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>≠ watermarks in each frame of the video (and thus in static scenes)</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>

→ main security “danger” is Intra video collusion.
Rules fighting against video \textit{intra} collusion

if two frames are quite the same,  
then the embedded watermarks should be highly correlated.

if two frames are different,  
the watermarks should be uncorrelated.

\rightarrow it is a form of informed watermarking.
Outline

1. Preamble
2. H.264
3. Watermarking
   - Definitions
   - Video watermarking
   - Security of video watermarking
   - A practical example: the traitor tracing (active fingerprinting)
4. Conclusion & Perspectives
A practical example: the traitor tracing (active fingerprinting)

Traitor tracing concept
Watermarking

A practical example: the traitor tracing (active fingerprinting)

Example of watermarking for security: traitor tracing application

An investigation experiment:


- The best probabilistic code (coming from cryptography community): The Tardos code.
- A video watermarking technique inside H.264, before quantization, taking into account RD optimization, robust to photometric attacks, and real time.
100 users maximum
20 colluders maximum
Probability accusing an innocent $10^{-3}$
User ID (codeword) on 92 104 bits

10 bits / frame
Intra CIF 352x288
25 fps
92 104 bits
$\approx$ 3 minutes

Macroblocks hiding the same bit
Spread Spectrum embedding
(ACs coefficients modification)
Collusion attacks

\( f_k \): a video frame from a colluder \( k \).
\( C \): the set of colluders.
\( K \): the number of colluders.

\[
\begin{align*}
  f_{\text{min}} &= \min\{f_k\}_{k \in C} & f_{\text{max}} &= \max\{f_k\}_{k \in C} \\
  f_{\text{avg}} &= \sum_{k \in C} \frac{f_k}{K} & f_{\text{median}} &= \text{median}\{f_k\}_{k \in C} \\
  f_{\text{minmax}} &= \frac{f_{\text{min}} + f_{\text{max}}}{2} & f_{\text{modNeg}} &= f_{\text{min}} + f_{\text{max}} - f_{\text{median}}
\end{align*}
\]

'bus', 'city', 'foreman', 'football', 'soccer', 'harbour', 'ice' and 'mobile', have been concatenated and repeated 4 times.
Detection of the colluders

<table>
<thead>
<tr>
<th>$K$</th>
<th>No. of colluders detected for attacks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>avg</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>17</td>
<td>16</td>
</tr>
<tr>
<td>20</td>
<td>18</td>
</tr>
</tbody>
</table>
Visual evaluation

DEMO
Original video
Watermarked Video
Colluded video with 17 colluders (avg attack)
Colluded video with 8 colluders (modNeg attack)
An interesting practical scheme,
but the watermarking scheme is not enough secure,
and the algorithm is not robust to spatial and temporal desynchronization.

Another interesting approach (outside H.264):


There is still lots of work...
Outline

1 Preamble

2 H.264

3 Watermarking
   - Definitions
   - Video watermarking
   - Security of video watermarking
   - A practical example: the traitor tracing (active fingerprinting)

4 Conclusion & Perspectives
Conclusion and perspectives

- lots of possible ways to do watermarking inside H.264 (depends on application)
- If **desynchronization (spatial & temporal) robustness** is a requirement
  ⇒ Very few algorithms; still an open problem.
- If **security** is a requirement (but not desynchronization (spatial & temporal) robustness)
  ⇒ Very few algorithms; still an open problem
- If **desynchronization (spatial & temporal) robustness & security** are requirements
  ⇒ The Graal quest!
End

Slides may be downloaded at: http://www.lirmm.fr/~chaumont/Publications.html

e-mail: marc.chaumont@lirmm.fr
References:

**Spread Spectrum:**

[Cox et al., TIP’1997]


**DPTC:**

[Miller et al., TIP’2004]


**Percetpual-QIM:**

[Li and Cox, TIFS’2007]

References:

**Temporal watermarking:**

[Haitsma and Kalker, ICIP’2001]


[Chen et al., IWDW’2009]


**3D DFT:**

[Deguillaume et al., SPIE’1999]


**On-off keying:**

[Xie et al., MM&Sec’2008]

References:

**Luma modification:**


**Motion vector modification:**

[2] Zhang et al., SCGIP’2001


**GOP structure modification:**


References:

[Shahid et al. EUSIPCO'2009]

[Noorkami and Mersereau, TIFS'2008]

[Hartung and Girod, Signal Processing 1998]

[Gong and Lu, ISM’2008]
References:


[Z] Zou and Bloom, SPIE'2009

References:

[Bas and Westfeld, MM&Sec’2009]

[Xie et al., SPIE’2010]

[Bas et Doërr, MM&Sec’2008]

[Mathon et al., AT’2009]

[Pérez-Freire et Pérez-González, IH' 2007]