# GEOMETRICALLY INVARIANT DIGITAL WATERMARKING USING ROBUST FEATURE DETECTORS

CHI-MAN PUN

UNIVERSITY OF MACAU

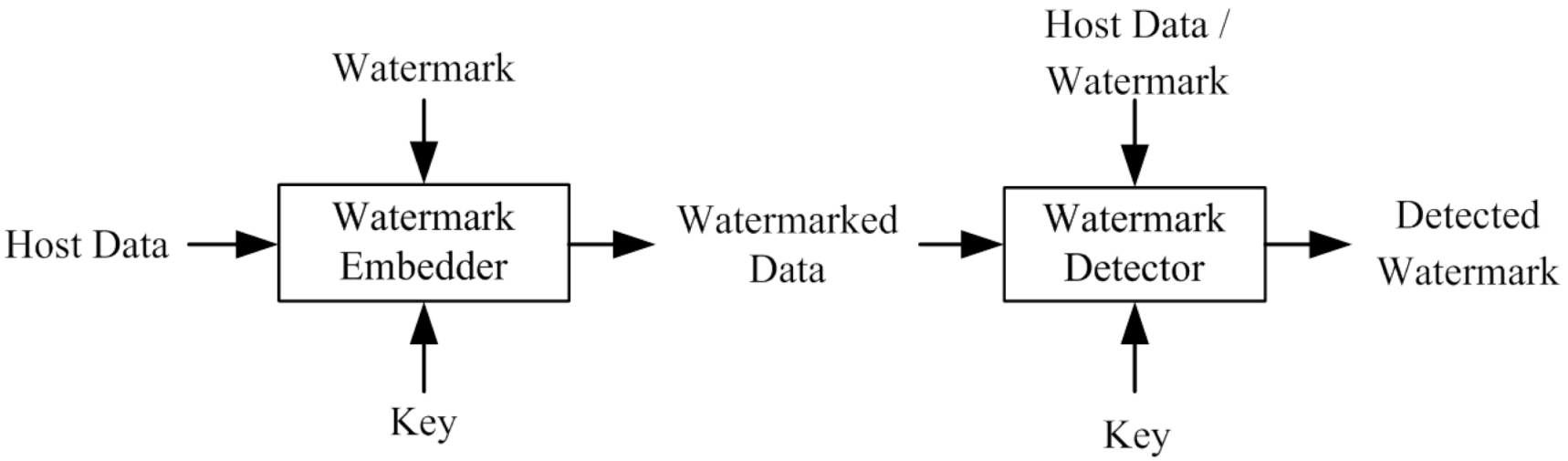
# AGENDA

- **Introduction & Related Works**
- **Digital Image Watermarking**
  - Robust Feature Detectors
  - Geometrically Invariant Watermarking Methods
  - Experimental Results
- **Digital Audio Watermarking**
  - De-synchronization Resilient Audio Watermarking
  - Experimental Results
- **Conclusions & Future Works**

# INTRODUCTION

- **Digital Watermarking Categories**
  - *Digital Image Watermarking*
  - *Digital Audio Watermarking*
  - *Digital Video Watermarking*
- **Digital Watermarking Applications**
  - *Authentication*
  - *Copyright Protection*
  - *Database Retrieval and Data Hiding*
  - *Content Description*
  - *Copy and Usage Control*

# Diagram of a Watermarking System




- **Robust Watermarking**
  - *If the watermarked data is altered, the detected watermark should still well match the watermark.*
- **Fragile Watermarking**
  - *As long as the watermarked data is slightly modified, the detected watermark should be significantly different from the watermark.*

# RELATED WORKS

- **Invariant-Domain-Based Digital Image Watermarking Scheme**
  - In [4-12], researchers have embedded the watermark in the geometric domains such as the Fourier-Mellin transform to achieve robustness to geometric attacks.
- **Histogram-Based Digital Image Watermarking Scheme**
  - In [13-16], the schemes are implemented based on histogram specifications.
- **Template-Based Digital Image Watermarking Scheme**
  - In [17-20, 27-32], a template is embedded in addition to the watermark as side information to calculate the geometric variations and return the destroyed image to the corresponding original shape before extracting the watermark.
- **Feature-Based Digital Image Watermarking Scheme**
  - In [21,23,24,33], the watermark is embedded into regions invariant to geometric attacks.
- **Decomposition-Based Digital Image Watermarking Scheme**
  - In [25,26], the image / watermark was decomposed into components based on a set of function, and consequently the watermark was embedded

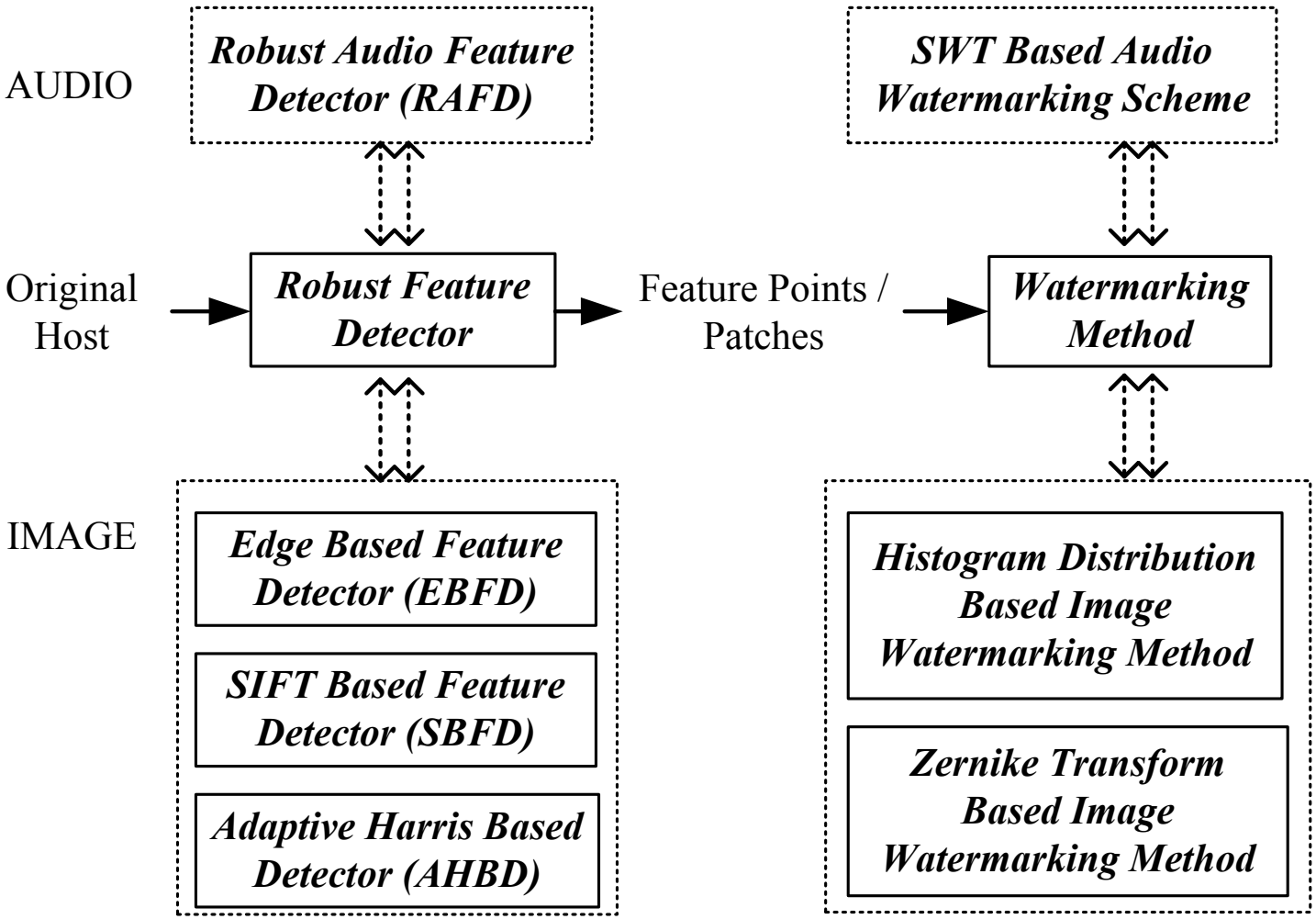
# CHALLENGE / MOTIVATION

- O'Ruanaidh and Pun [4] calculated the DFT of the host image and applied a Fourier-Mellin Transform (FMT) to the magnitude.
  - **However, both the Log-Polar Mapping (LPM) and Inverse LPM can bring severe distortion to the watermarks and images.**
- Kim et al. [5] used the invariant centroid and reordered FMT to implement a RST invariant watermarking scheme for images.
  - **However, the watermark would be damaged since it needs to go through ILPM.**
- Zheng et al. [7,8] embedded watermarks in the LPM of the Fourier magnitude spectrum of an image.
  - **However, this method depends on the original image, so an exhaustive search must be performed if the original image is unavailable.**
- Kang et al. [18] developed the DWT-DFT composite watermarking method. It performs well against both affine transformations and low-quality JPEG compression.
  - **However, the embedded templates can be removed to generate peaks in a particular transform domain.**
- Tang and Hang [22] adopted Mexican Hat wavelet filtering to extract feature points, and thus embed the watermark in the normalized local regions centered at the feature points.
  - **However, it was clearly shown in the experimental results that the performance of the scheme is not good against rotation.**

# RESEARCH GOALS

- Among the watermarking schemes, the **feature extraction based methods** are proved to be superior to others
  - **Robust feature detector** bears the brunt of the importance
  - the extracted feature points / patches should be **robust against most of the attacks to the greatest extent**
- After extracting the feature points / patches, the **watermarking method** is the next vital thing to consider.
  - Binary data bits sequence
  - Signal sequence of certain distribution

# METHODOLOGY / FRAMEWORK



# AGENDA

- **Introduction & Related Works**
- **Digital Image Watermarking**
  - Robust Feature Detectors
  - Geometrically Invariant Watermarking Methods
  - Experimental Results
- **Digital Audio Watermarking**
  - De-synchronization Resilient Audio Watermarking
  - Experimental Results
- **Conclusions & Future Works**

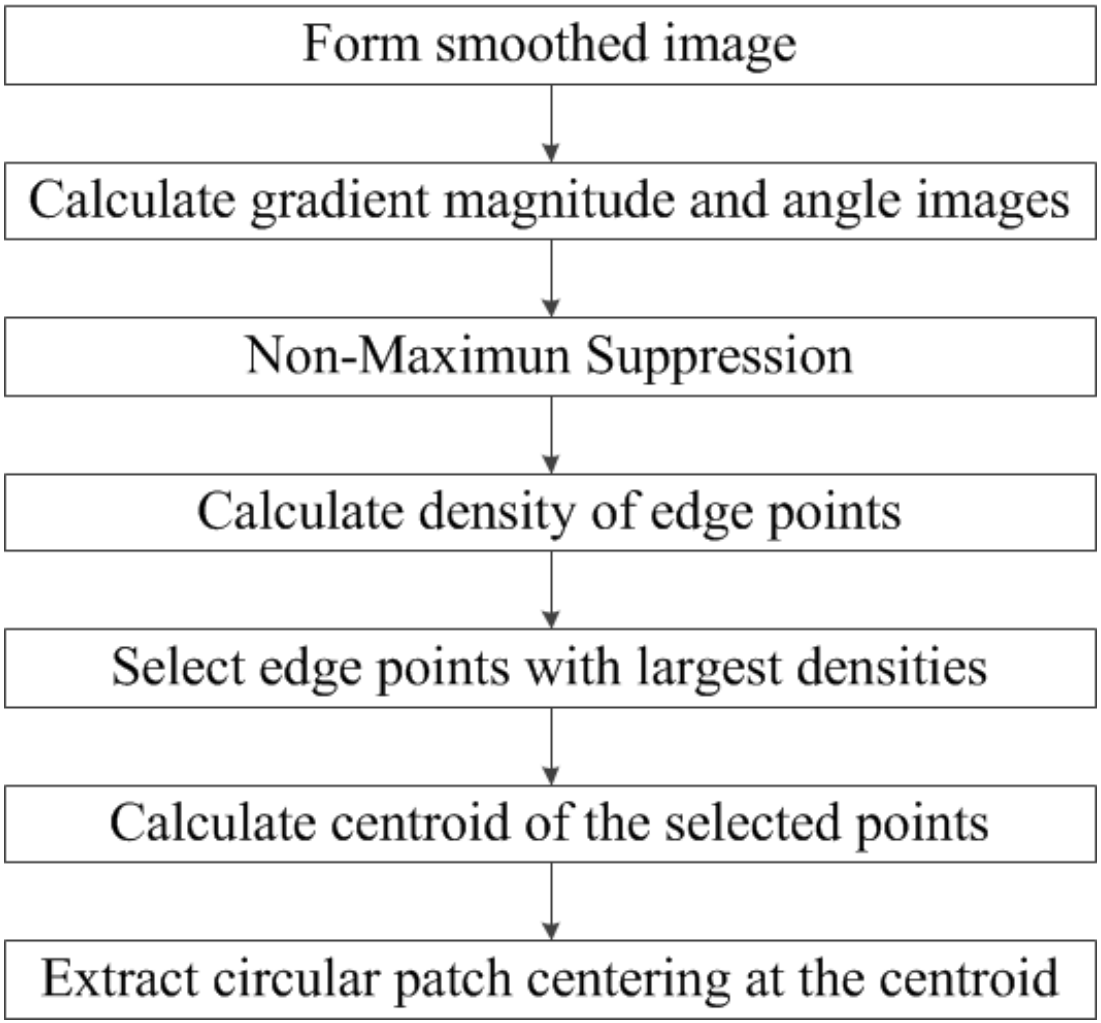
# Existing Feature Detectors

- **Difference-of-Gaussians (DoG)**
  - [36-39]. Extracts feature points by approximating the Laplacian of Gaussian; scale invariant.
- **Speed Up Robust Features (SURF)**
  - [40]. Computes the fraction of pixels within a neighborhood which have similar intensity to the center pixel; scale invariant.
- **Features from Accelerated Segment Test (FAST)**
  - [41]. Compares pixels only on a circle of fixed radius around the pixel.
- **Harris Detector**
  - [42]. Second moment matrix.
- **Smallest Univalue Segment Assimilating Nucleus (SUSAN)**
  - [43]. Uses the morphological approach to detect the features.

# Proposed Feature Detectors

- **Edge Based Feature Detector (EBFD)**
  - extract unique feature in the specific region
- **SIFT Based Feature Detector (SBFD)**
  - number of reliable feature points
  - the descriptor is needed for relocating the features
- **Adaptive Harris Based Detector (AHBD)**
  - number of reliable feature points
  - no extra information needed

# EDGE BASED FEATURE DETECTOR (EBFD)




$$f_s(x,y) = G(x,y) * f(x,y)$$

$$G(x,y) = e^{-(x^2+y^2)/2\sigma^2}$$

$$M(x,y) = mag(\nabla f) = \sqrt{g_x^2 + g_y^2}$$

$$\alpha(x,y) = \tan^{-1}(g_y / g_x)$$

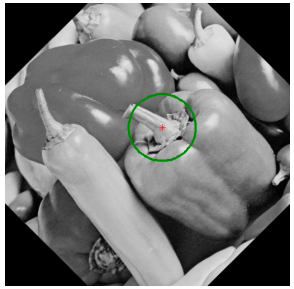
$$D\{d_i\}$$

# EBFD Extracted Circular Patches Under Various Attacks

(a) Original 'Pepper'

(b) Rotation_45

(c) Cropping_30%

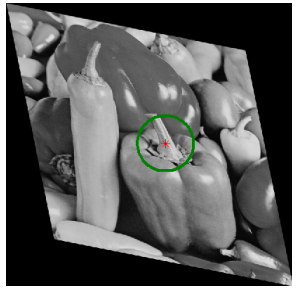


(d) Rotation_45 & Cropping

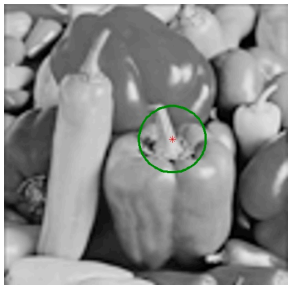
(e) Jitter_10_20

(f) Vertical Shearing_30%

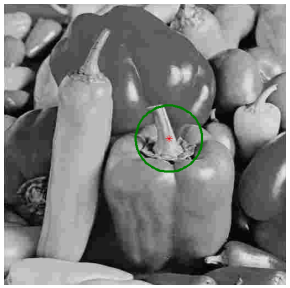
(g) Horizontal Shearing_30%



(h) Affine Transform_20%



(i) Scaling_0.2

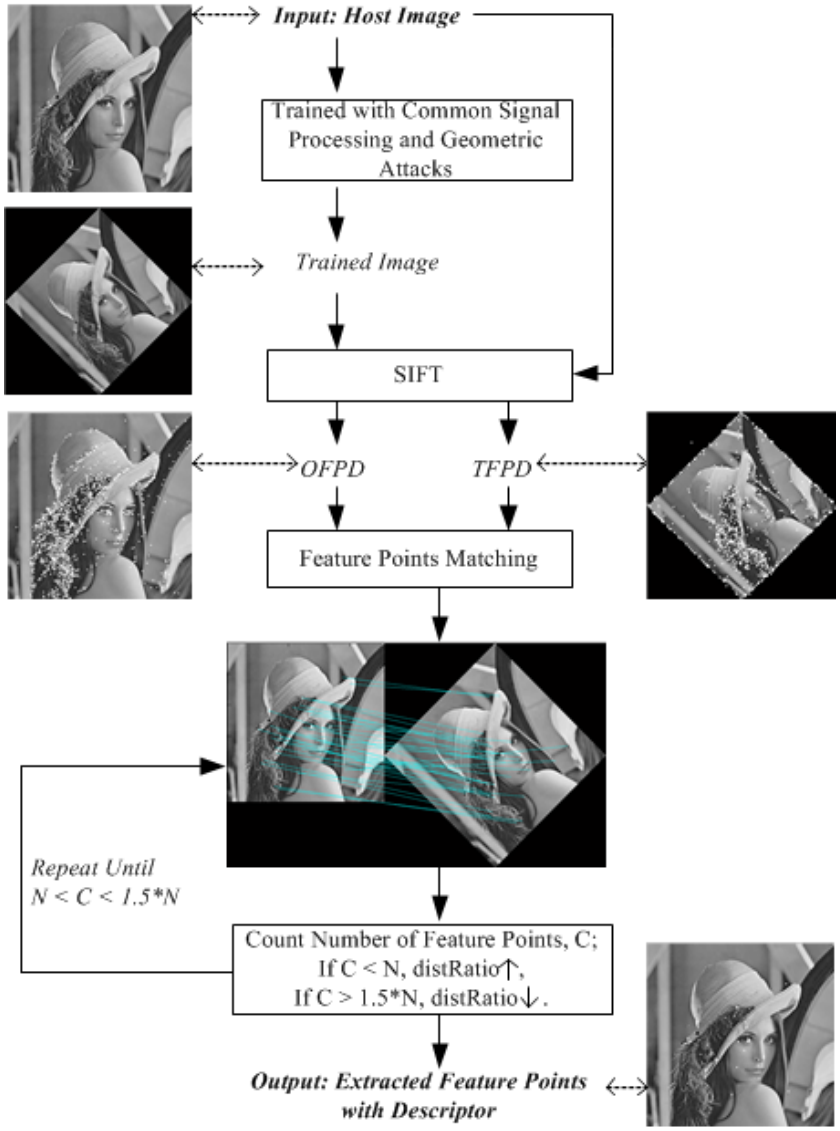


(j) JPEG_10

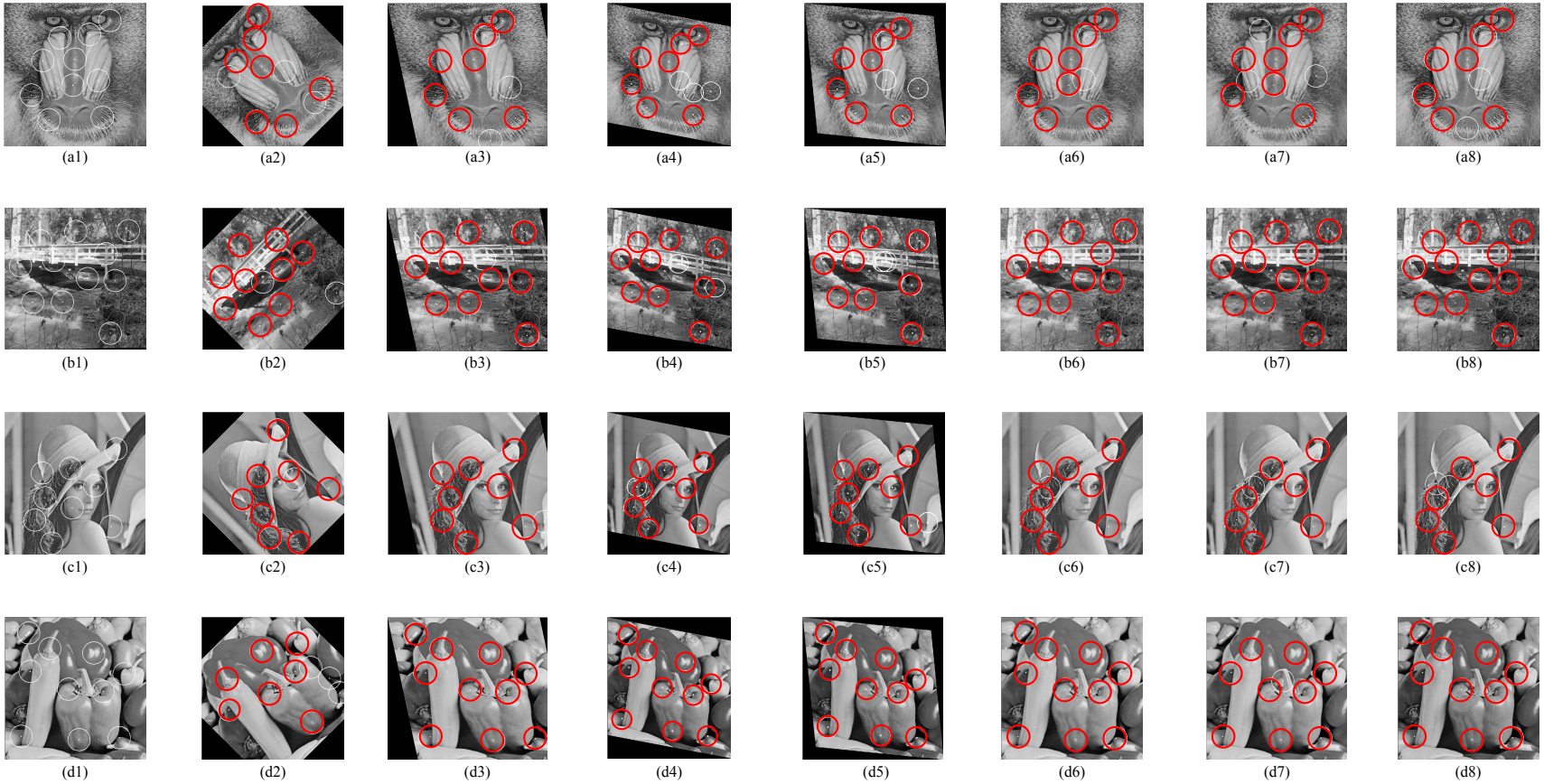
(k) Median Filtering_4x4

(l) Gaussian Filtering_9x9, 1.5

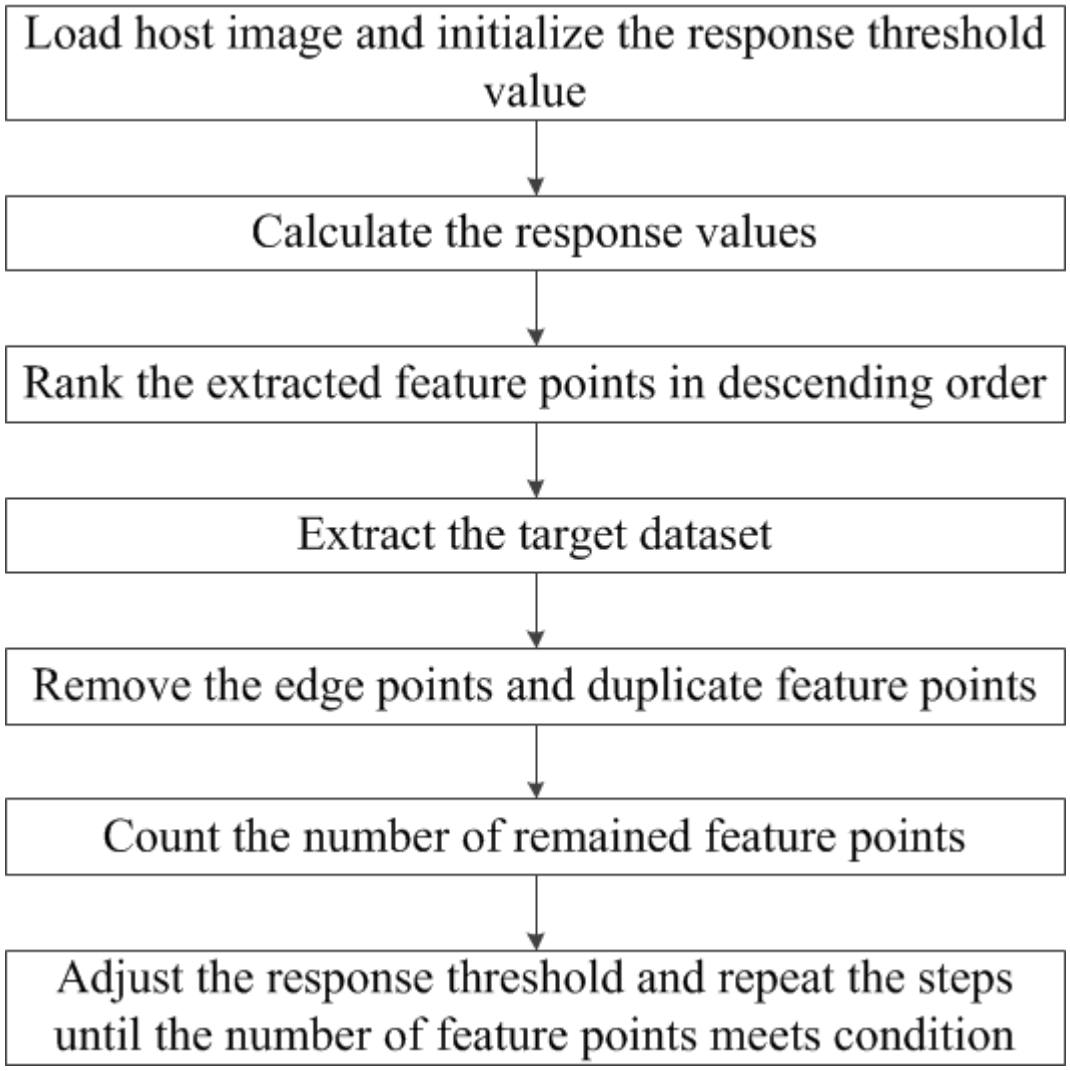
# SIFT BASED FEATURE DETECTOR (SBFD)



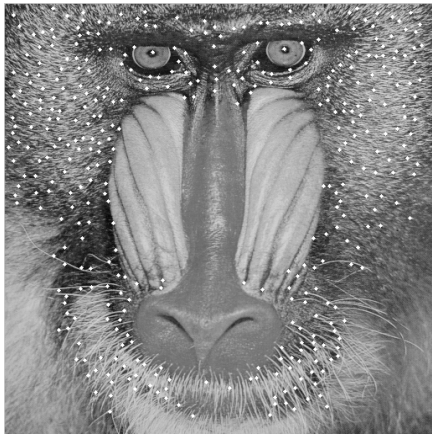
# SBFD Feature Extraction Under Various Attacks



# ADAPTIVE HARRIS BASED DETECTOR (AHBD)



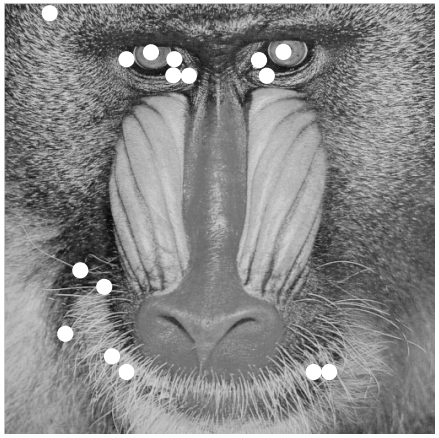
# AHBD Feature Extraction Comparison



(a1)

(b1)

(c1)

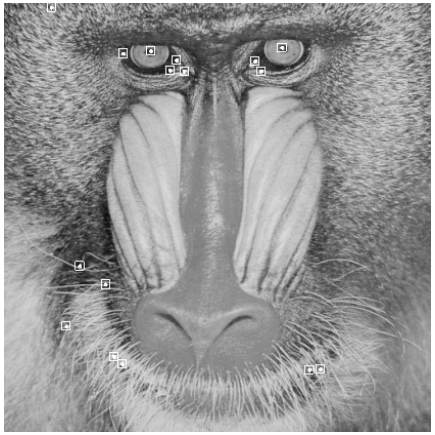


(a2)

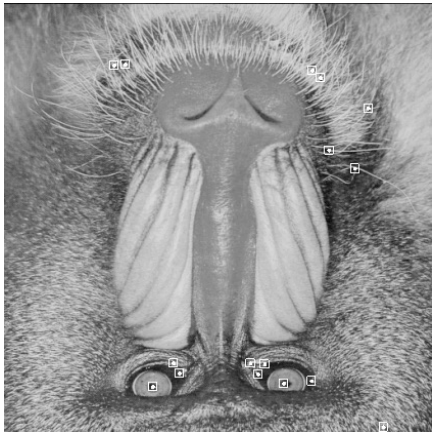
(b2)

(c2)

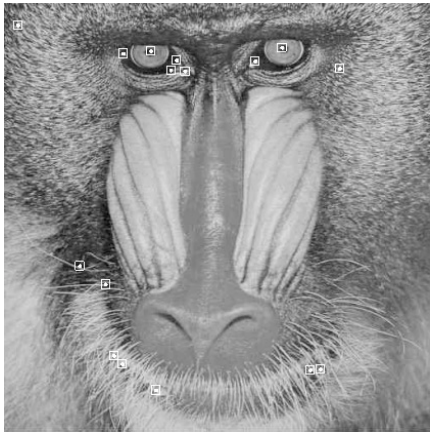
# AHBD Feature Extraction Under Various Attacks



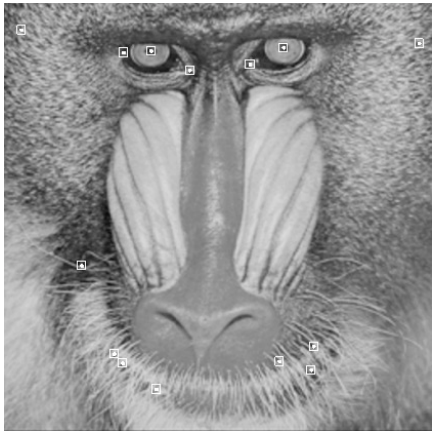
(a)



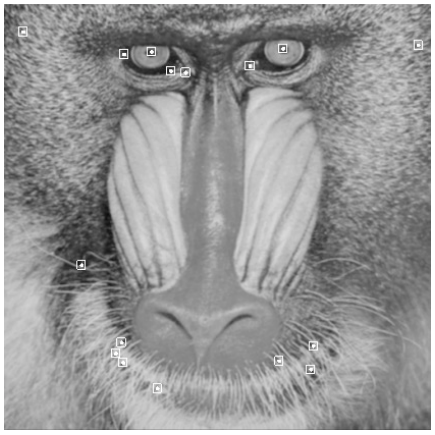
(b)



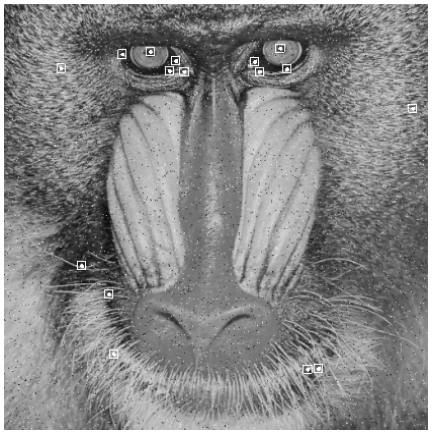
(c)



(d)



(e)



(f)

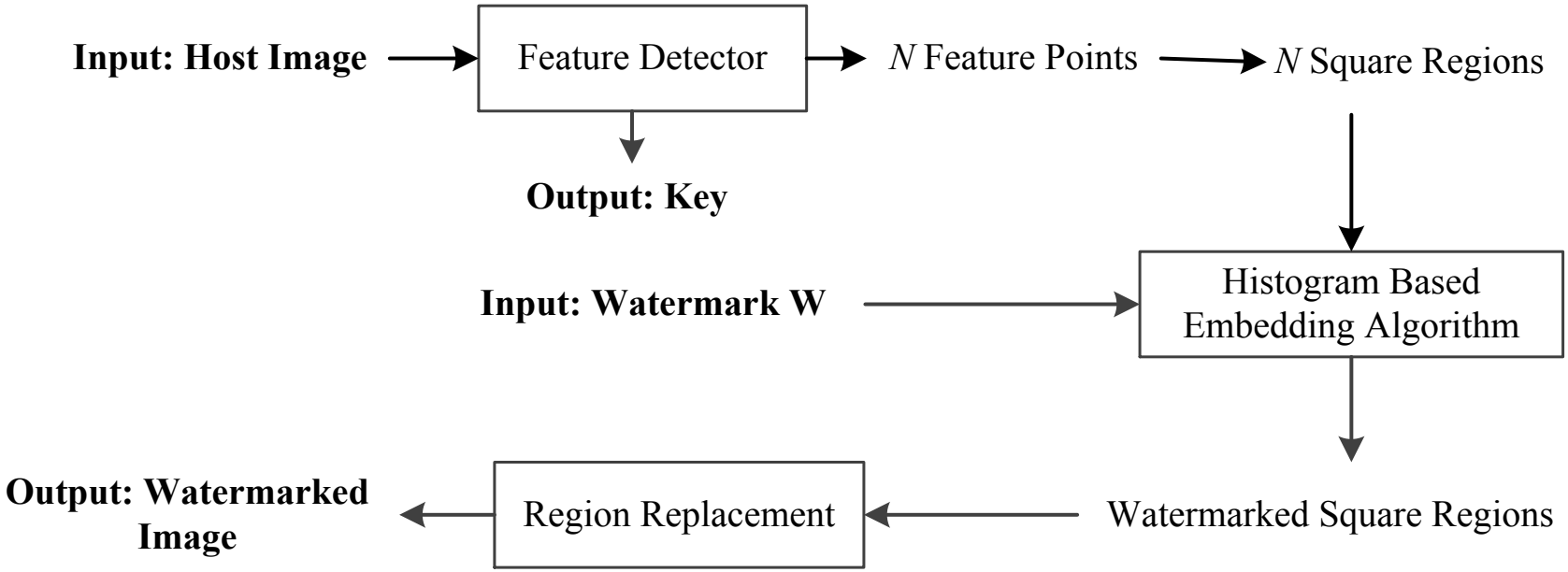
# AGENDA

- **Introduction & Related Works**
- **Digital Image Watermarking**
  - Robust Feature Detectors
  - Geometrically Invariant Watermarking Methods
  - Experimental Results
- **Digital Audio Watermarking**
  - De-synchronization Resilient Audio Watermarking
  - Experimental Results
- **Conclusions & Future Works**

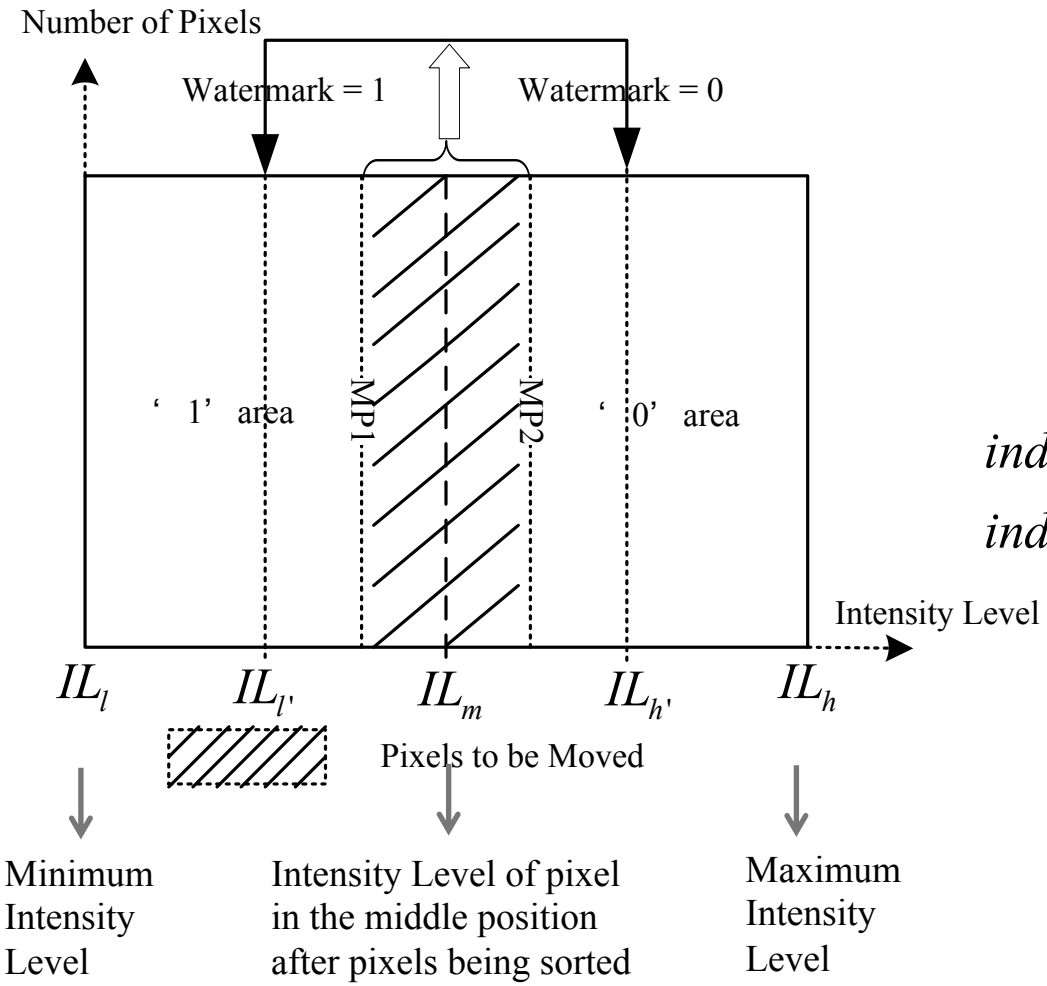
# GEOMETRICALLY INVARIANT WATERMARKING METHODS

- **Histogram Distribution Based Watermarking Method**
  - Sequence of binary data bits
  - Intensity-level histogram
- **Zernike Transform Based Watermarking Method**
  - Sequence of specific distribution
  - Circular patch decomposition
  - Magnitudes of local Zernike moments
  - Linear correlation

# Histogram Distribution Based Watermark Embedding



# Histogram Modification

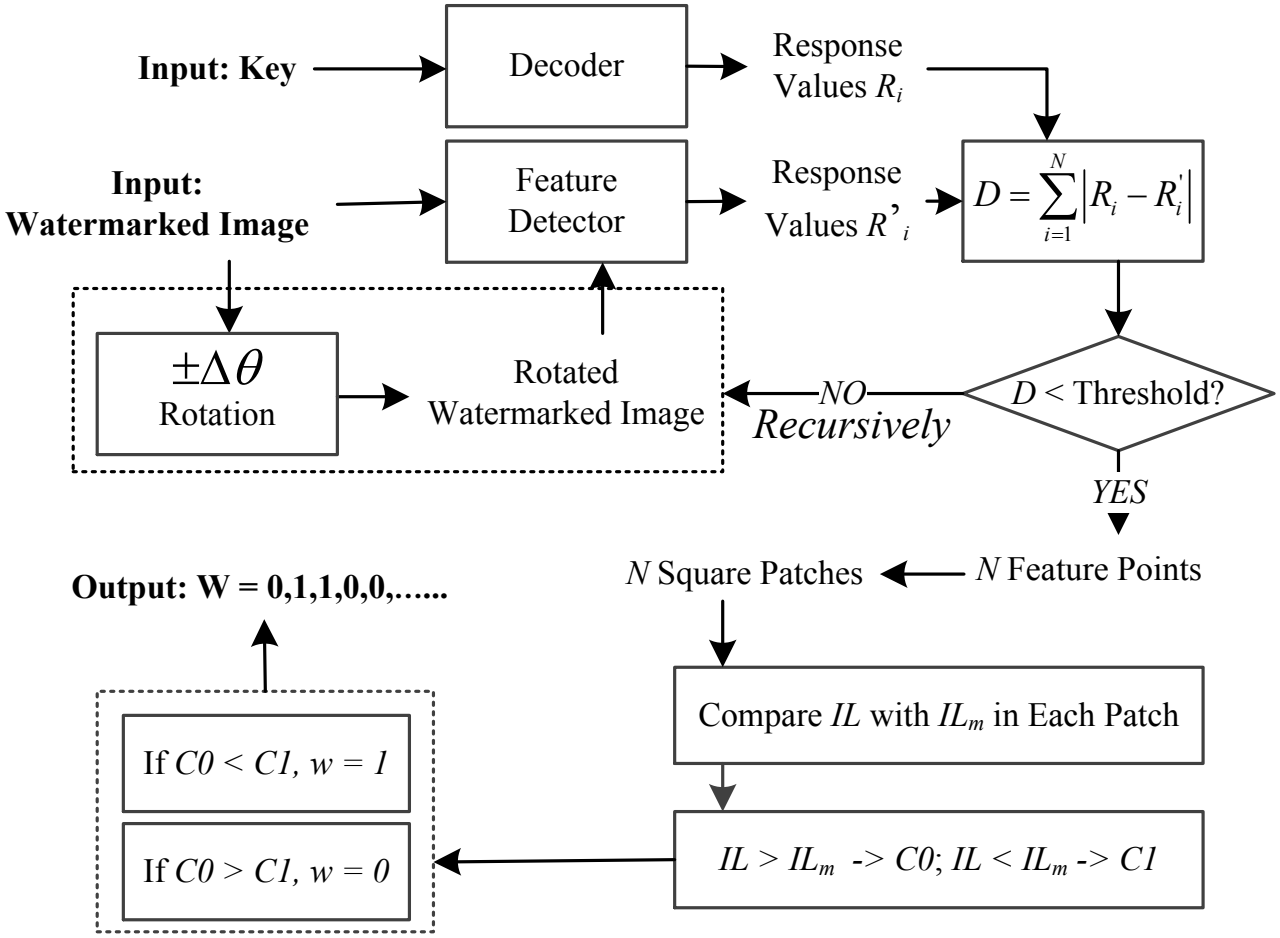



$$IL_{l'} = IL_m - \gamma(IL_m - IL_l)$$
$$IL_{h'} = IL_m + \gamma(IL_h - IL_m)$$

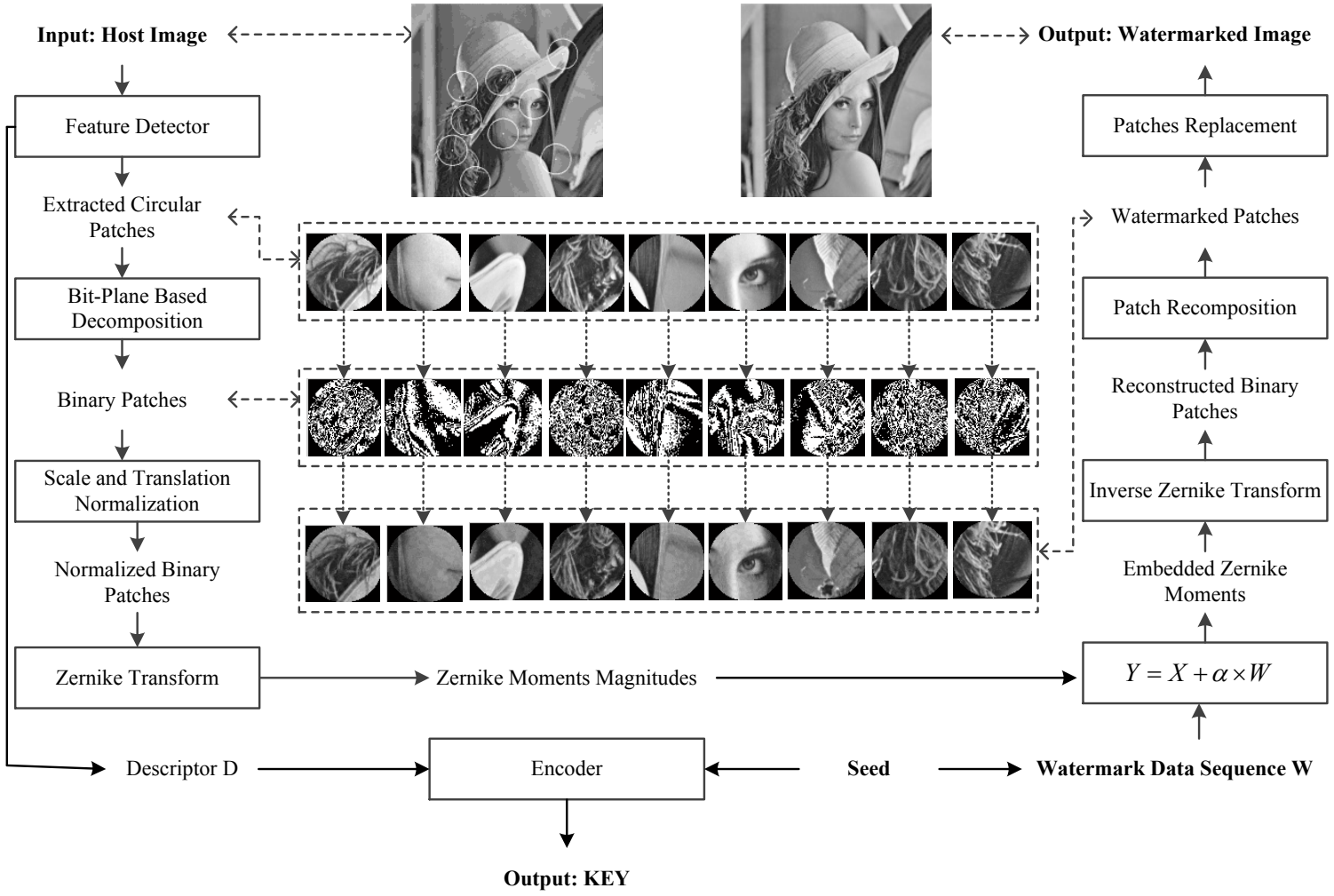
$$index(MP1) = index(IL_m) - N\_MP/2$$
$$index(MP2) = index(IL_m) + N\_MP/2$$

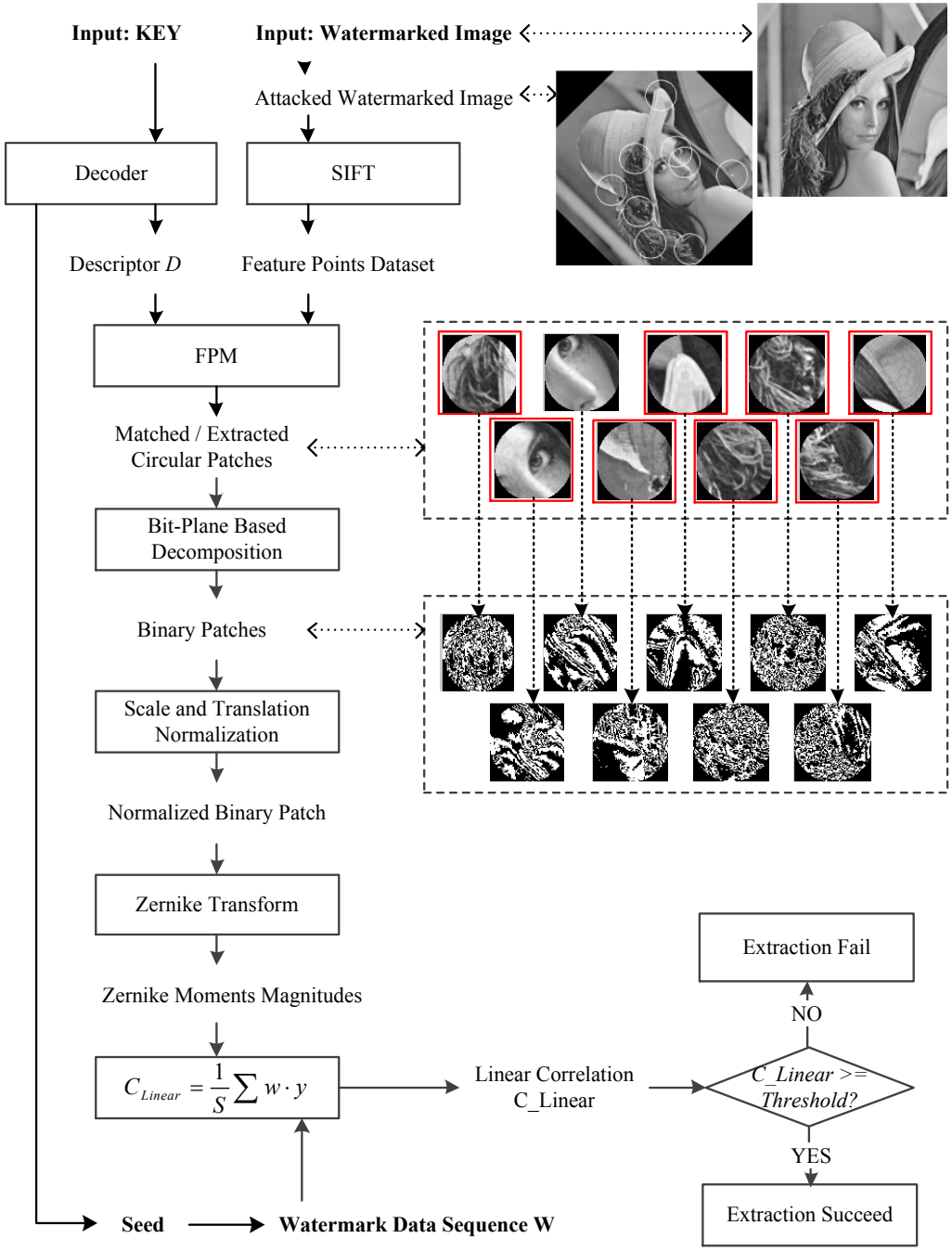
$$N\_MP = S\_M \times S\_N \times a$$

# Histogram Distribution Based Watermark Extraction



# Zernike Transform Based Watermark Embedding






# Zernike Transform Based Watermark Extraction

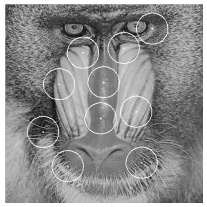
# Zernike Transform Based Watermarking

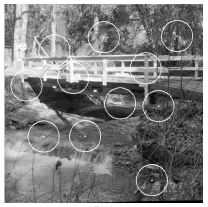
- $Y = X + \alpha \times W$ **Spread spectrum communication techniques**
  - X - Zernike moments magnitudes;
  - $\alpha$ - Predefined parameter to control the watermark embedding strength;
  - $W$ - random watermark sequence of Gaussian distribution.
- $I = I_{m-1} \cdot 2^{m-1} + I_{m-2} \cdot 2^{m-2} + \ldots + I_1 \cdot 2^1 + I_0 \cdot 2^0$ (4.11)
  - $I_i$ - corresponding decomposed bit plane patch.
- $C_{Linear} = \frac{1}{S} \sum w \cdot y$ (4.12)
  - S - size of the Zernike moments magnitudes;
  - w - watermark data sequence;
  - y - watermarked data.

# AGENDA

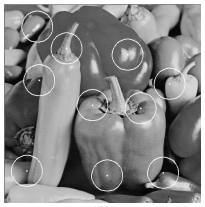
- **Introduction & Related Works**
- **Digital Image Watermarking**
  - Robust Feature Detectors
  - Geometrically Invariant Watermarking Methods
  - Experimental Results
- **Digital Audio Watermarking**
  - De-synchronization Resilient Audio Watermarking
  - Experimental Results
- **Conclusions & Future Works**

# SBFD Extracted Features and Watermarked Images


(a1)

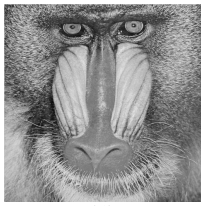

(b1)


(c1)


(d1)

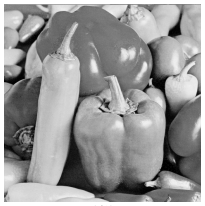

(e1)


(f1)


(a2)


(b2)


(c2)

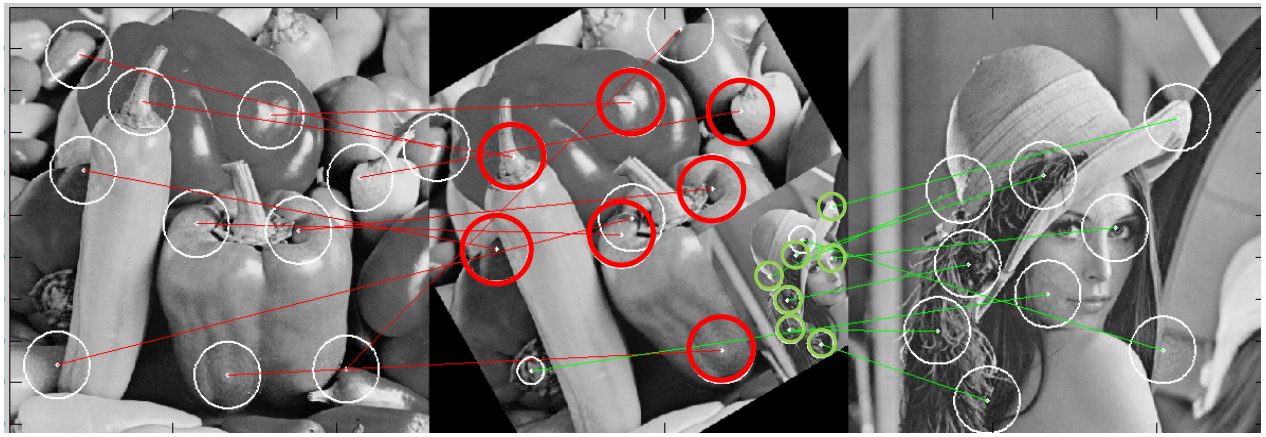

(d2)


(e2)

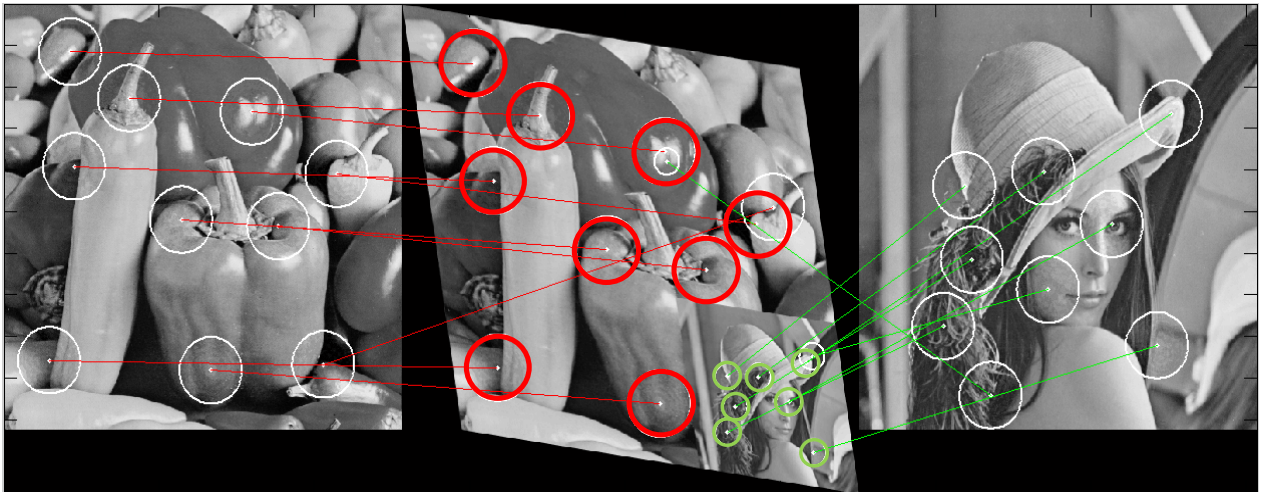

(f2)

| ‘Baboon’ | ‘Bridge’ | ‘Lena’ | ‘Pepper’ | Blurry ‘Scene’ | Blurry ‘Jet’ |
|---|---|---|---|---|---|
| PSNR: 39.10dB ; | PSNR: 39.55dB ; | PSNR: 39.96dB ; | PSNR: 38.93dB ; | PSNR: 39.89dB ; | PSNR: 38.85dB ; |
| AVG_PSNR: 32.43dB | AVG_PSNR: 32.78dB | AVG_PSNR: 32.61dB | AVG_PSNR: 32.78dB | AVG_PSNR: 32.45dB | AVG_PSNR: 32.55dB |

# SBFD and Zernike Transform Results Under Mixed Attacks

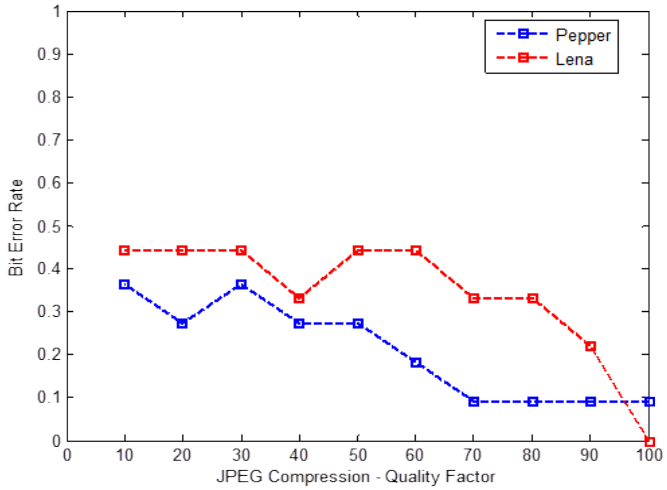


(a)

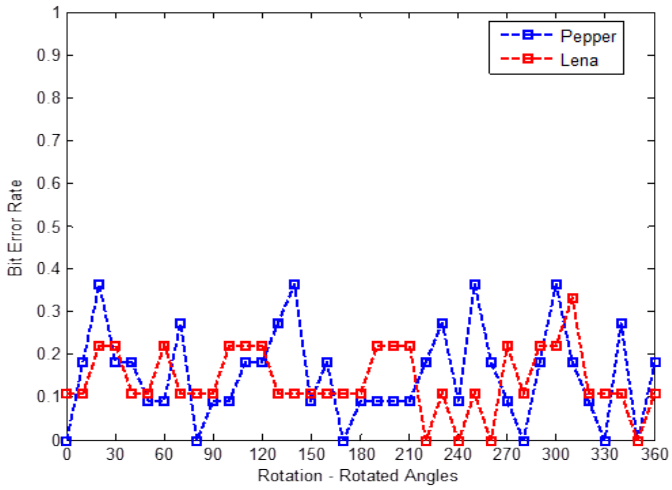


(b)

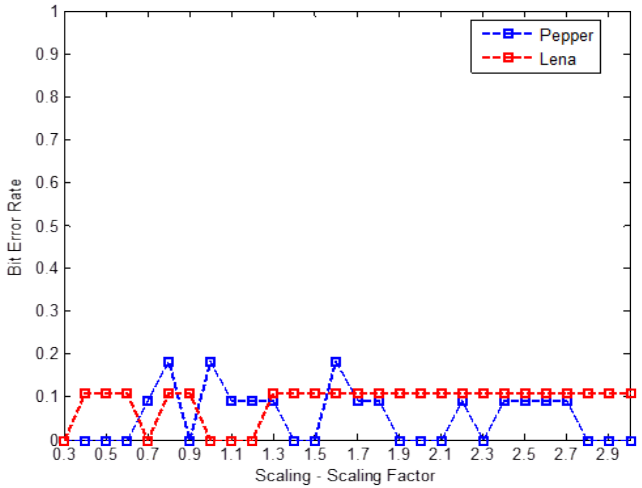
# Bit-Error Rates against Strength of Attacks



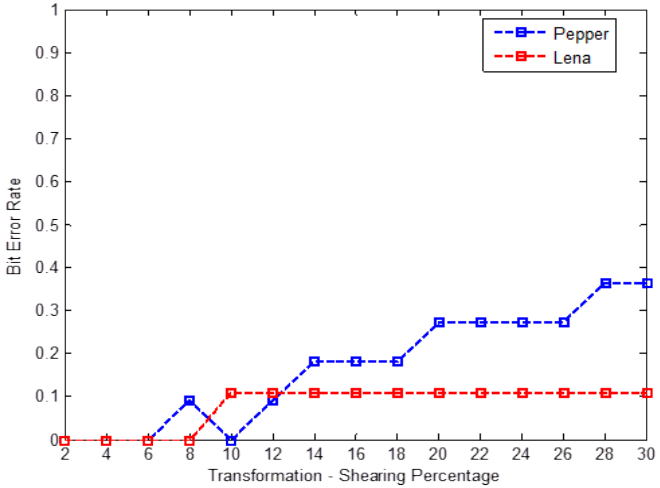

(a)




(b)




(c)




(d)

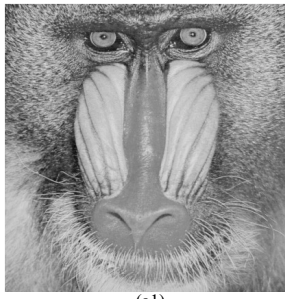
# SBFD and Zernike Transform Results Under Common Signal Processing

| Attacks | Baboon | | | | Lena | | | | Pepper | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Ours | 2010 [87] | 2006 [86] | 2003 [22] | Ours | 2010 [87] | 2006 [86] | 2003 [22] | Ours | 2010 [87] | 2006 [86] | 2003 [22] |
| Median filter 3 x 3 | 9/10 | 6/15 | 0/13 | 2/11 | 8/9 | 6/10 | 2/11 | 1/8 | 9/10 | 17/23 | 2/13 | 1/4 |
| Gaussian filter 3 x 3 | 9/10 | 4/15 | 1/13 | 7/11 | 8/9 | 5/10 | 1/11 | 3/8 | 9/10 | 9/23 | 2/13 | 1/4 |
| JPEG 90 | 9/10 | 8/15 | 3/13 | 9/11 | 8/9 | 6/10 | 3/11 | 6/8 | 10/10 | 16/23 | 4/13 | 3/4 |
| JPEG 70 | 8/10 | 8/15 | 2/13 | 8/11 | 8/9 | 4/10 | 3/11 | 5/8 | 10/10 | 14/23 | 2/13 | 3/4 |
| JPEG 50 | 7/10 | 8/15 | 0/13 | 6/11 | 7/9 | 6/10 | 1/11 | 4/8 | 10/10 | 14/23 | 1/13 | 2/4 |
| JPEG 30 | 7/10 | 8/15 | 0/13 | 4/11 | 7/9 | 5/10 | 1/11 | 2/8 | 8/10 | 10/23 | 1/13 | 0/4 |
| Median 3 x 3 + JPEG 90 | 9/10 | 5/15 | 0/13 | 1/11 | 8/9 | 3/10 | 2/11 | 1/8 | 9/10 | 16/23 | 2/13 | 1/4 |
| Gaussian 3 x 3 + JPEG 90 | 9/10 | 4/15 | 1/13 | 7/11 | 8/9 | 4/10 | 1/11 | 3/8 | 9/10 | 5/23 | 2/13 | 1/4 |

# SBFD and Zernike Transform Results Under Geometric Attacks

| Attacks | Baboon | | | | Lena | | | | Pepper | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Ours | 2010 [87] | 2006 [86] | 2003 [22] | Ours | 2010 [87] | 2006 [86] | 2003 [22] | Ours | 2010 [87] | 2006 [86] | 2003 [22] |
| Cropping (10%off) | 8/10 | 6/15 | 1/13 | 1/11 | 8/9 | 5/ 10 | 4/11 | 2/8 | 9/10 | 6/23 | 1/13 | 2/4 |
| Scaling (1.5) | 8/10 | 7/15 | 3/13 | 0/11 | 8/9 | 6/10 | 3/11 | 0/8 | 9/10 | 13/23 | 4/13 | 0/4 |
| Scaling (0.7) | 7/10 | 2/15 | 0/13 | 0/11 | 8/9 | 4/10 | 1/11 | 0/8 | 9/10 | 6/23 | 0/13 | 0/4 |
| Rotation 5 | 9/10 | 5/15 | 0/13 | 2/11 | 8/9 | 4/10 | 3/11 | 2/8 | 9/10 | 9/23 | 2/13 | 0/4 |
| Rotation 30 | 9/10 | 4/15 | 0/13 | 0/11 | 8/9 | 4/10 | 0/11 | 0/8 | 9/10 | 7/23 | 0/13 | 0/4 |
| Shearing (1%) | 7/10 | 6/15 | 0/13 | 4/11 | 9/9 | 4/10 | 2/11 | 2/8 | 8/10 | 7/23 | 0/13 | 1/4 |
| Removed 5 rows and 17 columns | 6/10 | 6/15 | 2/13 | 2/11 | 5/9 | 5/10 | 3/11 | 1/8 | 6/10 | 11/23 | 0/13 | 0/4 |
| Cropping (10% off) + JPEG 70 | 8/10 | 6/15 | 0/13 | 1/11 | 8/9 | 5/10 | 2/11 | 1/8 | 9/10 | 6/23 | 0/13 | 2/4 |
| Rotation5+ Cropping+JPEG 70 | 7/10 | 5/15 | 0/13 | 0/11 | 7/9 | 5/10 | 2/11 | 0/8 | 9/10 | 7/23 | 0/13 | 0/4 |
| Removed5rows17columns + JPEG70 | 6/10 | 6/15 | 1/13 | 2/11 | 5/9 | 4/10 | 1/11 | 0/8 | 6/10 | 7/23 | 0/13 | 0/4 |

# AHBD and Histogram Distribution Based Watermarked Images

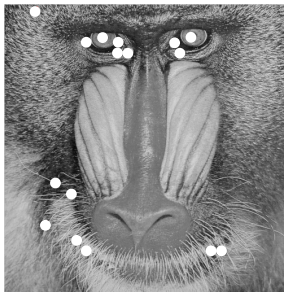

(a1)


(b1)


(c1)

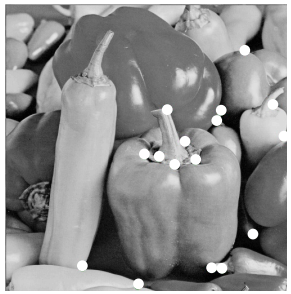

(d1)

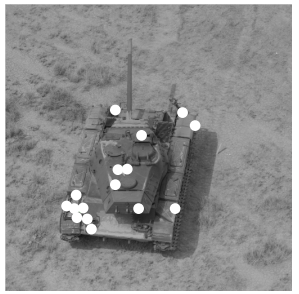

(e1)

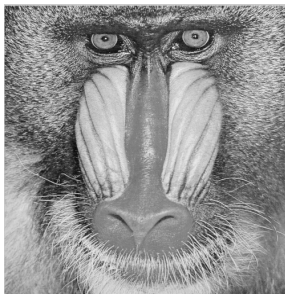

(a2)


(b2)


(c2)


(d2)


(e2)


(a3)

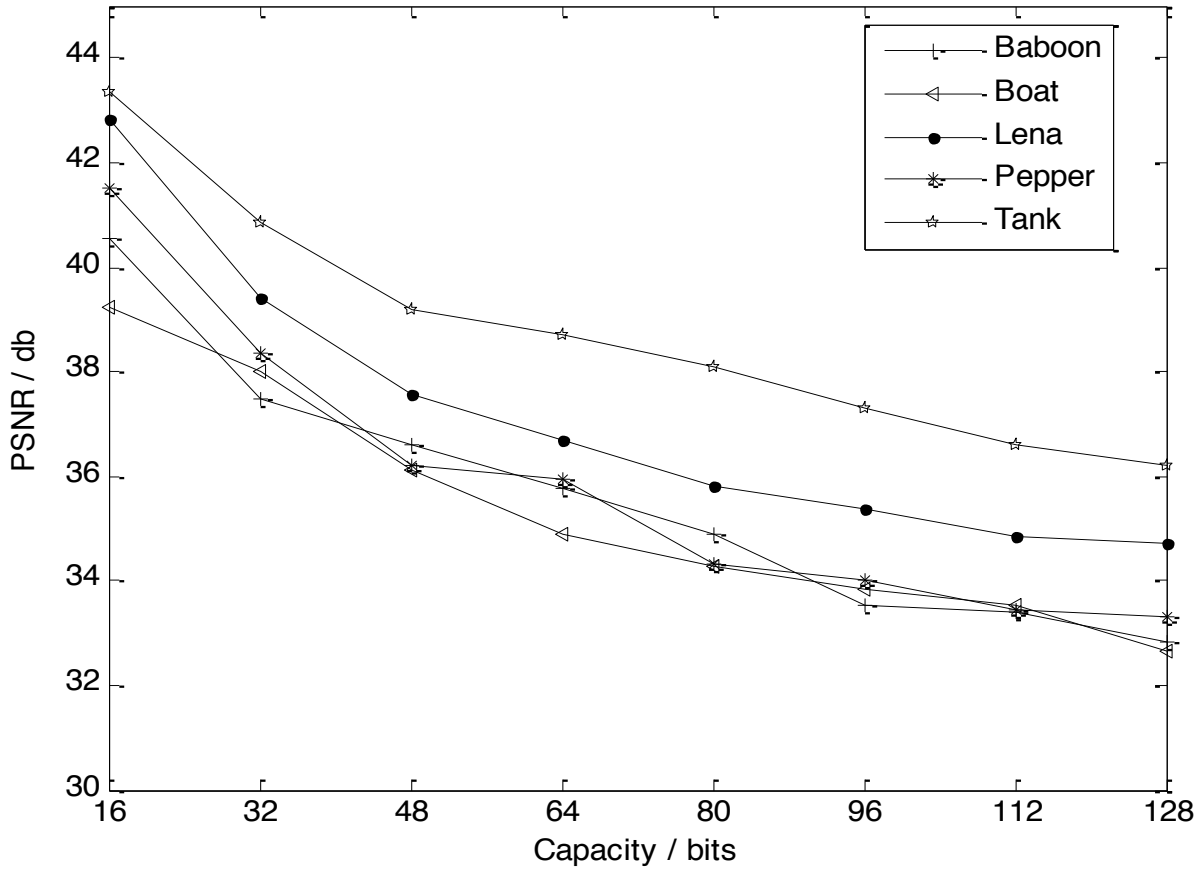

(b3)


(c3)


(d3)


(e3)

# Relationships between Capacity and Transparency



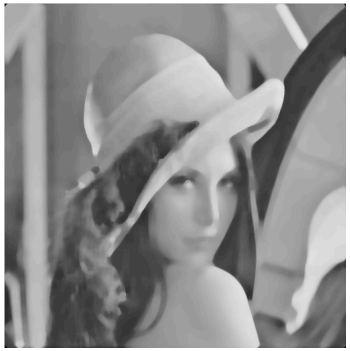
# Various Attacks and Correctly Extracted Bits

(a)

(b)

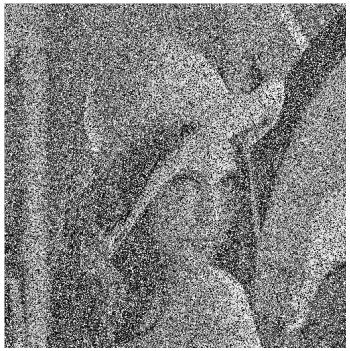
(c)

(d)

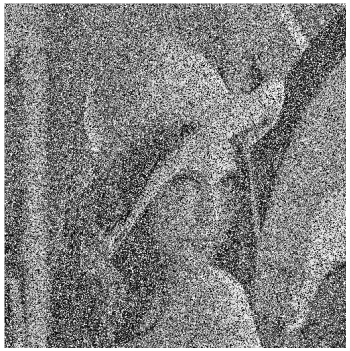


(e)

(f)

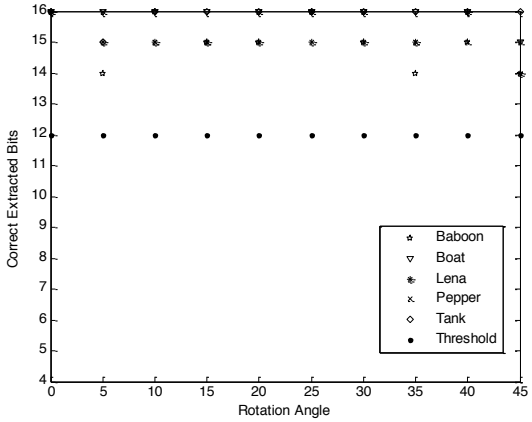


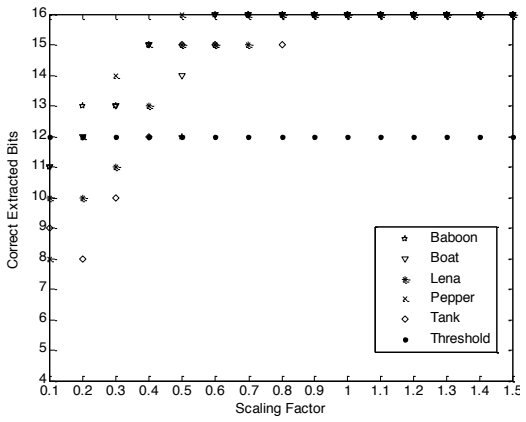
(g)

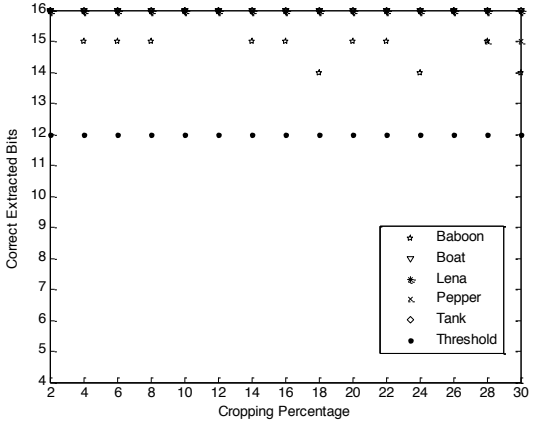


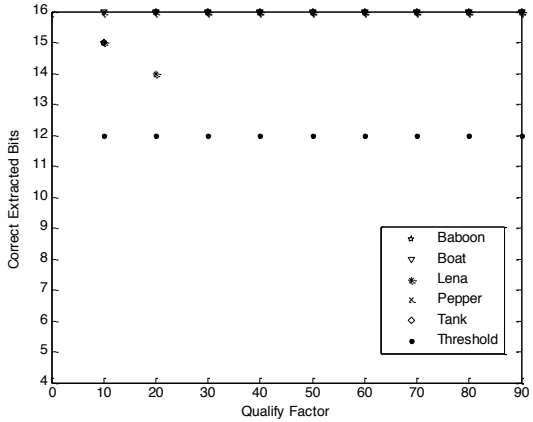
(h)

# Correctly Extracted Bits When Under Various Attacks

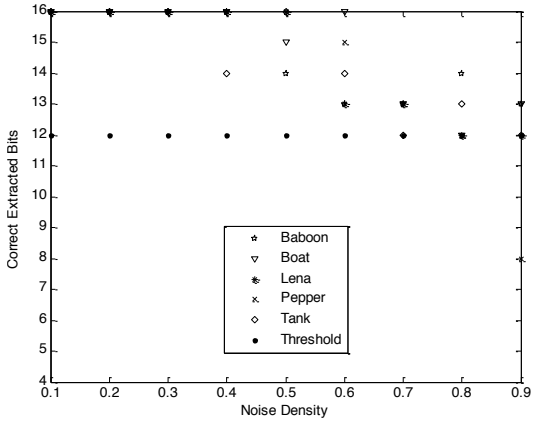


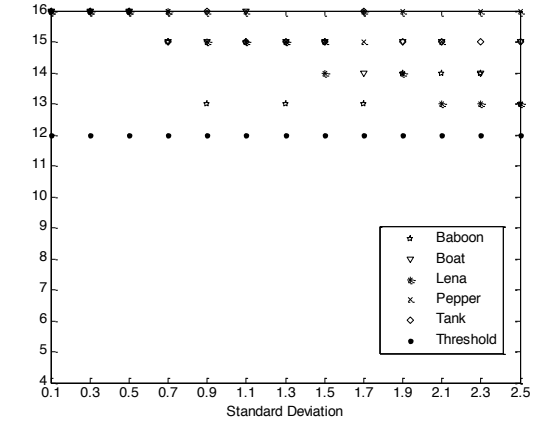




(d) JPEG compression




(e) ‘ Salt & Pepper’ Noise Pollution




(f) Gaussian Low-Pass Filtering

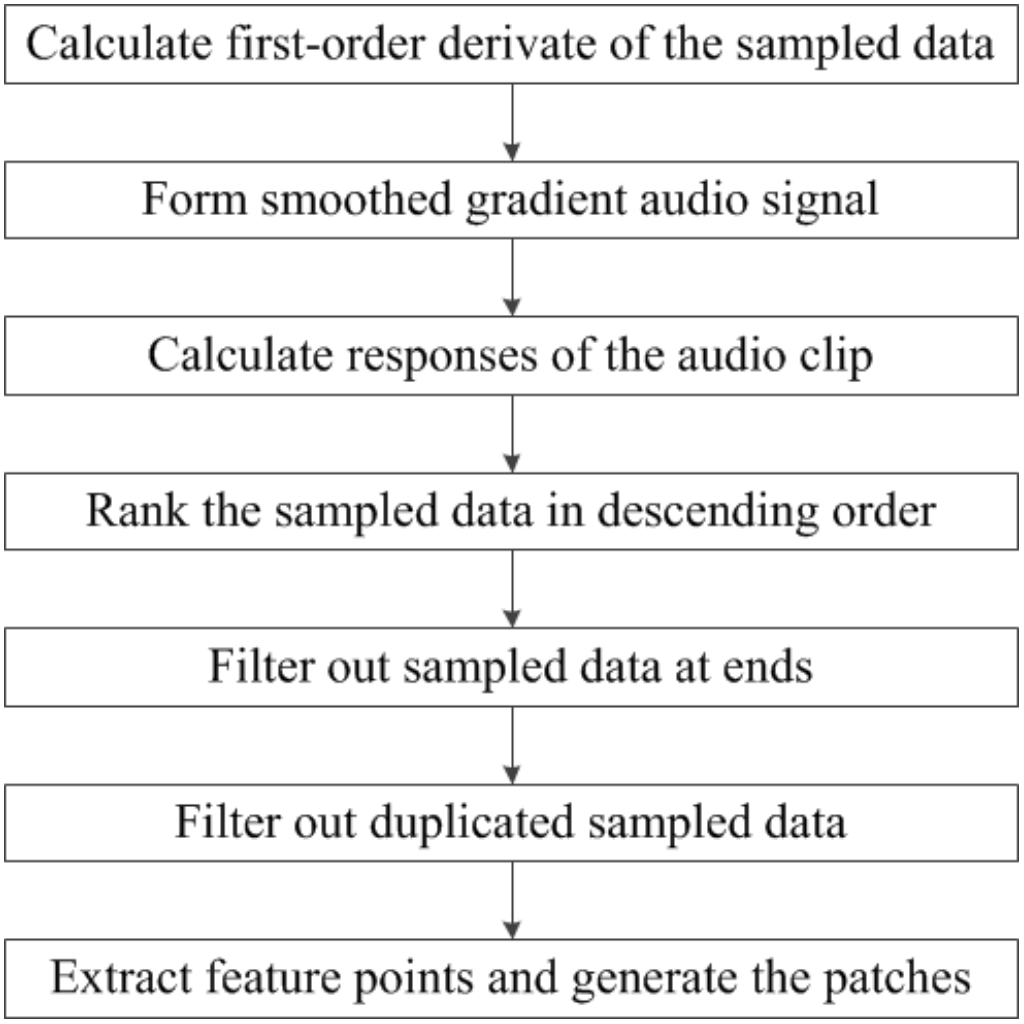
# Results Comparison

| Attacks | Methods | | | |
|---|---|---|---|---|
| | Tang and Hang [22] | Zheng et al. [65] | SBFD and Zernike Based Scheme | AHBD and Histogram Based Scheme |
| Image Rotation | $1^o$ – $5^o$ | $0^o$ – $360^o$ | $0^o$ – $360^o$ | $0^o$ – $360^o$ |
| Image Scaling | – | 0.7 – 1.8 | 0.3 – 3 | 0.3 – 4 |
| JPEG Compression | 40 – 100 | 10 – 100 | 10 – 100 | 10 – 100 |
| Median Filter | 3 x 3 | – | 10 x 10 | 14 x 14 |
| 3 x 3 Gaussian Filter | Pass | <= 0.5 | <= 2 | <= 1.7 |
| 'S&P' Noise Pollution | <= 0.2 | – | – | 0.1 – 0.8 |
| Gaussian Noise Pollution | – | 0.001 – 0.1 | – | 0.01 – 0.07 |
| Affine Transformation | Up to 5% | – | Up to 20% | – |
| Cropping | 10% | – | 40% | 24% |
| Rows/Columns Removal | 5 rows 17 cols | – | – | 34 rows 17 cols |
| Embedding Image | Gray Images | Gray Images | Gray Images | Gray Images |
| Watermark Length | 16 bits | 8 regions | Signal Sequence | 128 bits |

# AGENDA

- **Introduction & Related Works**
- **Digital Image Watermarking**
  - Robust Feature Detectors
  - Geometrically Invariant Watermarking Methods
  - Experimental Results
- **Digital Audio Watermarking**
  - De-synchronization Resilient Audio Watermarking
  - Experimental Results
- **Conclusions & Future Works**

# Robust Audio Feature Detector




$$f'(x) = \frac{\delta f}{\delta x} = f(x+1) - f(x-1)$$

$$f_s'(x) = G(x) * f'(x)$$

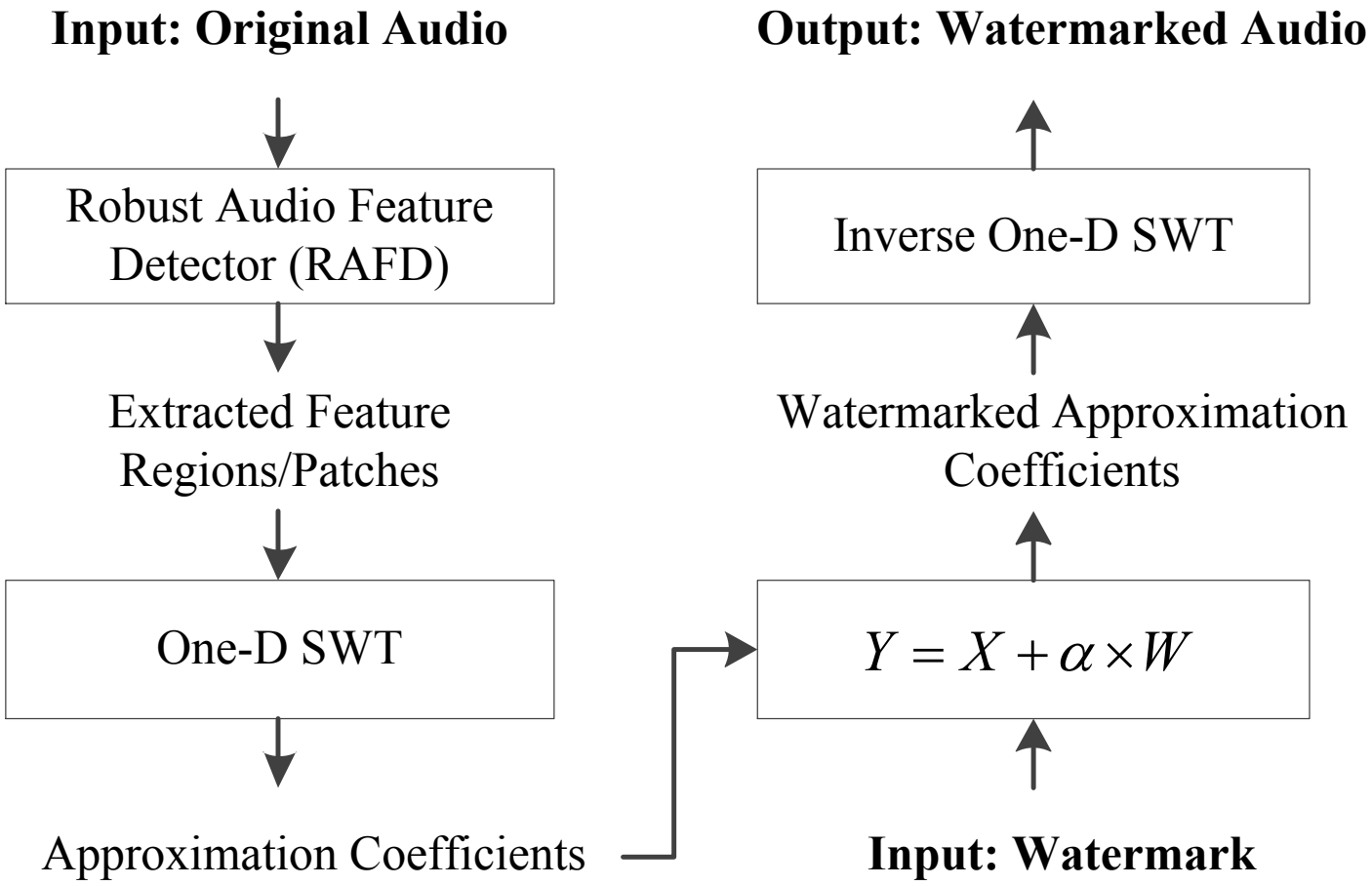
$$R(x) = k(f_s'(x))^2$$

$$((P - M/2) \geq 1) \cap ((P + M/2 - 1) \leq L)$$

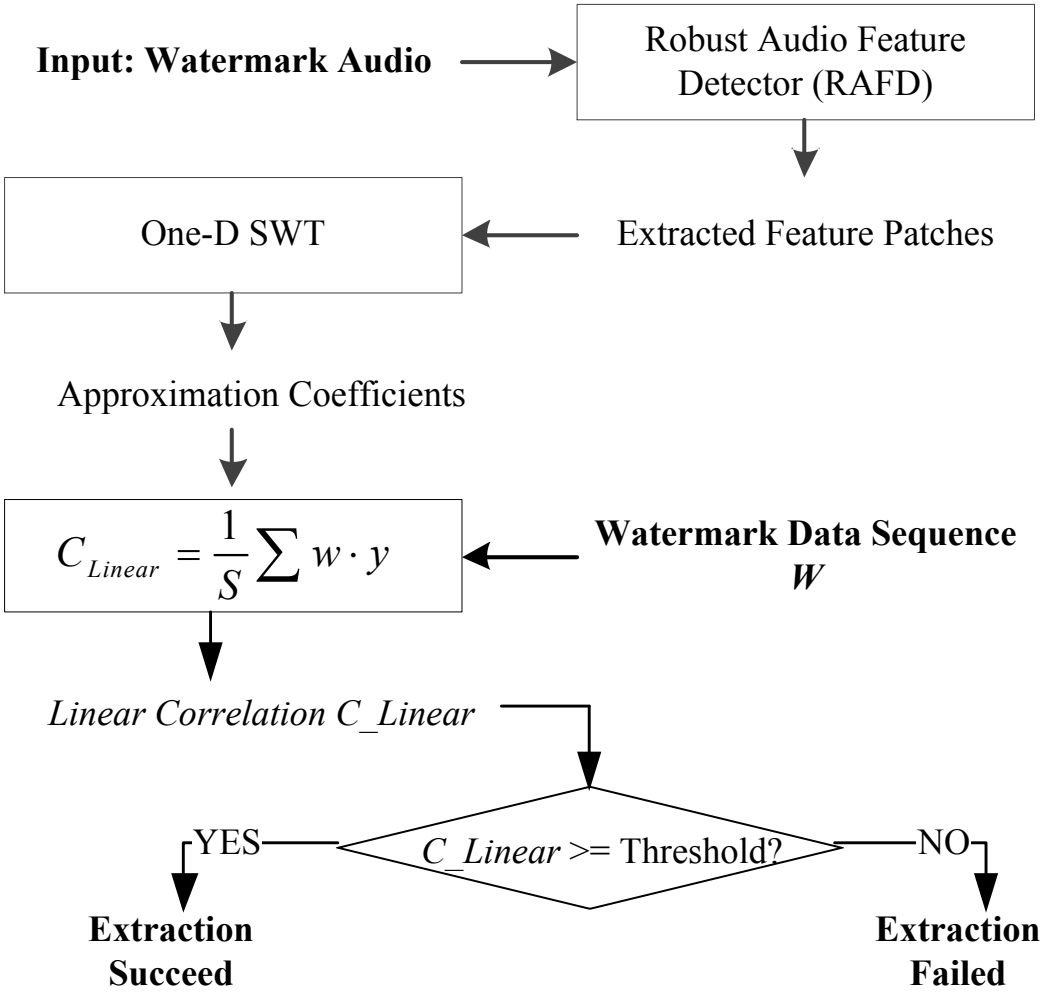
$$|P_{r\_y} - P_{r\_x}| \geq M$$

$$\{[P_1 - M/2, P_1 + M/2 - 1], [P_2 - M/2, P_2 + M/2 - 1], \ldots, [P_N - M/2, P_N + M/2 - 1]\}.$$

# SWT Based Audio Watermark Embedding



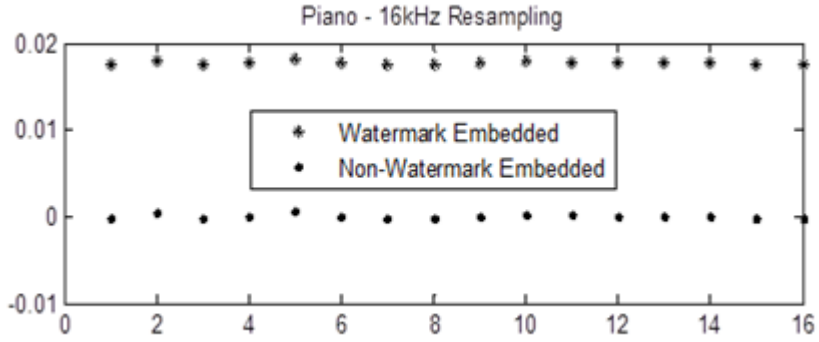
# SWT Based Audio Watermark Extraction



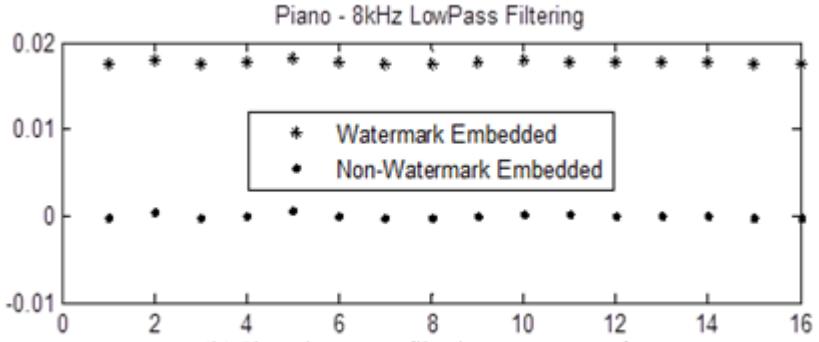
# AGENDA

- **Introduction & Related Works**
- **Digital Image Watermarking**
  - Robust Feature Detectors
  - Geometrically Invariant Watermarking Methods
  - Experimental Results
- **Digital Audio Watermarking**
  - De-synchronization Resilient Audio Watermarking
  - Experimental Results
- **Conclusions & Future Works**

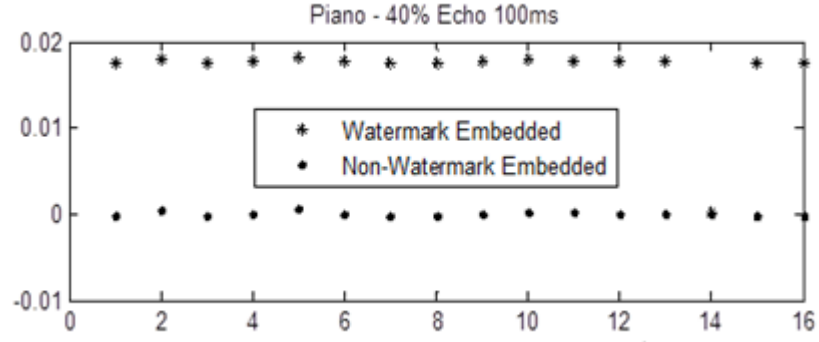
# Audio Watermarking Detection Results



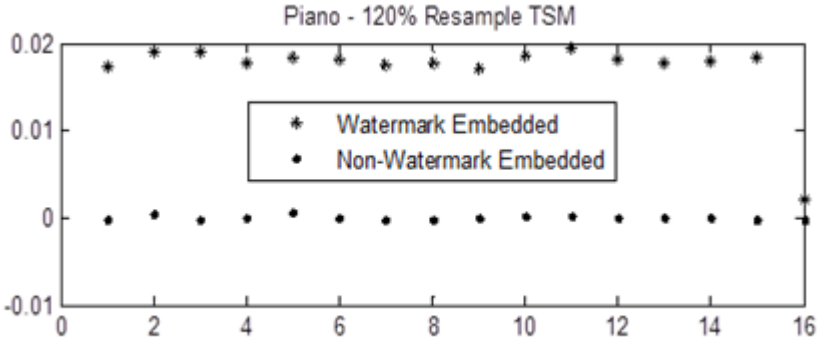

(a) 16kHz resampling, *RCDP = 16/16*



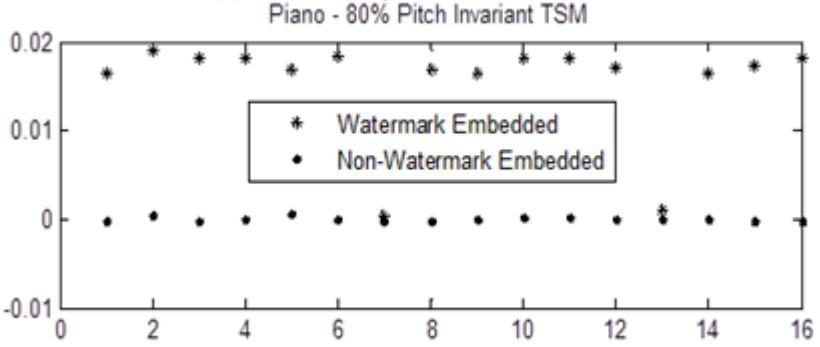

(b) 8kHz low-pass filtering, *RCDP = 16/16*



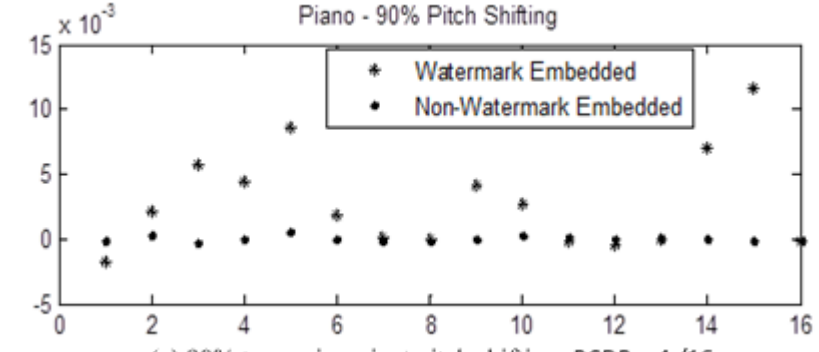

(c) 40% echo & 100ms delay, *RCDP = 15/16*




(a) 120% resample TSM, *RCDP = 15/16*

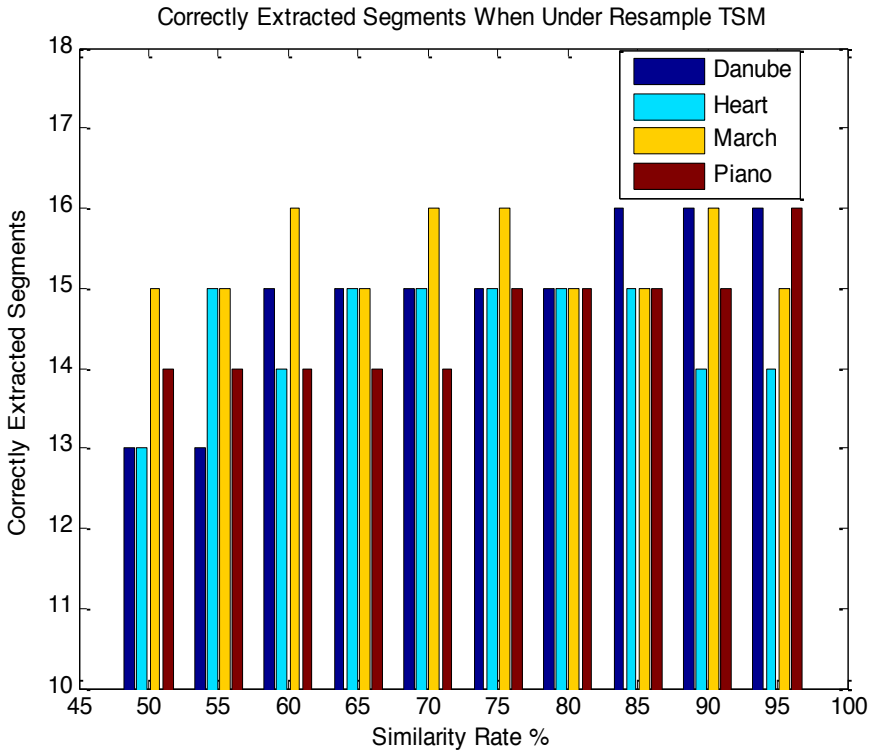


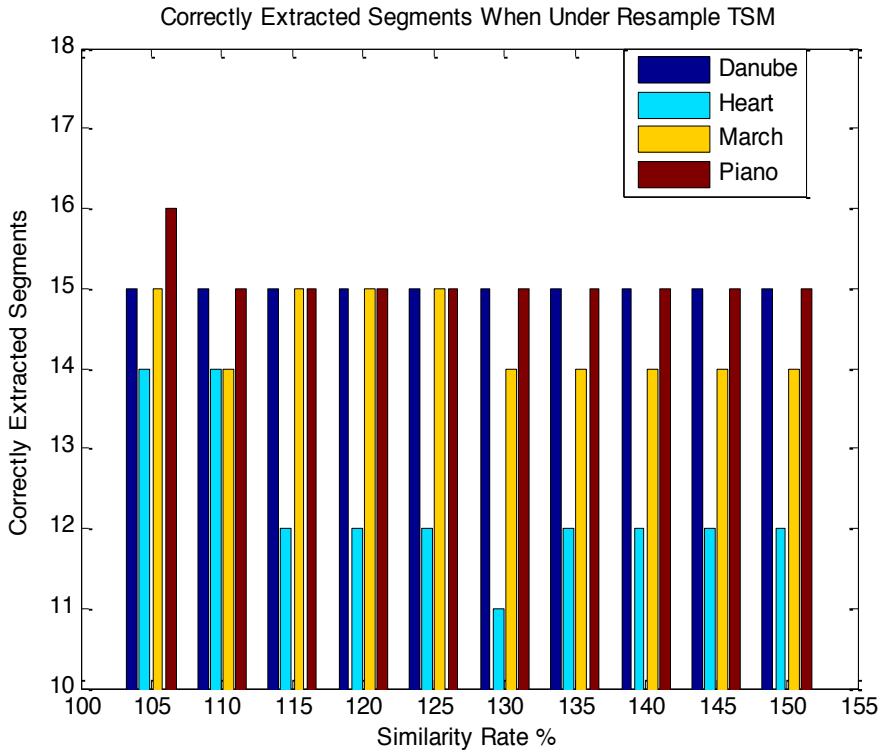

(b) 80% pitch invariant TSM, *RCDP = 14/16*



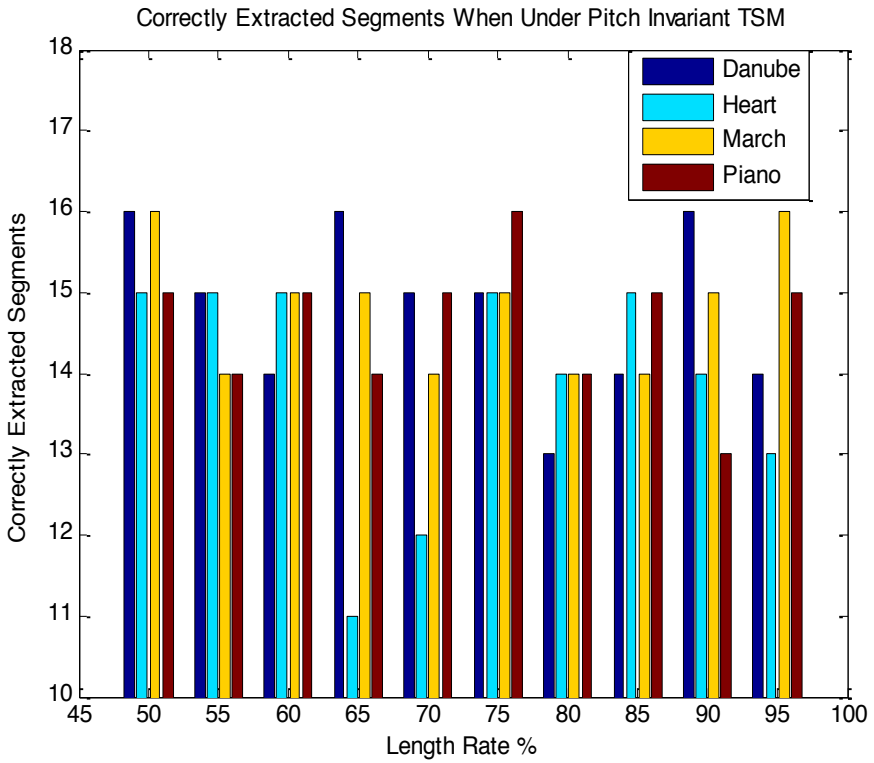

(c) 90% tempo invariant pitch shifting, *RCDP = 4 /16*

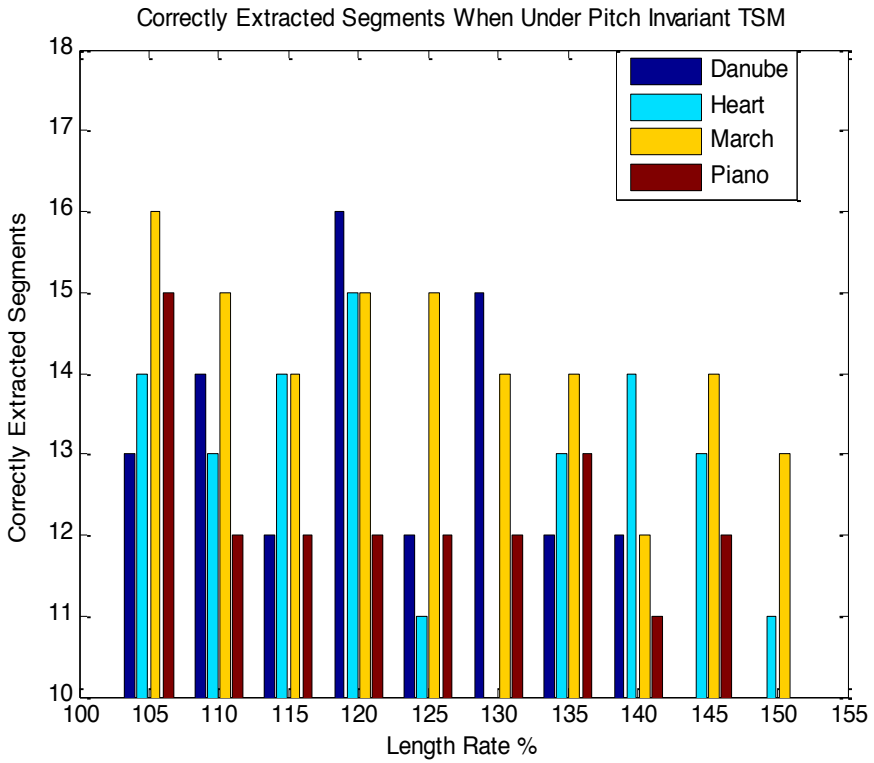
# Correctly Detected Patches Under Resample TSM



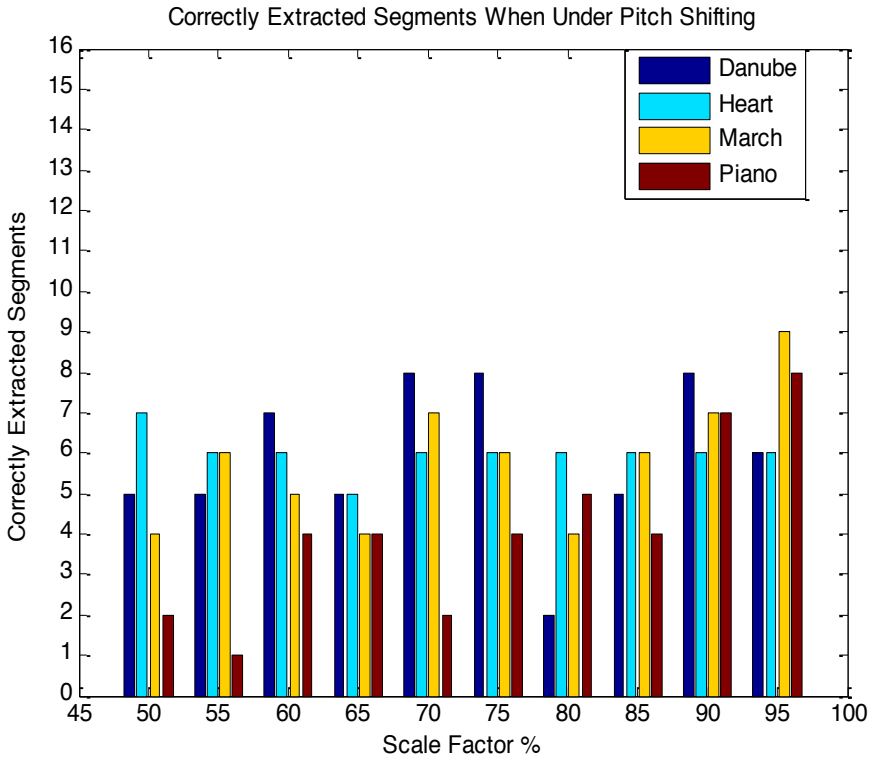



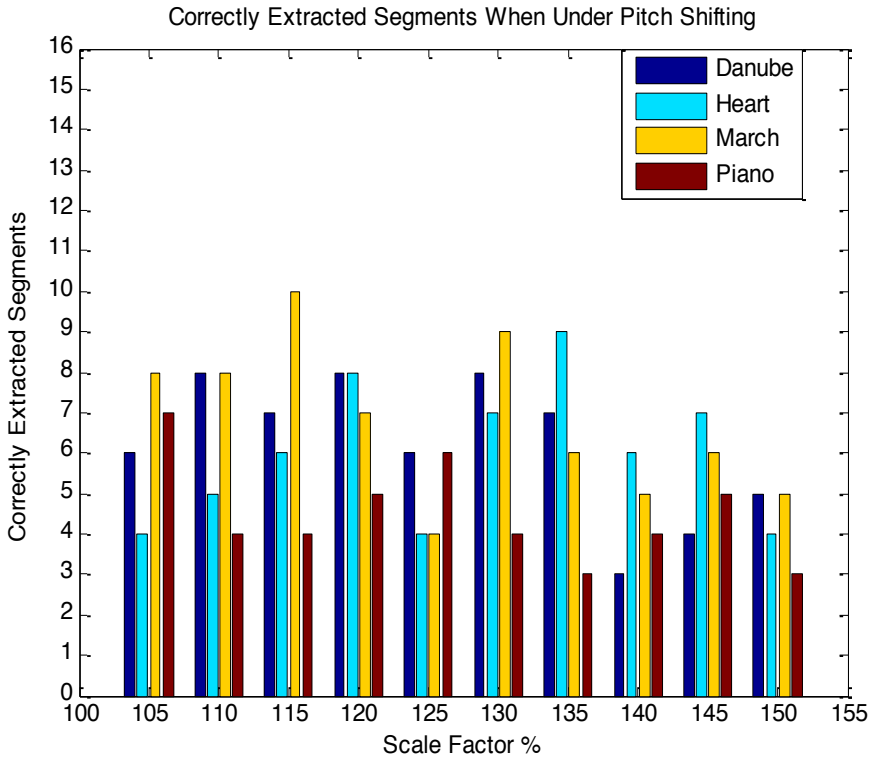
# Correctly Detected Patches Under Pitch Invariant TSM



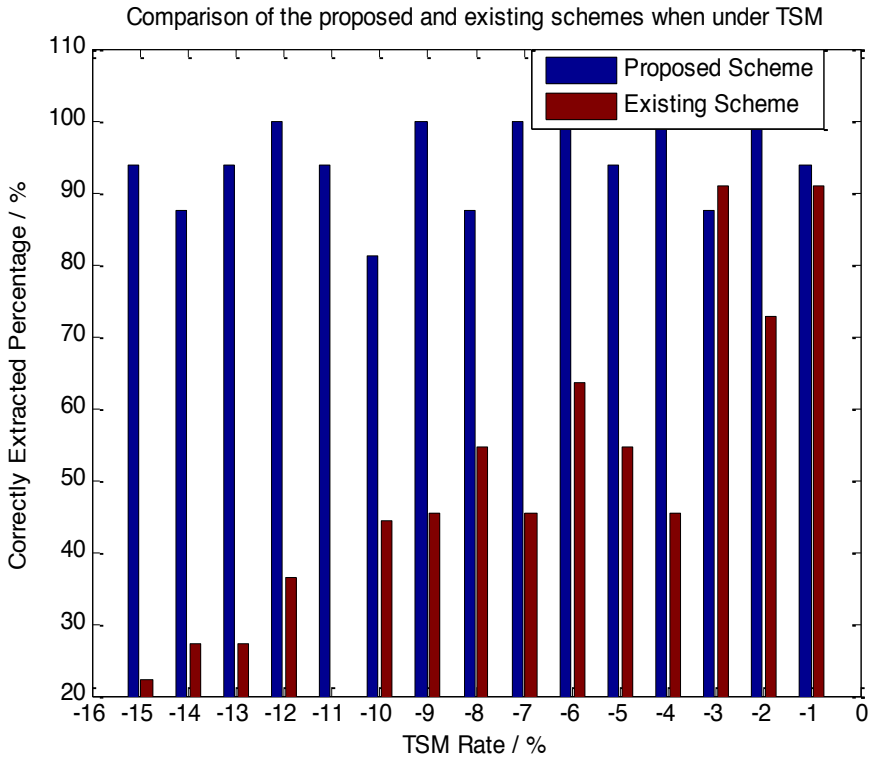



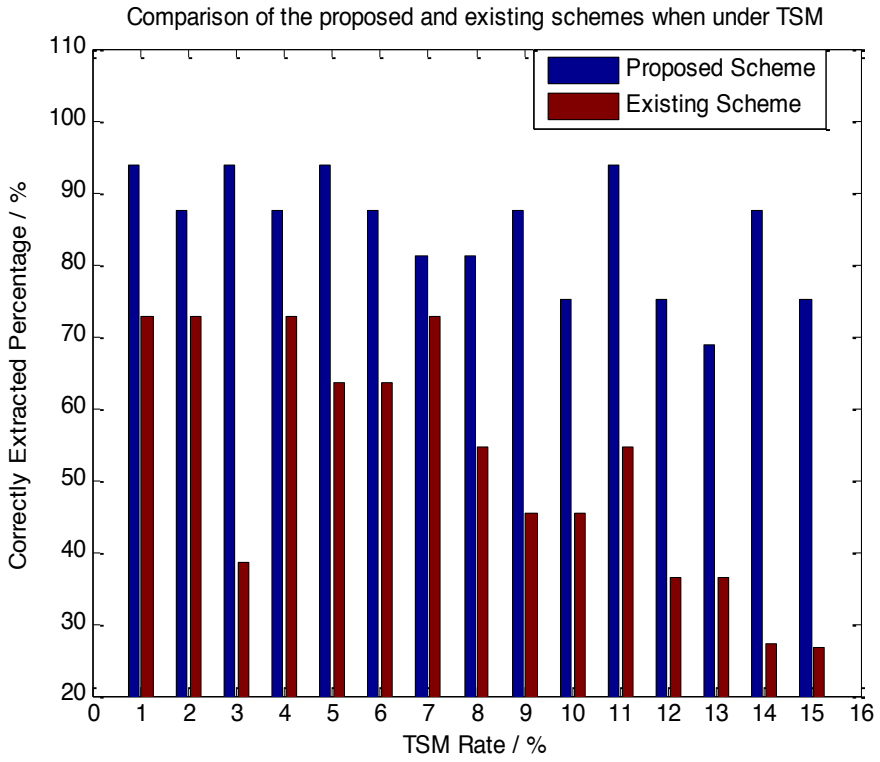
# Correctly Detected Patches Under Pitch Shifting



# Comparison of Extracted Results Under TSM



# Results Comparison For Audio Watermarking

| | Li et al. [74] | Tachibana [68] | Kang et al. [89] | RAFD and SWT Based Watermarking |
|---|---|---|---|---|
| **Resistance to resample TSM (%)** | Fail | 90 ~ 110 | 75 ~ 140 | 50 ~ 150 |
| **Resistance to pitch invariant TSM (%)** | 15 | 10 | 20 | 50 |
| **Resistance to pitch shifting (%)** | Fail | N / A | 20 | 50 |

# AGENDA

- **Introduction & Related Works**
- **Digital Image Watermarking**
  - Robust Feature Detectors
  - Geometrically Invariant Watermarking Methods
  - Experimental Results
- **Digital Audio Watermarking**
  - De-synchronization Resilient Audio Watermarking
  - Experimental Results
- **Conclusions & Future Works**

# CONTRIBUTIONS

- **Feature Detectors:**
  - Edge Based Feature Detector (EBFD)
  - SIFT Based Feature Detector (SBFD)
  - Adaptive Harris Based Detector (AHBD)
- **Watermarking Methods:**
  - Histogram Distribution Based Watermarking
  - Zernike Transform Based Watermarking
- **Digital Audio Watermarking:**
  - Robust Audio Feature Detector (RAFD)
  - SWT Based Watermarking Method

# LIMITATIONS

- **EBFD & Zernike Transform Based Watermarking**
  - Capacity limited
  - Computationally expensive
- **SBFD & Zernike Transform Based Watermarking**
  - Not absolutely blind
  - Computational expenses duplicates accordingly with the increasing of the capacity
- **AHBD & Histogram Distribution Based Watermarking**
  - Cannot robust against some histogram related attacks
    - gamma correction
    - histogram equalization
    - sharpening

# FUTURE WORKS

- **Capacity**
  - EBFD – can extract only one unique feature
  - SBFD – computational expenses duplicates accordingly with the increasing of the capacity
  - Relationship – capacity, computational expenses, and robustness
- **Robustness**
- **Digital Video Watermarking**
  - Apply feature extraction based algorithms into the digital videos to evaluate the performances
  - Propose schemes specially for digital video watermarking

# PUBLICATIONS


1. **C.-M. Pun** and X.-C. Yuan, "Robust Segments Detector for De-Synchronization Resilient Audio Watermarking," *IEEE Transactions on Audio, Speech, and Language Processing*, 21(11), pp. 2412 – 2424, 2013.

2. X.-C. Yuan, **C.-M. Pun** and C. L. Philip Chen, "Geometric invariant watermarking by local Zernike moments of binary image patches," *Signal Processing*, 93(7), pp. 2087–2095, 2013.

# THANK YOU !

# Q & A