A method for the Estimation and Recovering from General Affine Transforms in Digital Watermarking Applications

Frédéric Deguillaume, Sviatoslav Voloshynovskiy, and Thierry Pun

CUI - University of Geneva - Switzerland

http://vision.unige.ch
http://watermark.unige.ch
Outline

1. Introduction
2. Watermarking paradigm
3. A self-reference watermark
4. Estimation of general affine transforms
5. Hough transform / Radon transform approach
6. Results
7. Conclusions
1. Introduction

- Robust watermarking for copyright protection
- Requirements:

  - **Capacity**: As large as possible
  - **Robustness**: As high as possible
  - **Visibility**: As low as possible

*WM should be invisible and detectable without the original image (= oblivious)!*
2.1. Watermarking Paradigm

- **x**: cover image
- **b**: copyright message
- **K**: secret key

Diagram:
- Perceptual model
- Encoder
- Watermark Embedder
- Embedding

Symbols:
- $M$
- $y$
- $K$
- $b$
- $x$
2.1. Watermarking Paradigm

- **Embedding**
  - Perceptual model
  - Watermark Embedder
  - Encoder

- **Attacks**
  - $y'$

$x$: cover image
$b$: copyright message
$K$: secret key
2.1. Watermarking Paradigm

- **Embedding**
  - Perceptual model
  - Watermark Embedder
  - Encoder
  - Watermark

- **Attacks**
  - y' = b

- **Extraction**
  - Desynchronization estimator
  - Compensator
  - Watermark Extractor
  - Decoder

**Variables:**
- $x$: cover image
- $b$: copyright message
- $K$: secret key
2.1. Watermarking Paradigm

$\mathbf{x}$: cover image

$b$: copyright message

$K$: secret key
2.2. Watermarking Paradigm: state-of-art of geometrical transforms

- Invariant domain watermarks
  
  **Fourier-Mellin transform + log-polar or log-log map**
2.2. Watermarking Paradigm: state-of-art of geometrical transforms

- Invariant domain watermarks
  
  Fourier-Mellin transform + log-polar or log-log map

- Watermarks with additional templates
  
  Log-polar or log-log map, or exhaustive search
2.2. Watermarking Paradigm: state-of-art of geometrical transforms

- **Invariant domain watermarks**
  - Fourier-Mellin transform + log-polar or log-log map

- **Watermarks with additional templates**
  - Log-polar or log-log map,
  - or exhaustive search

- **Self-reference watermarks**...
3.1. A self-reference watermark

Message → Encoder → Encryption → Allocation in the block → Addition of reference WM

K → Upsampling → Flipping → Tiling → Watermark
3.2. A self-reference watermark: auto-correlation function (ACF)

ACF descending property

ACF extracted peaks
3.2. A self-reference watermark: magnitude spectrum (MS)

Less descending MS, dense peaks structure

MS extracted peak
3.2. A self-reference watermark: effect of distortions

Periodical WM:

Rotated periodical WM:

MS peaks (with *perceptual embedding*):
- without compression
- with JPEG, QF=50%
4.1. Estimation of general affine transforms

- An affine transform $A$ shears, flips and/or rotates the underlying grid of ACF or MS, but keeps it regular.

$$A \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x' \\ y' \end{pmatrix}$$
4.2. Estimation of general affine transforms: a noisy set of peaks

To estimate $A$, only 3 points are required, but...

Signal degradation

Signal degradation

+ rotation
4.2. Estimation of general affine transforms: a noisy set of peaks

To estimate $A$, only 3 points are required, but...

- Outlier peaks!
4.2. Estimation of general affine transforms: a noisy set of peaks

To estimate $A$, only 3 points are required, but...

- Outlier peaks!
- Missing peaks!
4.2. Estimation of general affine transforms: a noisy set of peaks

To estimate $A$, only 3 points are required, but...

- Outlier peaks!
- Missing peaks!
- Diagonals ambiguity!
4.2. Estimation of general affine transforms: a noisy set of peaks

To estimate $A$, only 3 points are required, but...

- Outlier peaks!
- Missing peaks!
- Diagonals ambiguity!
4.3. Estimation of general affine transforms: general paradigm

- The set of peaks is generally noisy:

\[ \hat{A} = \arg \min_{A \in \Phi} \left\{ \rho \left( \begin{pmatrix} x' \\ y' \end{pmatrix} - A \begin{pmatrix} x \\ y \end{pmatrix} \right) + \mu \cdot \Omega(A) \right\} \]

- \( \hat{A} \) = estimate of the affine transform within the set of possible transforms \( A \in \Phi \)
- \( \rho \) = cost function (quadratic norm for Gaussian noise, \( l_1 \) norm for Laplacian noise)
- \( \mu \) = regularization parameter
- \( \Omega(A) \) = possible variations of coefficients \( a, b, c, d \) in \( A \) (constraints)
4.3. Estimation of affine transforms: a fast and robust approximation

- We just need to estimate the main axes and periods of the underlying grid.
- Exploit the prior known regularity and redundancy:
  - => avoids exhaustive search
  - => allows high robustness
- The approach is also suitable for regular templates or any a priori known regular structure.
5.1. Hough transform (HT) / Radon transform (RT) approach

- Robust estimation of the features:
  1. Use *Hough transform (HT)* or *Radon transform (RT)* for alignments
  2. Estimate the most probable periods along each main axis

- This approach can be generalized to any other class of parametric geometrical transform, using any “suitable” robust parametric projection.
5.2. HT / RT approach : Finding the main axes

$N$ extracted points from the MS domain
5.2. HT / RT approach : Finding the main axes

\[ N \text{ extracted points from the MS domain} \]
5.2. HT / RT approach : Finding the main axes

Compute alignments contribution function $h(\theta)$: “sum” peaks from the HT or RT vertically.

$\mathcal{N}$ extracted points from the MS domain
5.2. HT / RT approach : Finding the main axes

\[ \Delta \text{extracted points from the MS domain} \]
5.2. HT / RT approach: Finding the main axes

$\mathbf{P}$ extracted points from the MS domain

$N$ extracted points from the MS domain

$H$, $\rho(\theta)$, $h$, $h(\theta)$

HT or RT

main axes

diagonals

$\theta$
5.2. HT / RT approach : Finding the main axes
5.2. HT / RT approach: Finding the main axes

... and estimate main axes directions
5.3. HT / RT approach: Estimating the periods

- **Idea:** Estimate the most probable period $\tau$ from each class of parallel lines
- **Approach:** Use a maximum likelihood (ML) estimate of a discrete signal; eg.:
  - a robust detector based on cross-correlation between a period-generated grid $G(\tau)$ and the noisy class of points $\Psi$. 
6. Results

- We are robust to affine transforms, even with:
  - Severe compression JPEG with QF up to 40-50%!
  - Printing and rescanning

- Examples:
6.1. Results

- Embedded message "Hi There"
- JPEG compressed QF=50% + rotated
- Printed, Rescanned with rotation

\[ A = \begin{pmatrix} 0.7986 & 0.6018 \\ -0.6018 & 0.7986 \end{pmatrix} \]
6.1. Results

- **ACF approach:**
  - ACF peaks
  - Hough transform
  - Fitted lines

\[
\hat{A}' = \begin{pmatrix}
0.5985 & 0.7984 \\
0.8000 & -0.6020
\end{pmatrix}
\]

**Decoded:** Hi There
6.1. Results

- MS approach:
  - MS peaks
  - Hough transform
  - Fitted lines

\[
\hat{A}'' = \begin{pmatrix}
0.6025 & 0.7984 \\
0.7991 & -0.6017
\end{pmatrix}
\]

Decoded: Hi There
## 6.2. Results

**Stirmark 3.1 score**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal enhancement</td>
<td>1.00</td>
</tr>
<tr>
<td>Compression (JPEG/GIF)</td>
<td>0.99</td>
</tr>
<tr>
<td>Scaling</td>
<td>1.00</td>
</tr>
<tr>
<td>Cropping</td>
<td>0.99</td>
</tr>
<tr>
<td>Shearing</td>
<td>1.00</td>
</tr>
<tr>
<td>Rotation (auto-crop, auto-scale)</td>
<td>0.99</td>
</tr>
<tr>
<td>Column &amp; line removal</td>
<td>1.00</td>
</tr>
<tr>
<td>Flip</td>
<td>1.00</td>
</tr>
<tr>
<td>Random geometric distortions (RBA)</td>
<td>1.00</td>
</tr>
</tbody>
</table>

---

**Total score** 0.996
7. Conclusions

- The approach shows very high robustness, due to:
  - Self-referencing WM
    => highly redundant synchronization data
  - Hough transform / Radon transform
    => robustly estimates the needed features
- The best known Stirmark 3.1 score;
  Checkmark results: coming soon.
- Many applications, not only WM
- Extension to resist random bending attack (RBA)
  => ICIP 2002, Thessaloniki, Greece, Sept. 2001
- A fast implementation of this approach is implemented in the BERKUT watermarking tool:
  [http://watermark.unige.ch](http://watermark.unige.ch) -> "Technology"