MULTI-LEVEL FEATURE EXTRACTION AND ADAPTIVE MATCHING FOR COPY-MOVE FORGERY DETECTION

CHI-MAN PUN

UNIVERSITY OF MACAU

AGENDA

- **Introduction**
- **Related Works**
- **Proposed Methodology**
  - Multi-Level Feature Extraction
  - Adaptive Matching Methods for Copy-Move Forgery Detection
  - Experimental Results
- **Conclusions & Future Works**

AGENDA

- **Introduction**
- **Related Works**
- **Proposed Methodology**
  - Multi-Level Feature Extraction
  - Adaptive Matching Methods for Copy-Move Forgery Detection
  - Experimental Results
- **Conclusions & Future Works**

INTRODUCTION

- **Digital Image Forgery Categories**
  - *Image Retouching*
  - *Splicing Forgery*
  - *Copy-Move Forgery*

INTRODUCTION

Image Retouching

INTRODUCTION

Splicing Forgery

INTRODUCTION

Copy-Move Forgery

INTRODUCTION

- **Digital Image Forgery Categories**
  - *Image Retouching*
  - *Splicing Forgery*
  - *Copy-Move Forgery*
- **Digital Image Forgery Detection Categories**
  - *Splicing Forgery Detection*
  - *Copy-Move Forgery Detection*

AGENDA

- **Introduction**
- **Related Works**
- **Proposed Methodology**
  - Multi-Level Feature Extraction
  - Adaptive Matching Methods for Copy-Move Forgery Detection
  - Experimental Results
- **Conclusions & Future Works**

RELATED WORKS

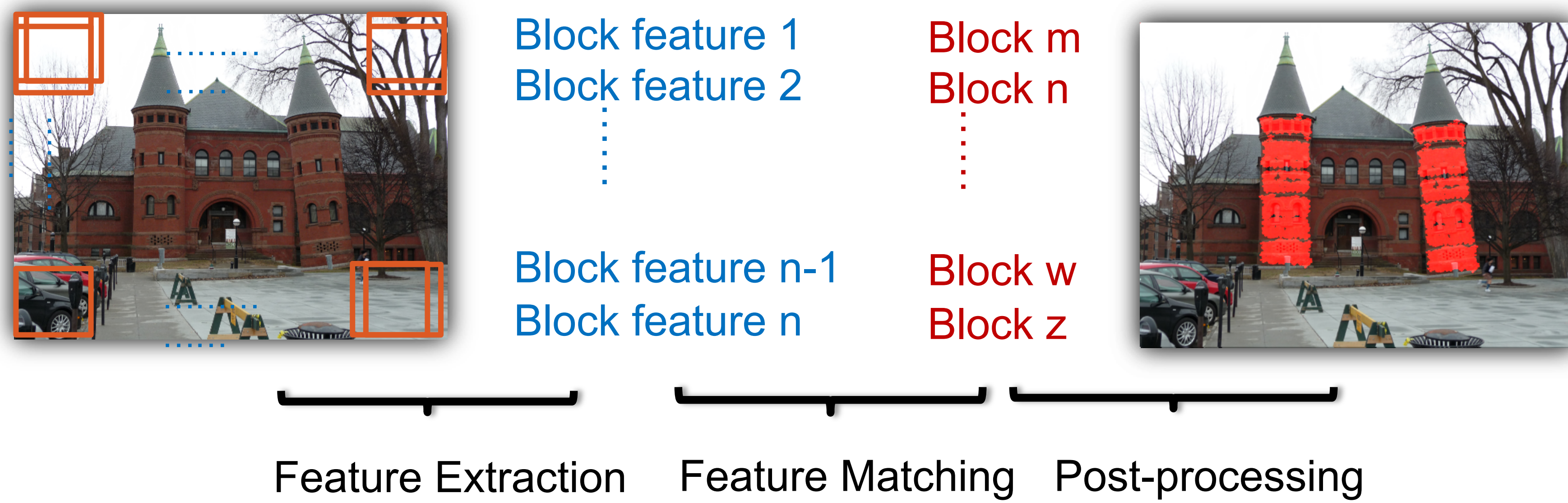
- **Copy-Move Forgery Detection Categories**
  - *Block-based Detection Methods*
  - *Keypoint-based Detection Methods*

Pre-processing → Feature Extraction → Feature Matching → Post-processing

The common process pipeline

RELATED WORKS

- **Block-based Detection Methods**



Extracted Feature: DCT, DWT, Zernike, PCT, PCA, HOG and so on.

RELATED WORKS

- **Block-based Detection Methods**

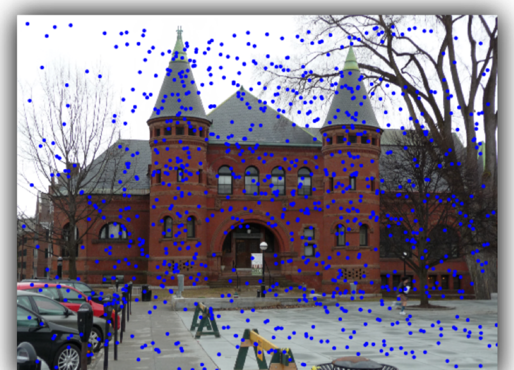
➢ can detect the forgery regions in any size image

➢ computational complexity is high

➢ cannot deal with the significant geometrical transformation of the forgery regions

RELATED WORKS

- **Keypoint-based Detection Methods**

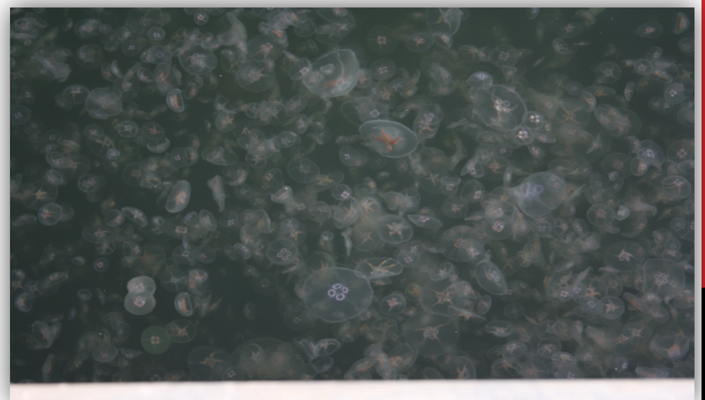


Keypoint Descriptor 1 Keypoint m
Keypoint Descriptor 2 Keypoint n
⋮ ⋮
Keypoint Descriptor n Keypoint z

Feature Extraction Feature Matching Post-processing

Extracted Keypoint: SIFT, SURF, BRISK, DAISY and so on.

RELATED WORKS



- **Keypoint-based Detection Methods**

- computational complexity is low
- can deal with the significant geometrical transformation of the forgery regions
- recall rate is low
- may fail when the host image is small or low lightness or low contrast

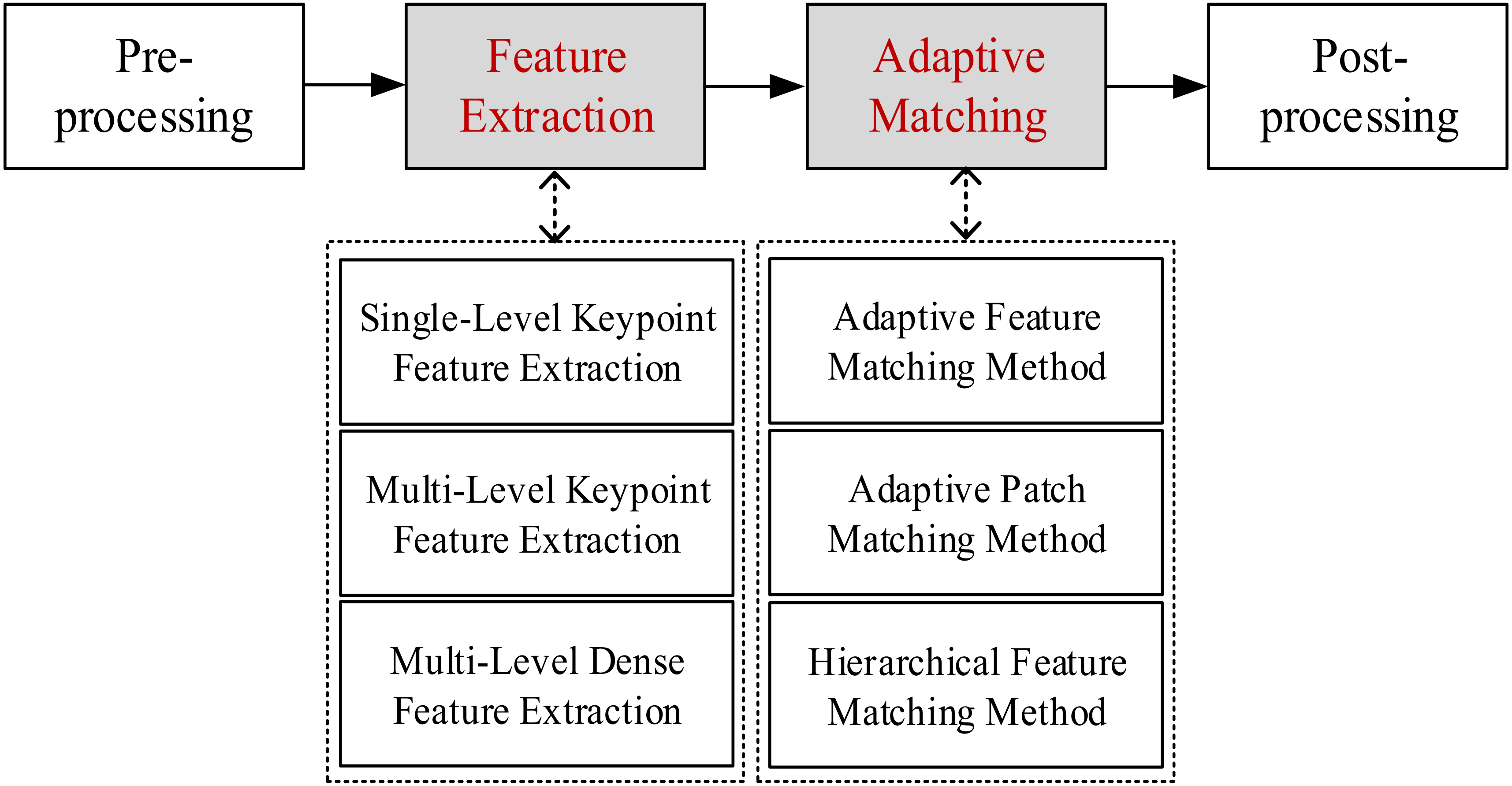
CHALLENGES

- Any image size
- Computational complexity
- Robustness (e.g geometric distortion, image degradation)

AGENDA

- **Introduction**
- **Related Works**
- **Proposed Methodology**
  - Multi-Level Feature Extraction
  - Adaptive Matching Methods for Copy-Move Forgery Detection
  - Experimental Results
- **Conclusions & Future Works**

PROPOSED METHODOLOGY



AGENDA

- **Introduction**
- **Related Works**
- **Proposed Methodology**
  - Multi-Level Feature Extraction
  - Adaptive Matching Methods for Copy-Move Forgery Detection
  - Experimental Results
- **Conclusions & Future Works**

Proposed Feature Detectors

- **Single-Level Keypoint Feature Extraction**
  - extract the keypoints from single-level segmentation.
- **Multi-Level Keypoint Feature Extraction**
  - extract the keypoints from multi-level segmentation.
- **Multi-Level Dense Feature Extraction**
  - extract a multi-level feature from each pixel.

SINGLE-LEVEL KEYPOINT FEATURE EXTRACTION

Challenges:

- Block-based detection method
  - ➢ overlapping block → long running time
  - ➢ cannot deal with significant geometrical transformation of the forgery regions
- Keypoint-based detection method
  - ➢ the recall rate is low
  - ➢ may fail in some cases

Research goals:

- ➢ non-overlapping block (superpixel)
- ➢ extract keypoint as block feature

SINGLE-LEVEL KEYPOINT FEATURE EXTRACTION

How to determine the initial size of the superpixel ?

- the image is smooth, the initial size can be set to be large.
- the image has more details, the initial size can be set to be small.

four-level 'Haar' wavelet

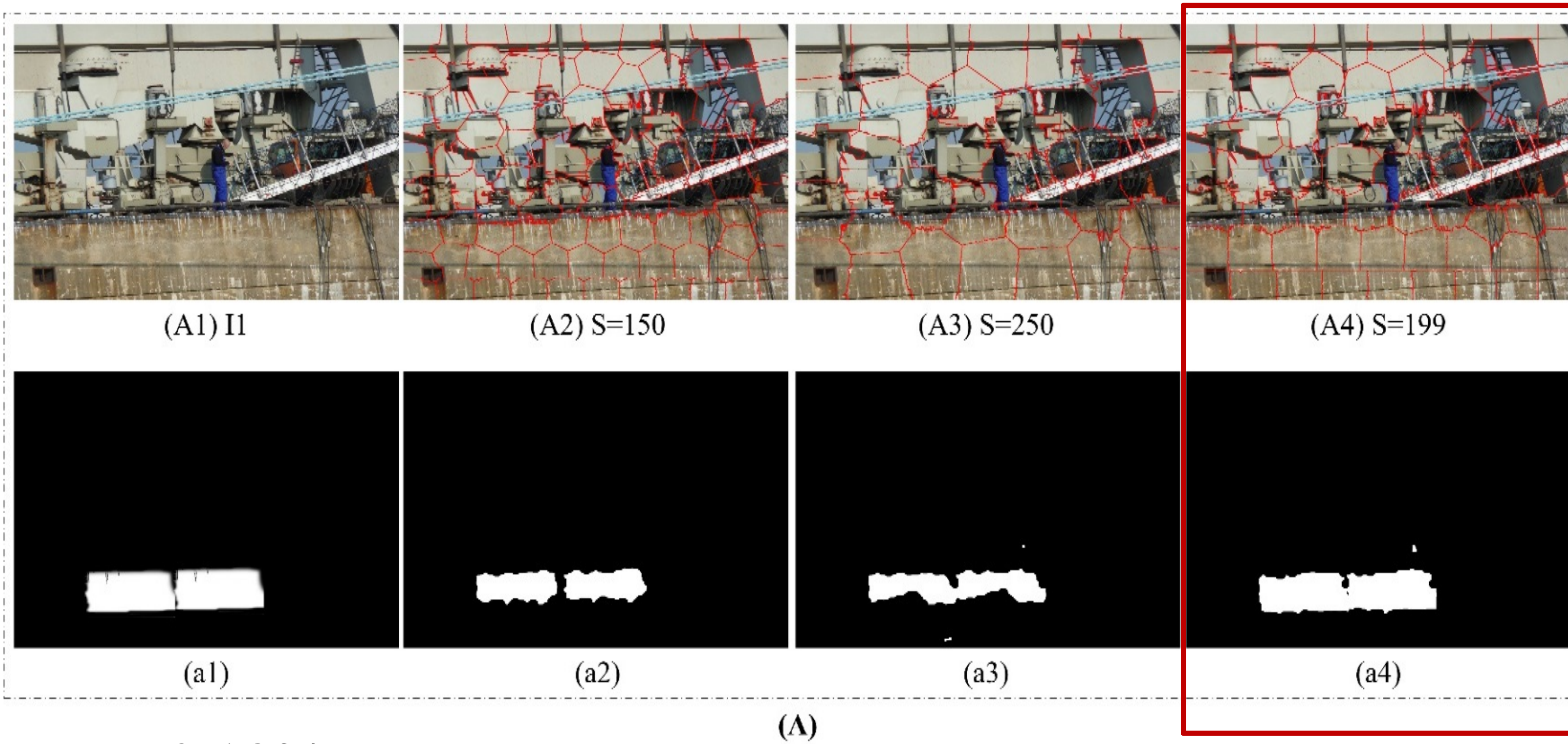
$$F_L = \sum |CA_4|$$

$$F_H = \sum_i \left(\sum |CD_i| + \sum |CH_i| + \sum |CV_i| \right), i = 1, \quad 2, \quad \ldots, \quad 4$$

$$P_F = \frac{F_L}{F_L + F_H} \cdot 100\%$$

$$S = \begin{cases} \sqrt{0.02 \times M \times N} & P_F > 50\% \\ \sqrt{0.01 \times M \times N} & P_F < 50\% \end{cases}$$

SINGLE-LEVEL KEYPOINT FEATURE EXTRACTION

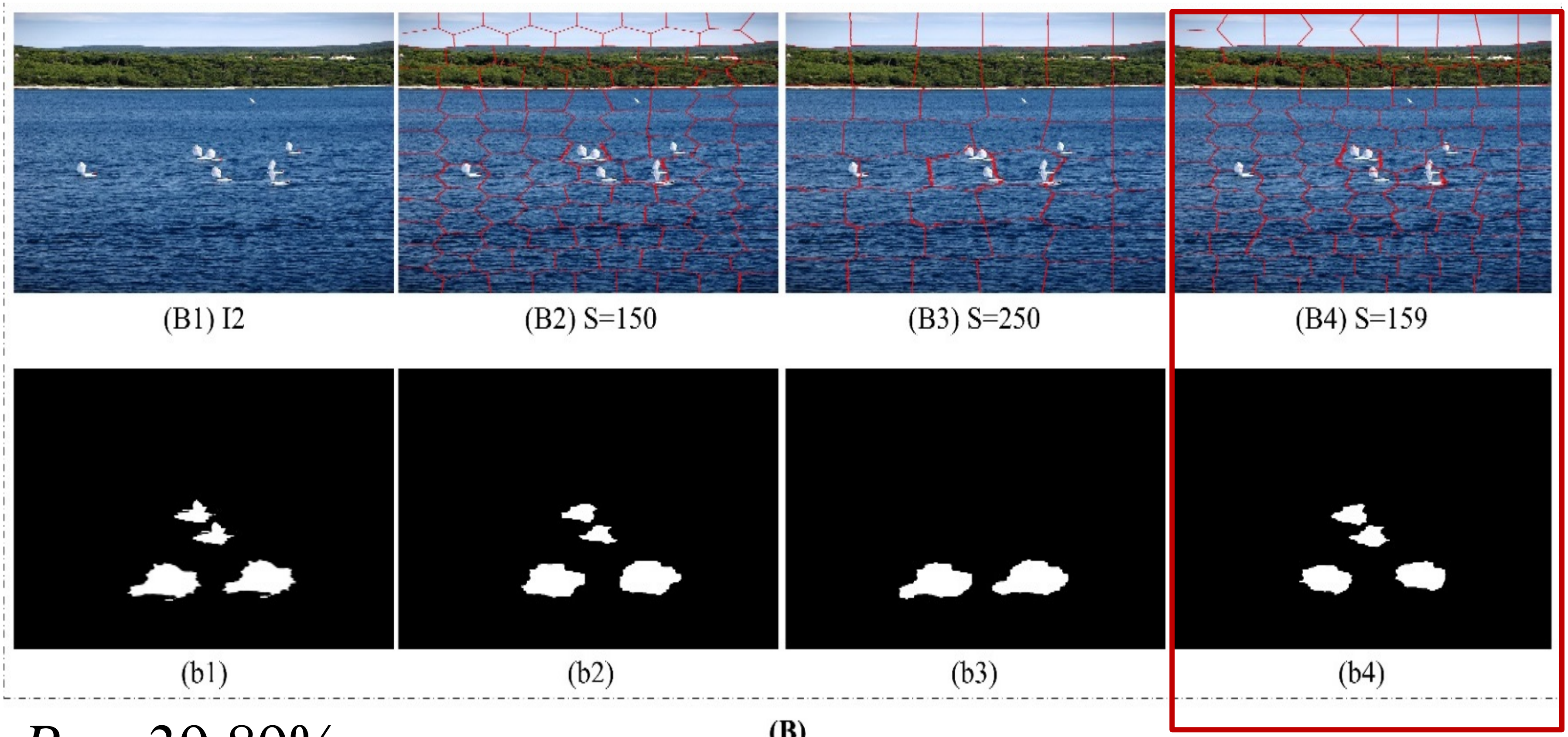


$$P_F = 50.19\%$$

$$M \times N = 1632 \times 1224$$

$$S = 199$$

SINGLE-LEVEL KEYPOINT FEATURE EXTRACTION

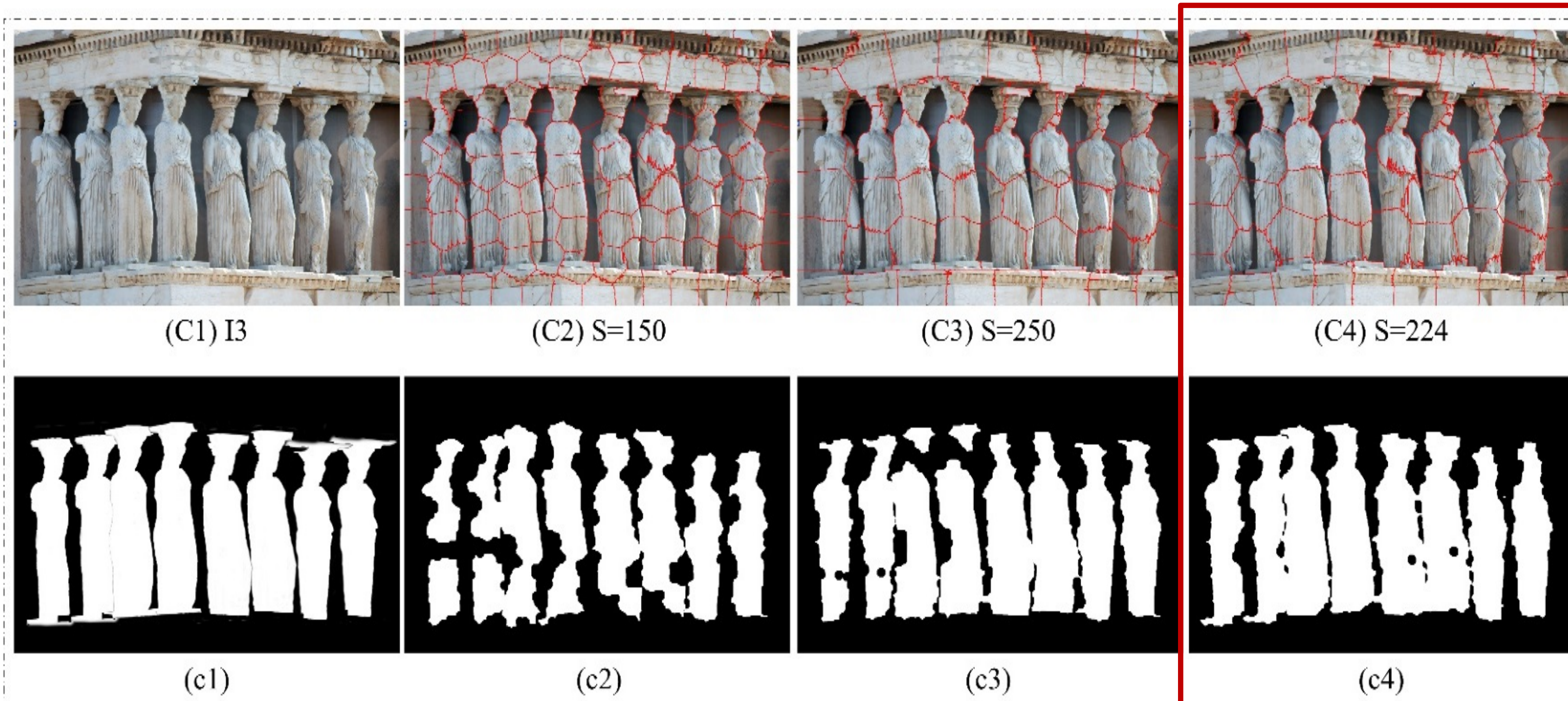


$P_F = 39.89\%$

$M \times N = 1306 \times 1950$

$S = 159$

SINGLE-LEVEL KEYPOINT FEATURE EXTRACTION



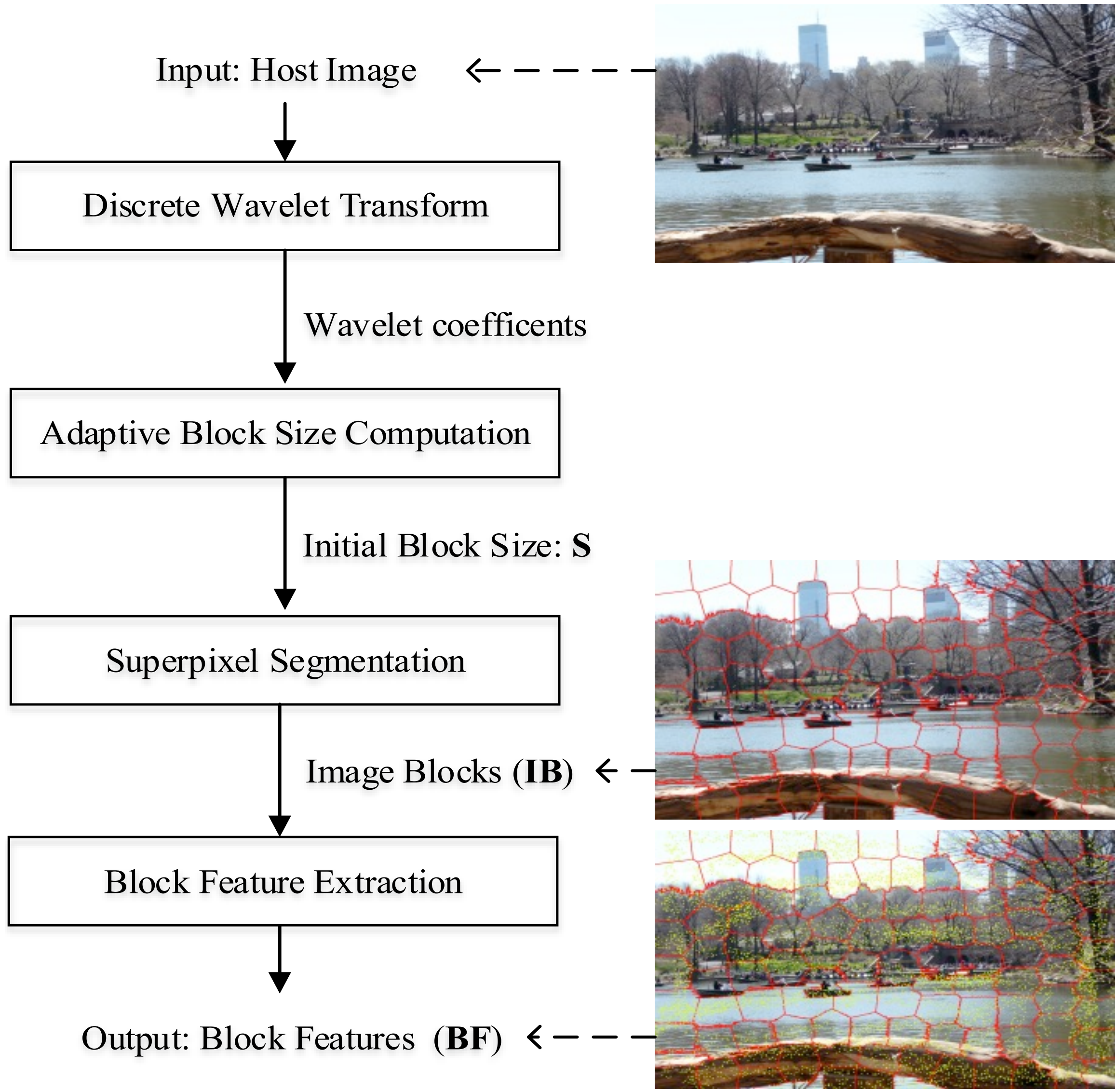
(C)

$$P_F = 59.92\%$$

$$M \times N = 1936 \times 1296$$

$$S = 224$$

SINGLE-LEVEL KEYPOINT FEATURE EXTRACTION



MULTI-LEVEL KEYPOINT FEATURE EXTRACTION

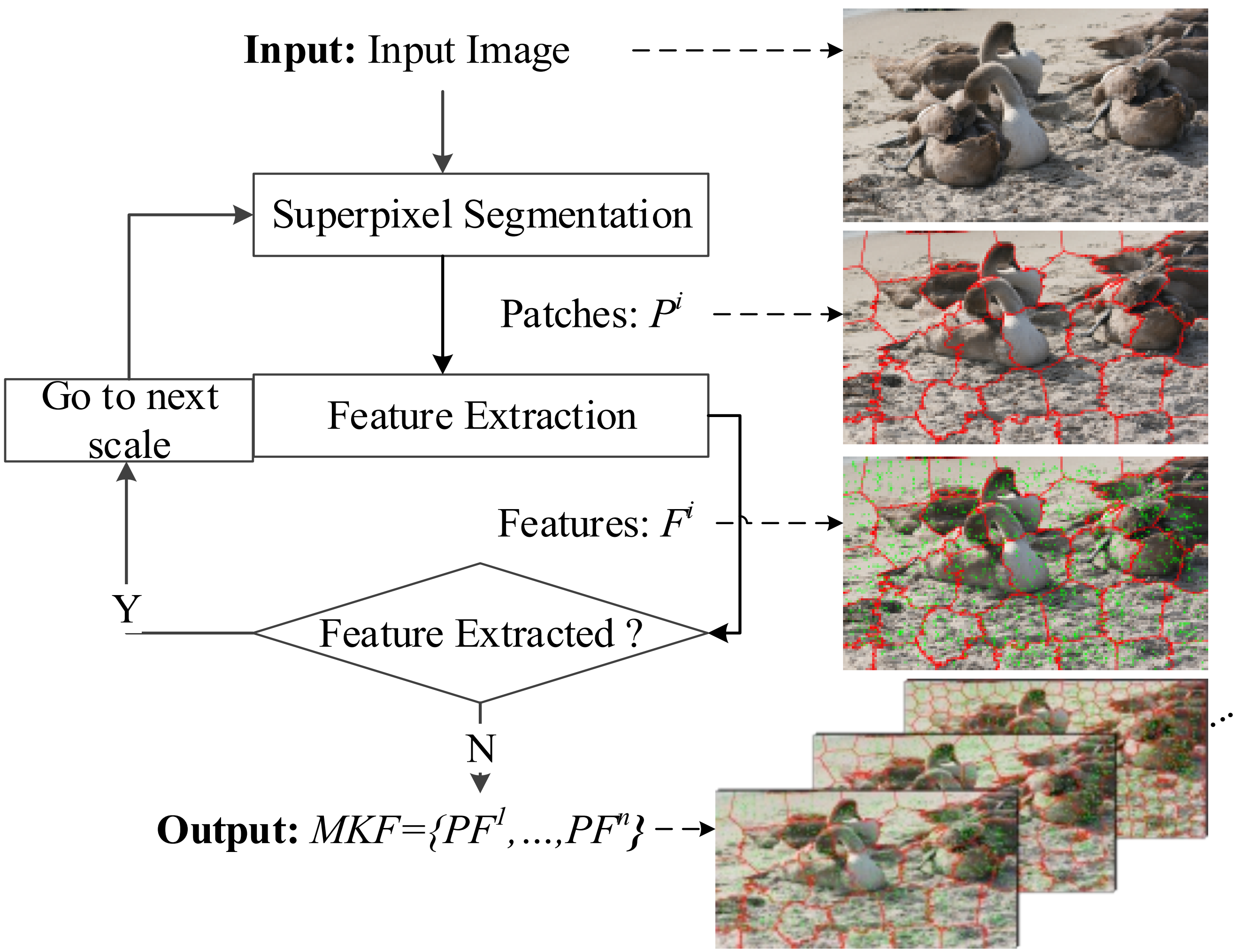
Challenges:

- multiple copy-move forgery regions
- the forgery regions are of different sizes
- the forgery regions may contain both smoothed and detailed textures

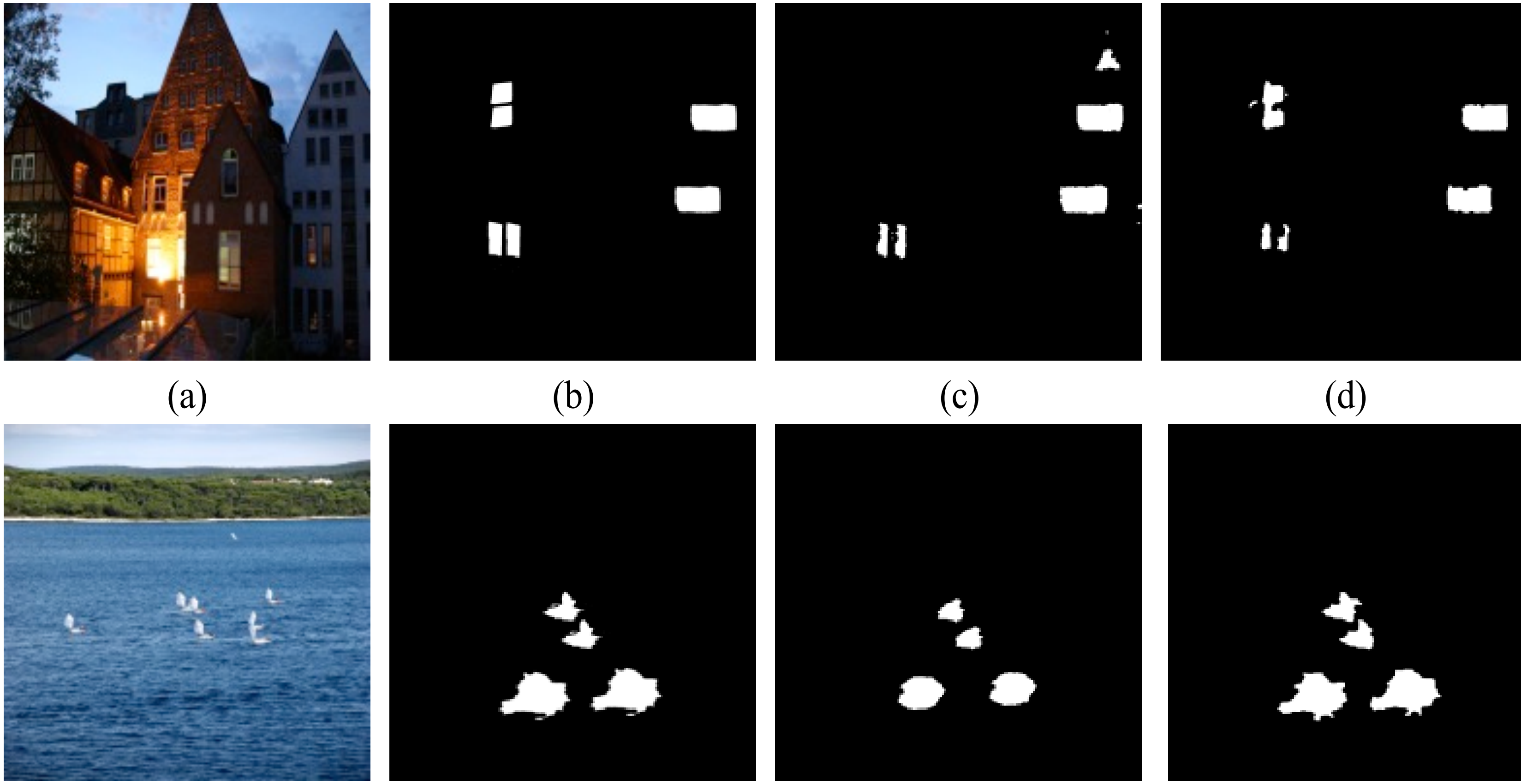
Research goals:

- multi-level non-overlapping block
- extract keypoint as block feature

MULTI-LEVEL KEYPOINT FEATURE EXTRACTION



MULTI-LEVEL KEYPOINT FEATURE EXTRACTION



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure: Demonstration of detection results of single-level and multi-leve keypoint feature extraction. **1st column**: the forgery images; **2nd column**: ground truth images; **3rd column**: the detected regions of single-level keypoint feature extraction; and **4th column**: the detected regions of multi-level keypoint feature extraction.

MULTI-LEVEL DENSE FEATURE EXTRACTION

Challenges:

- the host image is small (web image)
- keypoint extraction is highly related to the resolution of the images
- the existing block features also have bad performance

Research goals:

- combined feature
- dense feature (extracted from each pixel)

MULTI-LEVEL DENSE FEATURE EXTRACTION

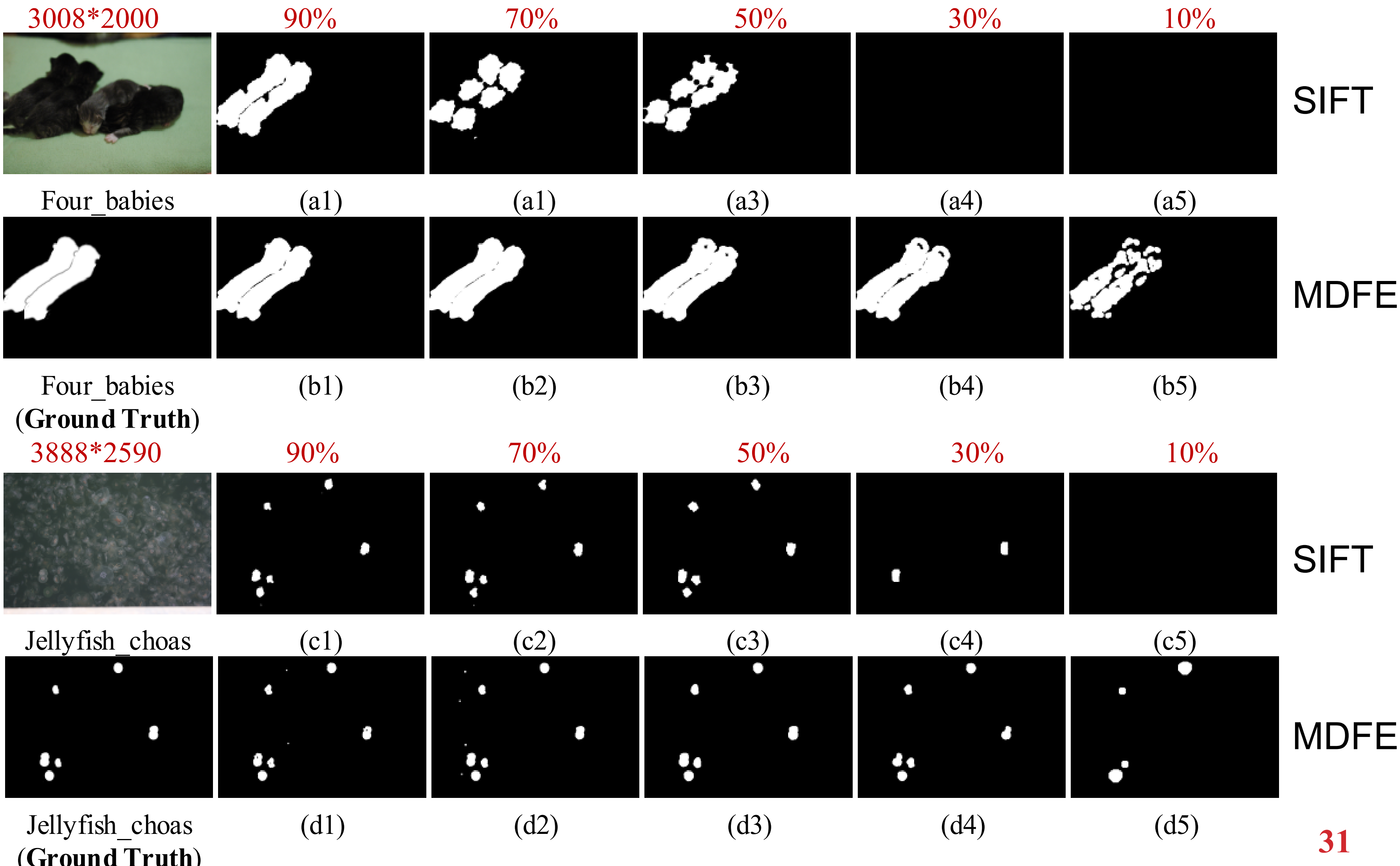
$$W_3 \begin{bmatrix}1&1&1\\1&0&1\\1&1&1\end{bmatrix} \quad W_5 \begin{bmatrix}1&1&1&1&1\\1&0&0&0&1\\1&0&0&0&1\\1&0&0&0&1\\1&1&1&1&1\end{bmatrix} \quad \cdots\cdots \quad W_{2L+1} \begin{bmatrix}1&1&1&1&1&1&1&1&1\\1&0&\cdots&&&&&0&1\\1&\vdots&\ddots&&&&&\vdots&1\\1&&&&&&&&1\\1&&&&\ddots&&&&1\\1&&&&&&&&1\\1&\vdots&&&&&\ddots&\vdots&1\\1&0&\cdots&&&&&0&1\\1&1&1&1&1&1&1&1&1\end{bmatrix} \quad W_{2L+1} \begin{bmatrix}1&1&1&1&1&1&1&1&1\\1&1&\cdots&&&&&1&1\\1&\vdots&\ddots&&&&&\vdots&1\\1&&&&&&&&1\\1&&&&\ddots&&&&1\\1&&&&&&&&1\\1&\vdots&&&&&\ddots&\vdots&1\\1&1&\cdots&&&&&1&1\\1&1&1&1&1&1&1&1&1\end{bmatrix}$$

$$MLDF\_CT(x,y) = \{CT_3^Y(x,y), CT_3^{Cb}(x,y), CT_3^{Cr}(x,y), CT_3^Y(x,y),,\ldots, CT_{2L+1}^{Cr}(x,y)\}$$

$$MLDF\_IM(x,y) = \left\{IM_{0,0}(x,y),\ldots,IM_{0,O_{\max}}(x,y), IM_{1,0}(x,y),\ldots,IM_{1,O_{\max}}(x,y),\ldots,IM_{O_{\max},O_{\max}}(x,y)\right\}$$

$$MLDF(x,y) = \left\{MLDF\_CT(x,y), MLDF\_IM(x,y)\right\}$$

DETECTION RESULTS UNDER DOWN-SAMPLING



AGENDA

- **Introduction**
- **Related Works**
- **Proposed Methodology**
  - Multi-Level Feature Extraction
  - Adaptive Matching Methods for Copy-Move Forgery Detection
  - Experimental Results
- **Conclusions & Future Works**

ADAPTIVE THRESHOLD CALCULATION

- match every feature with all other features to find the matched feature pairs.
- calculate the shift vector of each matched feature pairs.
- let H(A) be the number of feature pairs satisfying the same shift vector A.
- remove the matched feature pairs whose H(A) below threshold **T** .

Challenges:

- the initial setting threshold **T**

Research goals:

- the adaptive determined threshold **T**

ADAPTIVE THRESHOLD CALCULATION

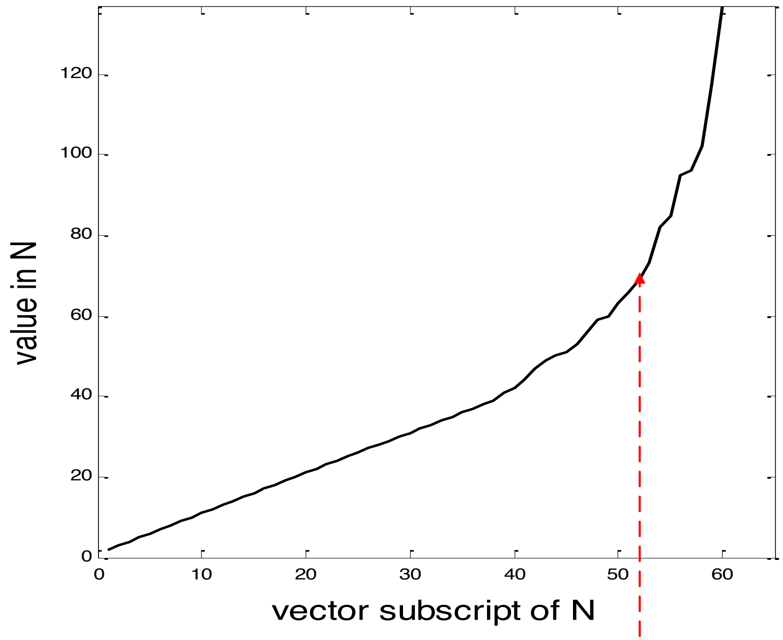
- a sequence $N = \{n_1 \quad n_2 \quad \dots \quad n_t\}$ is sorted in ascending order
- calculate the first derivative $\nabla(N)$ and the second derivative $\nabla^2(N)$
- calculate the mean value $\overline{\nabla(N)}$
- select the minimum number whose second derivative meet

$$\nabla^2(N) > \overline{\nabla(N)}$$

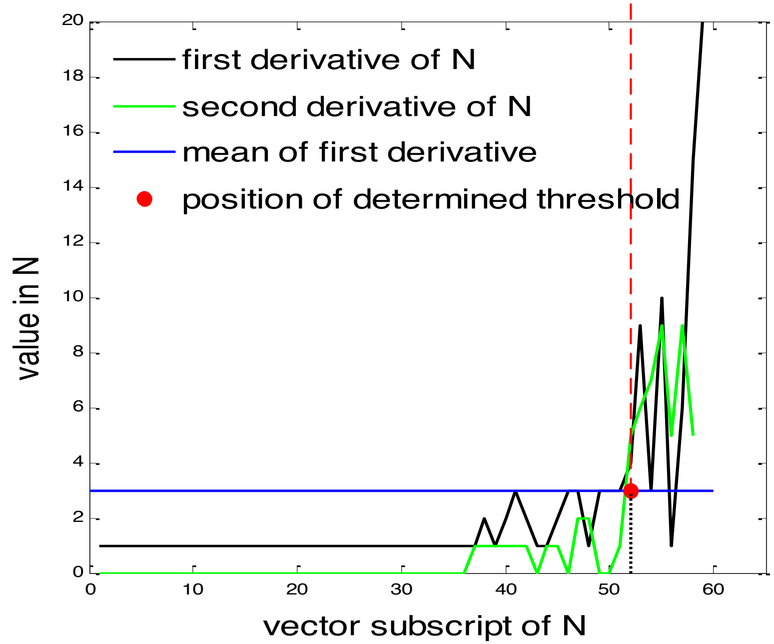
as the threshold **T**

copy-move forgery image

(a1)



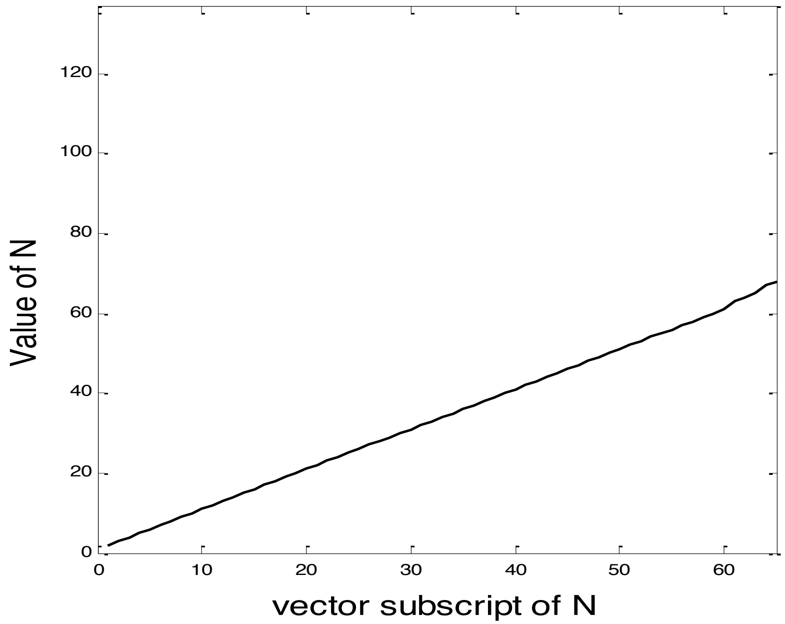
(a2)



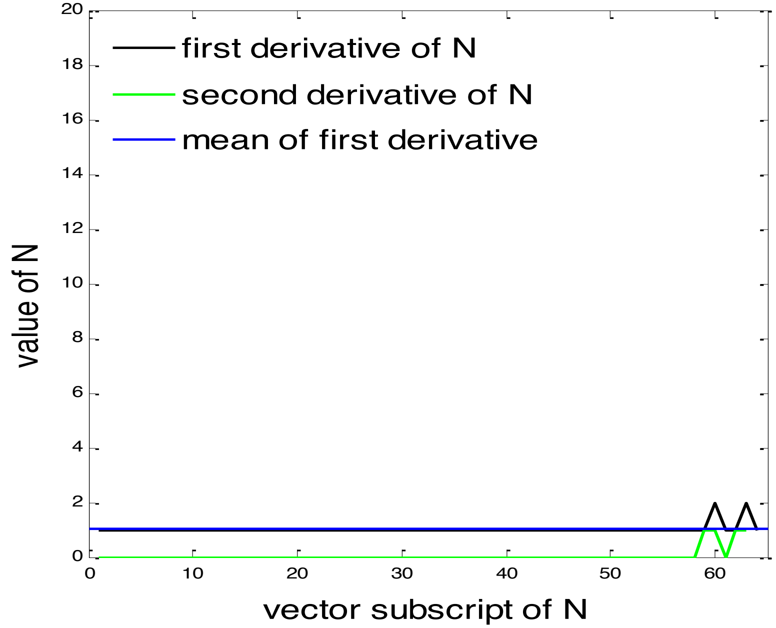
(a3)

original image

(b1)



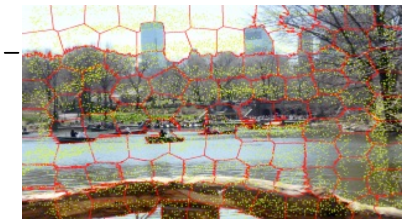
(b2)



(b3)

Adaptive Feature Matching Method (based on single-level keypoint feature)

Input: Block Features (**BF**)



Correlation Coefficient Map Generation

Correlation Coefficient Map (***CC***)

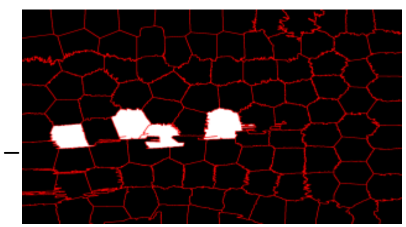
	1	2	3	4	5	6	7	8	9	10	...
1	0	4	2	2	2	4	8	0	6	4	...
2	0	0	4	5	2	2	6	2	2	1	...
3	0	0	0	2	0	0	6	4	1	1	...
4	0	0	0	0	2	9	6	3	2	2	...
5	0	0	0	0	0	7	4	1	1	2	...
6	0	0	0	0	0	0	1	3	1	2	...
7	0	0	0	0	0	0	0	0	0	1	...
8	0	0	0	0	0	0	0	0	1	0	...
9	0	0	0	0	0	0	0	0	0	5	...
10	0	0	0	0	0	0	0	0	0	0	...
...	...	...	...	...	...	...	...	...	...	...	...

Block Matching Threshold Calculation

Block Matching Threshold TR_B

Matched Blocks Location

Matched Blocks (***MB***)



Matched Feature Points Labelling

Labeled Feature Points (**LFP**)

Labeled Feature Points (**LFP**)

Small Superpixels Replacement

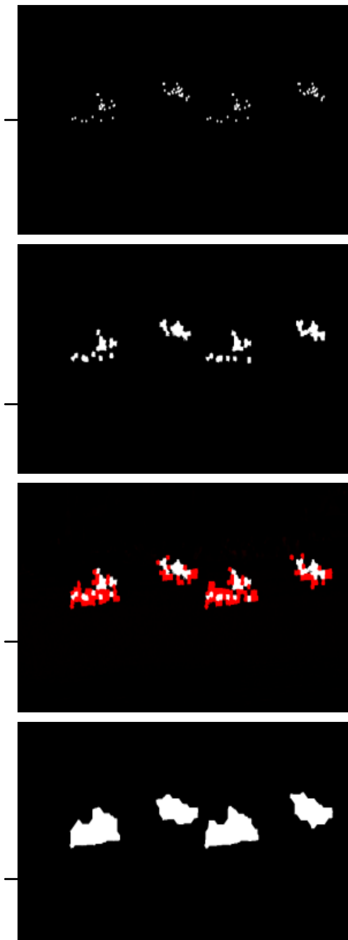
Suspected Regions (**SR**)

Local Color Feature Extraction

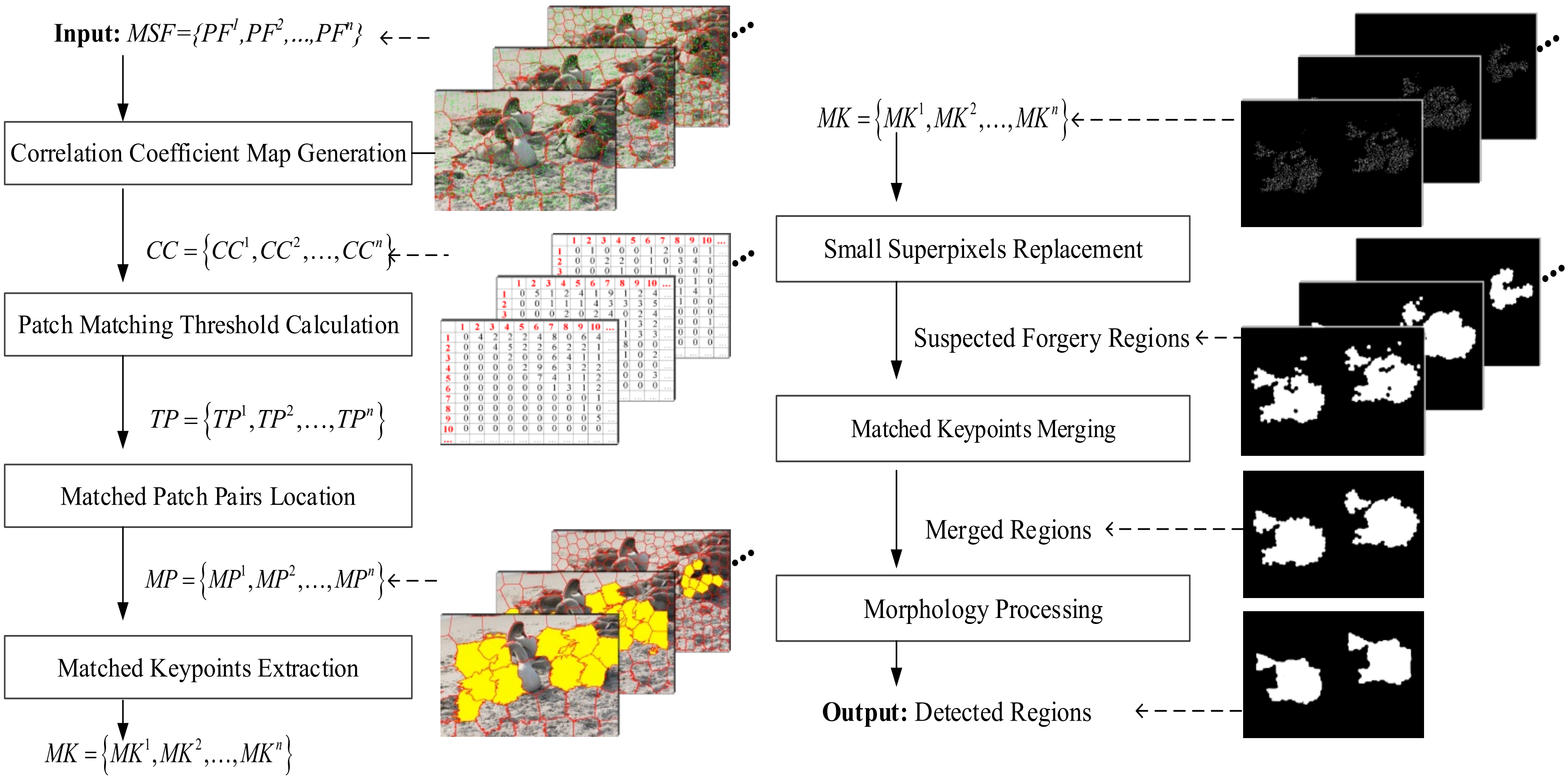
Merged Regions (**MR**)

Morphology Processing

Output: Detected Forgery Regions



Adaptive Patch Matching Method (based on multi-level keypoint feature)



Hierarchical Feature Matching Method (based on multi-level dense feature)

Step1: color texture-based filtering

$$
\begin{array}{l}
MLDF\_CT(1) \\
MLDF\_CT(2) \\
MLDF\_CT(3) \\
MLDF\_CT(4) \\
\vdots \\
\vdots \\
MLDF\_CT(M \times N)
\end{array}
\xrightarrow{\text{lexicographical sorting method}}
\begin{array}{l}
MLDF\_CT(idx\_1) \\
MLDF\_CT(idx\_2) \\
MLDF\_CT(idx\_3) \\
MLDF\_CT(idx\_4) \\
\vdots \\
\vdots \\
MLDF\_CT(idx\_\times N)
\end{array}
$$

idx_1	*idx_2*	···	*idx_i-R*	···	*idx_i-1*	*idx_i*	*idx_i+1*	···	*idx_i+R*	···	*idx_M×N-1*	*idx_M×N*

R R

the selected neighbor pixels set of pixel idx_i

Hierarchical Feature Matching Method (based on multi-level dense feature)

Step2: invariant moment-based matching

$$MLDF\_IM(idx\_i) \leftrightarrow \boxed{\begin{array}{c} MLDF\_IM(idx\_i-R) \\ \vdots \\ MLDF\_IM(idx\_i-1) \\ MLDF\_IM(idx\_i+1) \\ \vdots \\ MLDF\_IM(idx\_i+R) \end{array}} \longrightarrow$$

Matched Pixel

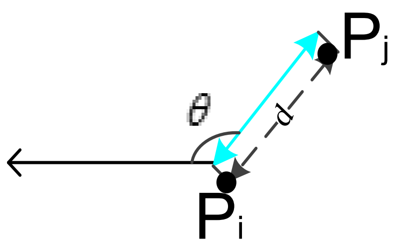
Pairs of pixel idx_i

the selected neighbor pixels set of pixel idx_i

Hierarchical Feature Matching Method (based on multi-level dense feature)

Step3: adaptive distance and orientation-based filtering

$$\left\langle \left(x_i, y_i\right), \left(x_j, y_j\right) \right\rangle$$



$$(d, \theta) \longrightarrow (d_t, \theta_k)$$

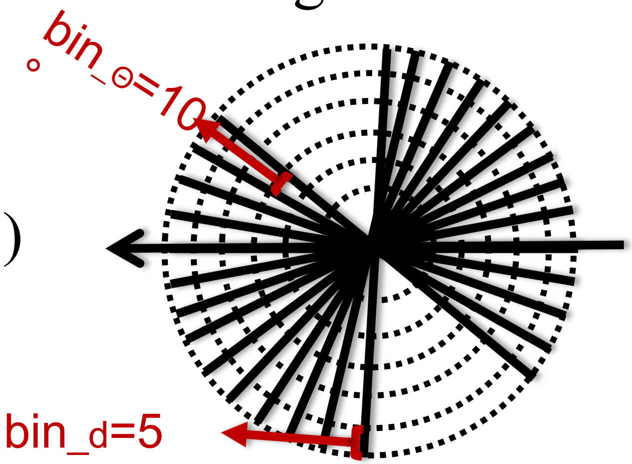
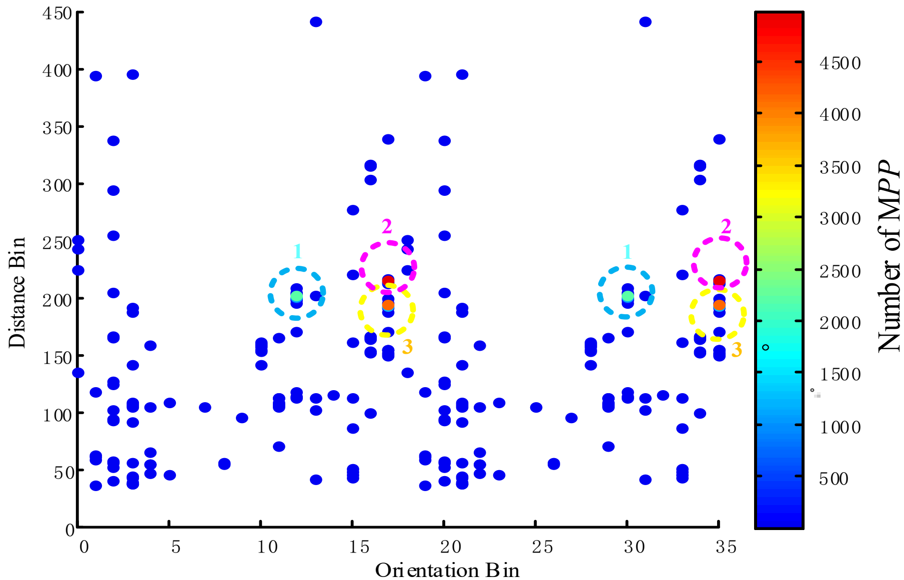
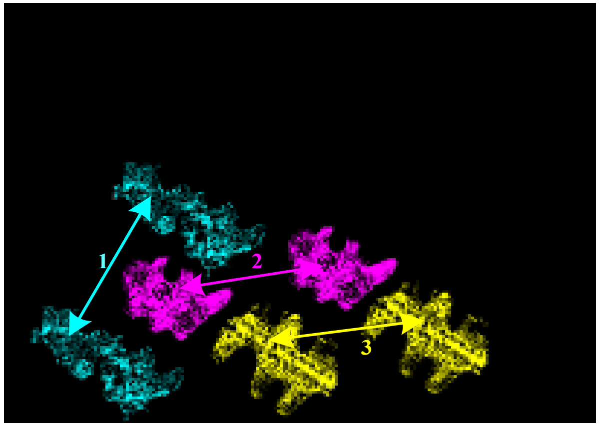


diagram of log-polar histogram bins



(a)



(b)

AGENDA

- **Introduction**
- **Related Works**
- **Proposed Methodology**
  - Multi-Level Feature Extraction
  - Adaptive Matching Methods for Copy-Move Forgery Detection
  - Experimental Results
- **Conclusions & Future Works**

EXPERIMENTAL RESULTS

------METRICS

$$precision = \frac{correctly \quad detected \quad forged \quad pixels \quad / \quad images}{detected \quad forged \quad pixels \quad / \quad images}$$

$$recall = \frac{correctly \quad detected \quad forged \quad pixels \quad / \quad images}{truely \quad forged \quad pixels \quad / \quad images}$$

$$F = \frac{2}{1/precision + 1/recall} = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

EXPERIMENTAL RESULTS

------DATABSET

	Parameter	Range	Step	CMFD_A (2064)	CMFD_B (2480)
Plain Copy-Move				48	80
Original Image				48	80
JPEG Compression	Quality Factor	20 ~ 100	10	432	720
Adding Noise	Standard Deviation	20 ~ 100	20	240	400
Scale	Scaling Factor	91 ~ 109%	2%	480	800
Rotation	Angle	2° ~ 10°	2°	240	400
Down-Sampling		90 ~ 10%	20%	240	-
Combined Trans	cmb1,cmb2,cmb3,cmb4,cmb5,cmb6			288	-
Multiple Copies				48	-

CMFD_A: www5.cs.fau.de
CMFD_B: www.grip.unina.it

EXPERIMENTAL RESULTS

------EVALUATED METHODS

- **Image Forgery Detection Using Adaptive Over-Segmentation and Feature Point Matching (ASBFD)**
  - Single-Level Keypoint Feautre and Adaptive Feature Matching Method
- **Multi-Level Feature Extraction and Adaptive Matching for Copy-Move Forgery Detection (MSBFD)**
  - Multi-Level Keypoint Feautre and Adaptive Patch Matching Method
- **Multi-Level Dense Descriptor and Hierarchical Feature Matching for Copy-Move Forgery Detection (MFHFD)**
  - Multi-Level Dense Feature and Hierarchical Feature Matching Method

Detection results under plain copy-move forgery in the image level

Methods	*precision* (%)	*recall* (%)	*F* (%)
Bravo [20]	87.27	100	93.20
Circle[22, 24]	92.31	100	96.00
SIFT[13]	88.37	79.17	83.52
SURF[13]	91.49	89.58	90.53
Zernike[21]	92.31	100.0	96.00
SBFD [9]	70.16	83.33	76.18
ASBFD	**96.0**	**100.0**	**97.96**
MSBFD	**90.57**	**100.0**	**95.05**
MFHFD	**88.89**	**100.0**	**94.12**

Detection results under plain copy-move forgery in the pixel level

Methods	*precision* (%)	*recall* (%)	*F* (%)
Bravo [20]	98.81	82.98	89.34
Circle[22, 24]	98.69	85.44	90.92
SIFT[13]	60.80	71.48	63.10
SURF[13]	68.13	76.43	69.54
Zernike[21]	95.07	87.72	91.25
SBFD [9]	84.90	54.095	65.16
ASBFD	**97.22**	**83.73**	**89.97**
MSBFD	**95.22**	**90.60**	**92.85**
MFHFD	**87.20**	**89.68**	**88.84**

Comprehensive Comparisons

Method	Robustness				Resolution(%)		
	JPEG Comp	Noise Addition	Scale	Rotation	High 100-70	Medium 70-40	Low 40-10
	Quality Factor	Standard Derivation	Scaling Factor	Angle	*F*-scores		
SIFT[13]	Fail	0.06	0.99-1.01	Fail	0.66-0.60	0.60-0.41	0.41-0.08
SURF[13]	50	0.06	0.97-1.09	10°	0.72-0.61	0.61-0.42	0.42-0.02
Bravo [20]	Fail	Fail	Fail	Fail	0.90-0.44	0.44-0.43	0.43-0.11
Circle[22, 24]	90	Fail	0.97-1.03	10°	0.86-0.24	0.24-0.23	0.23-0.03
Zernike[21]	50	0.02	0.93-1.09	10°	0.90-0.77	0.77-0.62	0.62-0.12
SBFD [9]	20	0.10	0.91-1.09	10°	-----		
ASBFD	**30**	**0.02**	**0.91-1.09**	**10°**	**0.93-0.83**	**0.83-0.61**	**0.61-0.04**
MSBFD	**20**	**0.10**	**0.91-1.09**	**10°**	**0.95-0.84**	**0.84-0.67**	**0.67-0.05**
MFHFD	**20**	**0.10**	**0.91-1.09**	**10°**	**0.91-0.88**	**0.88-0.87**	**0.87-0.54**

- In this table, the F measure is defined as: $F \geq 0.5$, to make detection succeed; otherwise, detection will fail.
- '-' means that the paper did not mention the results in that condition.

EXPERIMENTAL RESULTS

------RUNNING TIME

Methods	Running Time (s)
Bravo [20]	6180.42
Circle[22, 24]	5103.43
SIFT[13]	610.96
SURF[13]	1052.12
Zernike[21]	7065.18
SBFD [9]	1357.66
ASBFD	**473.58**
MSBFD	**1167.94**
MFHFD	**567.34**

AGENDA

- **Introduction**
- **Related Works**
- **Proposed Methodology**
  - Multi-Level Feature Extraction
  - Adaptive Matching Methods for Copy-Move Forgery Detection
  - Experimental Results
- **Conclusions & Future Works**

CONCLUSIONS

- **Feature Detectors:**
  - Single-Level Keypoint Feature Extraction
  - Multi-Level Keypoint Feature Extraction
  - Multi-Level Dense Feature Extraction
- **Feature Matching Methods:**
  - Adaptive Feature Matching Method
  - Adaptive Patch Matching Method
  - Hierarchical Feature Matching Method

FUTURE WORKS

- **Feature Extraction**
  - simple, robustness
- **Feature Matching Method**
  - fast matching method
  - exhaustive searching → approximately searching
- **Video Forgery Detection**

PUBLICATIONS

1. **Chi-Man Pun**, Xiao-Chen Yuan, and Xiu-Li Bi, "Image Forgery Detection Using Adaptive Over-Segmentation and Feature Points Matching," *Information Forensics and Security, IEEE Transactions on*, vol. 10, pp. 1705-1716, 2015. [SCI]
2. Xiu-Li Bi, **Chi-Man Pun**, Xiao-Chen Yuan, “Multi-Level Dense Descriptor and Hierarchical Feature Matching for Copy–Move Forgery Detection”, *Information Sciences*, Elsevier, Vol. 345, pp. 226-242, 2016. [SCI]
3. Xiu-Li Bi, **Chi-Man Pun**, and Xiao-Chen Yuan. "Multi-scale feature extraction and adaptive matching for copy-move forgery detection." *Multimedia Tools and Applications* (2016): 1-23. [SCI]

THANK YOU !

Q & A