# Digital Image Manipulation

# Anti-forensics

# *Previous Works*


[1] H. Farid, “Exposing digital forgeries by detecting traces of resampling,” IEEE Trans on SP, 2005.
[2] B. Mahdian, “Blind authentication using periodic properties of interpolation,” IEEE TIFS, 08.

[3] J. Huang, ”A novel method for detecting cropped and recompressed image block”, ICASSP 2007.
[4] H. Farid, “Exposing digital forgeries from JPEG ghosts,” IEEE TIFS, 09.

[5] K. J. R. Liu, “Blind forensics of contrast enhancement in digital images,” in ICIP, 2008.
[6] K. J. R. Liu, “Forensic detection of image tampering using intrinsic statistical fingerprints in histograms,” Proc.APSIPA,2009.
[7] K. J. R. Liu, “Forensic estimation and reconstruction of a contrast enhancement mapping,” ICASSP 2010.
[8] K. J. R. Liu, " Forensic detection of image forgeries using intrinsic fingerprints of pixel value mappings ", IEEE TIFS, 2010.
[9] Y. Zhao, “Forensic estimation of gamma correction in digital images,” ICIP, 2010.

[10] D. Hsiao and S. Pei, “Detecting digital tampering by blur estimation,” 1st SADFE, 2005.

[11] Y. Zhao, “Detection of image sharpening based on histogram aberration and ringing artifacts,” ICME, 2009.
[12] Y. Zhao, “Unsharp Masking Sharpening Detection via Overshoot Artifacts Analysis”, *IEEE SPL*, 2011.

[13] M. Kirchner and J. Fridrich, “On detection of median filtering in digital images,” SPIE, 2010.
[14] Y. Zhao, “Forensic detection of median filtering in digital images,” ICME, 2010.

[15] N. Memon, “Image manipulation detection,” Journal of Electronic Imaging, 2006.
[16] M. Wu, “Tampering identification using empirical frequency response,” ICASSP, 2009.

# Outline

**Anti-forensics of Contrast Enhancement**

# Attacks on Contrast Enhancement Forensics

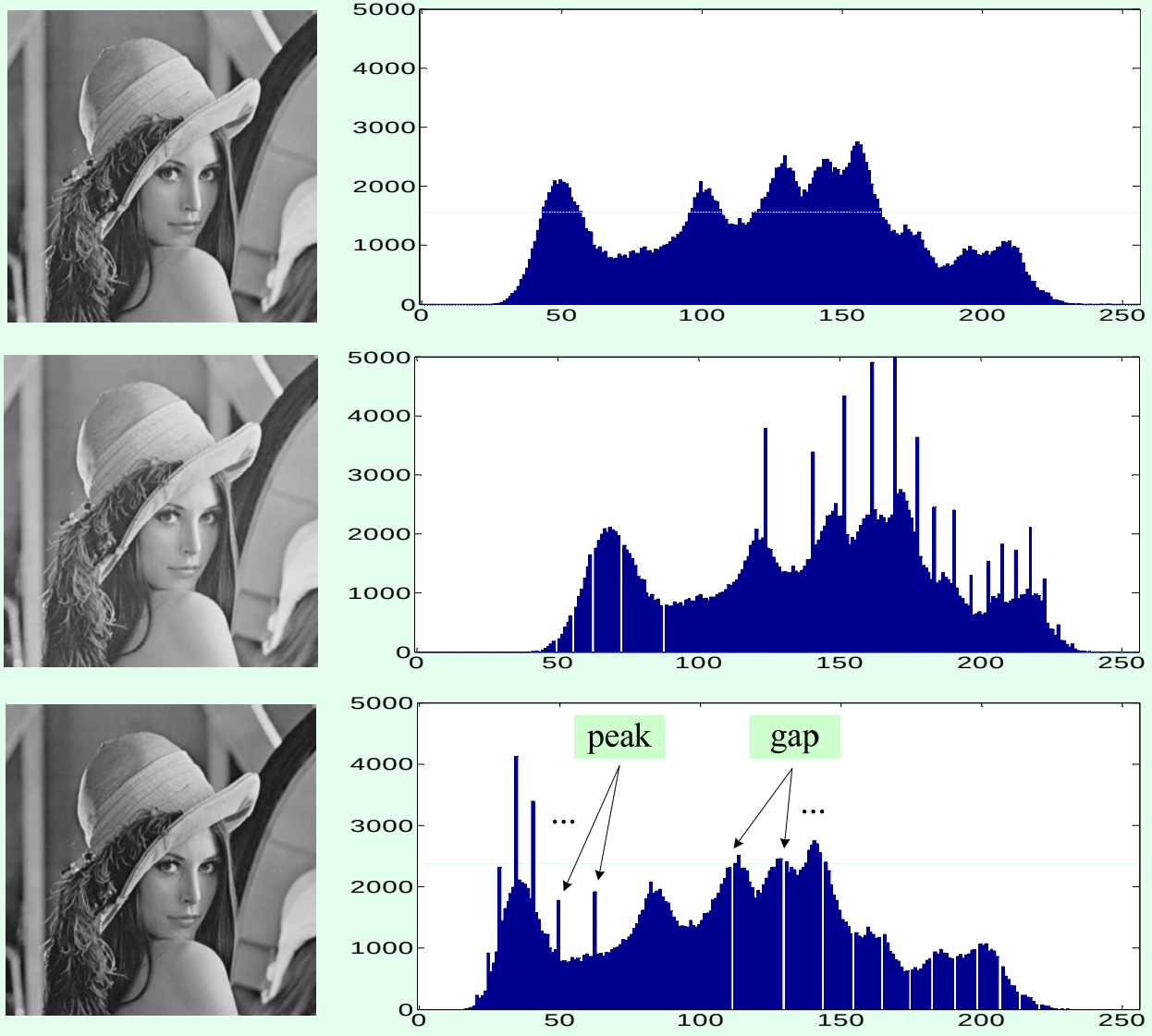
## ➢ Existing anti-forensic techniques

- ✓ Target: (1) undetectable image *resampling*,
  (2) synthesis of *color filter array pattern*,
  (3) anti-forensic of JPEG *compression*.
- ✓ Type: manipulation hiding attacks.

## ➢ Proposed attacks

- ✓ Target: undetectable *contrast enhancement* (CE)
- ✓ Type: manipulation forging & hiding attacks.

# Forensic Detection of CE (1)



# Forensic Detection of CE (2)

➢ **CE detection algorithm** (IEEE TIFS, 2010)

1). Obtain the test image's histogram $h(x)$.

2). Calculate g($x$) as follows,

$$g(x) = p(x) \cdot h(x)$$

where $p(x)$ is a pinch off function for combating the false high frequency effect caused by high end and low end saturated images.

3). Compute the high frequency measure $F$ :

$$F = \frac{1}{N} \sum_{\omega} \left| \beta(\omega) G(\omega) \right| \qquad G(\omega) = \mathrm{DFT}(g(x)) \qquad \beta(\omega) = \begin{cases} 1 & |\omega| \geq c \\ 0 & |\omega| < c \end{cases}$$

where $c$ is a user specified cutoff frequency.

4). Thresholding method to determine if contrast enhancement has been performed, $F$ greater than the threshold signifying the detection of contrast enhancement.

# Forensic Detection of CE (3)

## ➢ Detection of locally applied CE (IEEE TIFS, 2010)

✓ The above forensic technique is extended to detect locally applied contrast enhancement.

## ➢ Forensic estimation of CE mapping (ICASSP, 2010)

✓ An iterative algorithm to jointly estimate any arbitrary CE mapping as well as the pixel value histogram of the image before contrast enhancement.

✓It should be noted that the mapping estimation can be reliable only when the employed CE manipulations have been detected previously. Otherwise, the estimation is eyeless and couldn't be alleged as forensic reconstruction of CE.

# CE Forging Attack (1)

## ➢ Locate Peak & Gap Bins

- **Universal CE forging**

✓ Position of the peak/gap bins are designated randomly.

- **Targeted CE forging**

✓ Given the targeted mapping $y = m(x)$ which is to be falsified, we can compute the position set of peak and gap bins which should appear in the histogram of attacked original image.

$$\Psi_p = \left\{ y \,\middle|\, \sum_{x \in \Omega} l(m(x) = y) > 1 \right\} \qquad x \in \Omega \quad y \in \Omega$$

$$\Psi_g = \left\{ y \,\middle|\, \sum_{x \in \Omega} l(m(x) = y) = 0 \right\} \qquad \Omega = \{0, 1, \ldots, 255\}$$

# CE Forging Attack (2)

## ➢ Generate Peak Bin by Moving Histogram

✓ Invariance of the pixel number within local histogram:

$$M \cdot 2L + \lambda \cdot M = \sum_{x=x_i-L}^{x_i+L} h_X(x)$$

where $x_i \in \Psi_p$ $\quad M \triangleq \frac{1}{2L}\left(\sum_{y\in\Theta} h_Y(y)\right)$ $\quad \Theta = [x_i - L,\ x_i) \cup (x_i,\ x_i + L]$

✓ Generated peak bin: $h_Y(x_i) = \frac{\lambda}{2L+\lambda} \sum_{x=x_i-L}^{x_i+L} h_X(x)$

✓ Decrease of the neighboring bins: $\Delta_y = w(y - x_i) \cdot \left(h_Y(x_i) - h_X(x_i)\right)$ $\quad y \in \Theta$

where $w(x) \sim N(0, \sigma)$

# CE Hiding Attack (1)

## ➢ Mapping Decomposition

- ➢ Traditional CE : $m(x) = round\left[m_0(x)\right]$
  - ✓ Step (1): $y_0 = m_0(x)$ ( the primary mapping ) $m_0 : \mathbb{Z} \to \mathbb{R}$
  - ✓ Step (2): $y = round\left[y_0\right]$

- ➢ Histogram ~ CE: $h_Y(y) = \sum_{x \in \Omega} h_X(x)\, l(y_0 \in [y - \frac{1}{2}, y + \frac{1}{2}))$
  - ✓ Potential peak & gap:

$$\Psi_p = \left\{ y \;\middle|\; \sum_{x \in \Omega} l(y_0 \in [y - \tfrac{1}{2}, y + \tfrac{1}{2})) > 1 \right\}$$

$$\Psi_g = \left\{ y \;\middle|\; \sum_{x \in \Omega} l(y_0 \in [y - \tfrac{1}{2}, y + \tfrac{1}{2})) = 0 \right\}$$

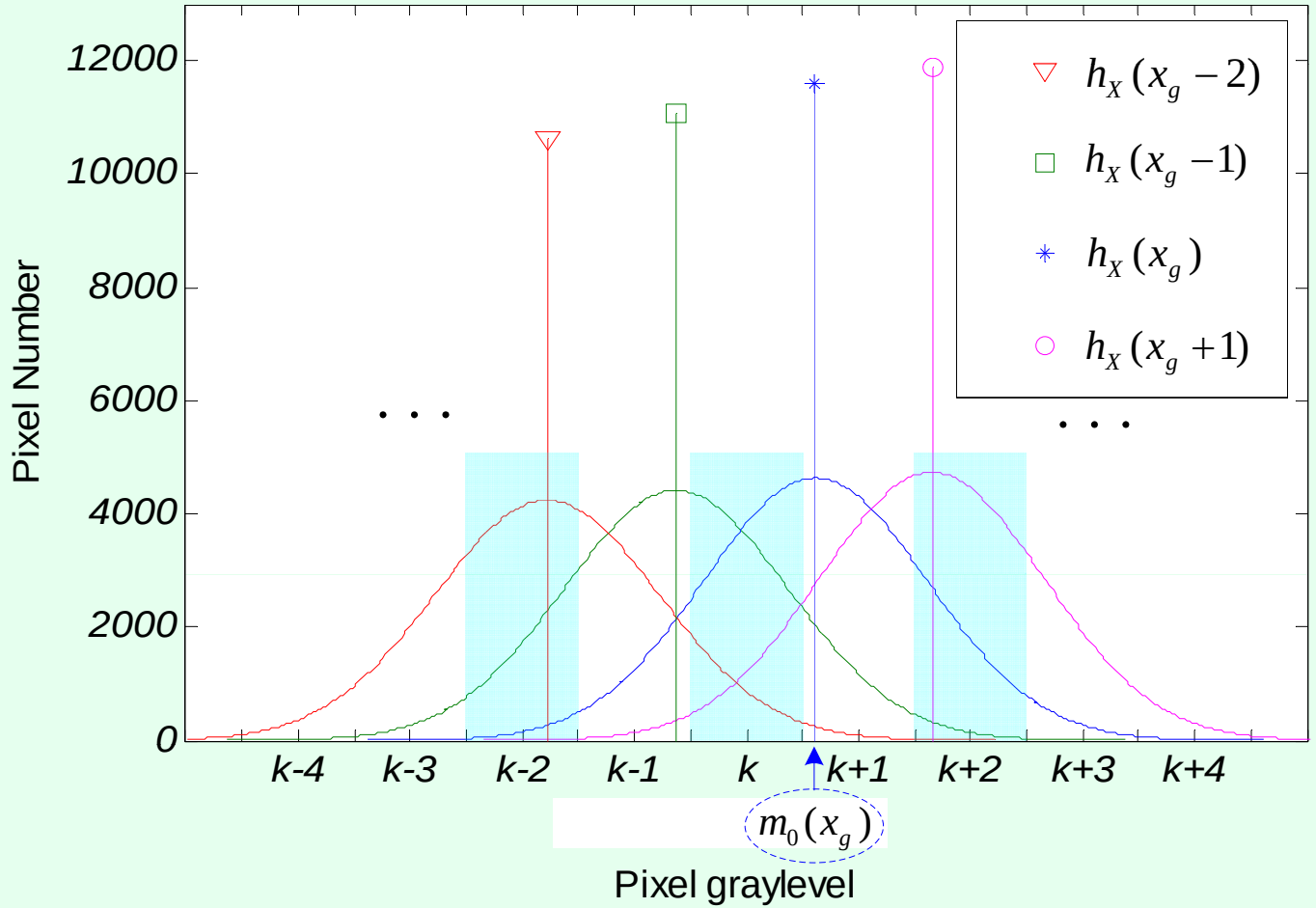
# CE Hiding Attack (2)

## ➢ Local Random Dithering (Integrated Attack)

$$m(x) = round\left[m_0(x) + n_x\right]$$

$$n_x \sim N(0,\ \sigma_x^2)$$

$$\sigma_x = \begin{cases} \sigma_1 & if \quad round\left[m_0(x)\right] \in \left\{\widehat{\Psi}_p \cup \widehat{\Psi}_g\right\} \\ \sigma_2 & else \end{cases}$$



# CE Hiding Attack (3)

## ➢ Analysis for Artifacts Removal

➢ New primary mapped value: $t_{x_g} = m_0(x_g) + n_{x_g}$

$$t_{x_g} \sim N(m_0(x_g),\ \sigma_x^2)$$

➢ Histogram ~ new CE

$$h_Y(k+i) = \sum_{j=-\infty}^{+\infty} h_X(x_g + j) \int_{k+i-\frac{1}{2}}^{k+i+\frac{1}{2}} G_t\left(m_0(x_g + j), \sigma_{x_g}^2\right) dt$$

where $G_t\left(m_0(x_g + j), \sigma_{x_g}^2\right) = \frac{1}{\sqrt{2\pi}\sigma_{x_g}} \exp\left[-\frac{\left(t - m_0(x_g + j)\right)^2}{2\sigma_{x_g}^2}\right]$

# CE Hiding Attack (4)

## ➢ Adding Noise (Postprocessing Attack )

➢ Formulated as:

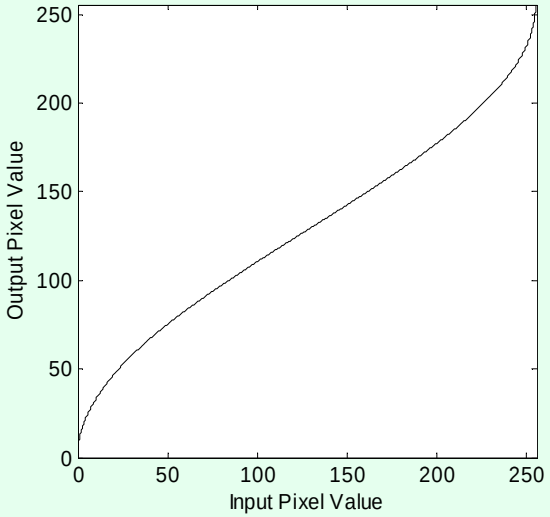
$$y = round\left[m(x) + n_x\right]$$
$$= round\left[round\left[m_0(x)\right] + n_x\right]$$

where $n_x \sim N(0, \sigma_x^2)$ $\quad \sigma_x = \begin{cases} \sigma_1 & if \quad round\left[m_0(x)\right] \in \left\{\widehat{\Psi}_p \cup \widehat{\Psi}_g\right\} \\ \sigma_2 & else \end{cases}$

✓ If unknowing the used mapping function, the potential peak set $\Psi_p$ and the potential gap set $\Psi_g$ have to be searched by applying a simple and strict threshold-based detector.

# Evaluation – *Test Data*

- **693 unaltered photograph images**
  - ✓ Size: 1200×900 ~ 2832×2128 pixels
  - ✓ JPEG
  - ✓ Use green channel as grayscale test image
- **Simulate traditional CE manipulations**
  - ✓ Gamma correction: $m(x) = round\left[255\left(\frac{x}{255}\right)^{\gamma}\right]$
  - ✓ 'S' mapping




  - ✓ Parameter Setting:

$\gamma = 0.5 \sim 2$ $\quad c = 7\pi/8$

$\sigma_1 = \sigma_2 = 1.0$

# Evaluation – *Performance Metric*

## ➢ (un)Detectability : Two Sets

- ✓ Training Set : 347 (random chosen)
  - $FN_0$ 、 $FP_0$ → $F$ - threshold
- ✓ Test Set : 346
  - $F$ - threshold → TN 、 TP

## ➢ Image Quality of Attacked Images

- ✓ PSNR
  - Forging attack: attacked image vs. unaltered image
  - Hiding attack: attacked image vs. contrast-enhanced image

# Baseline Detection Results (1)

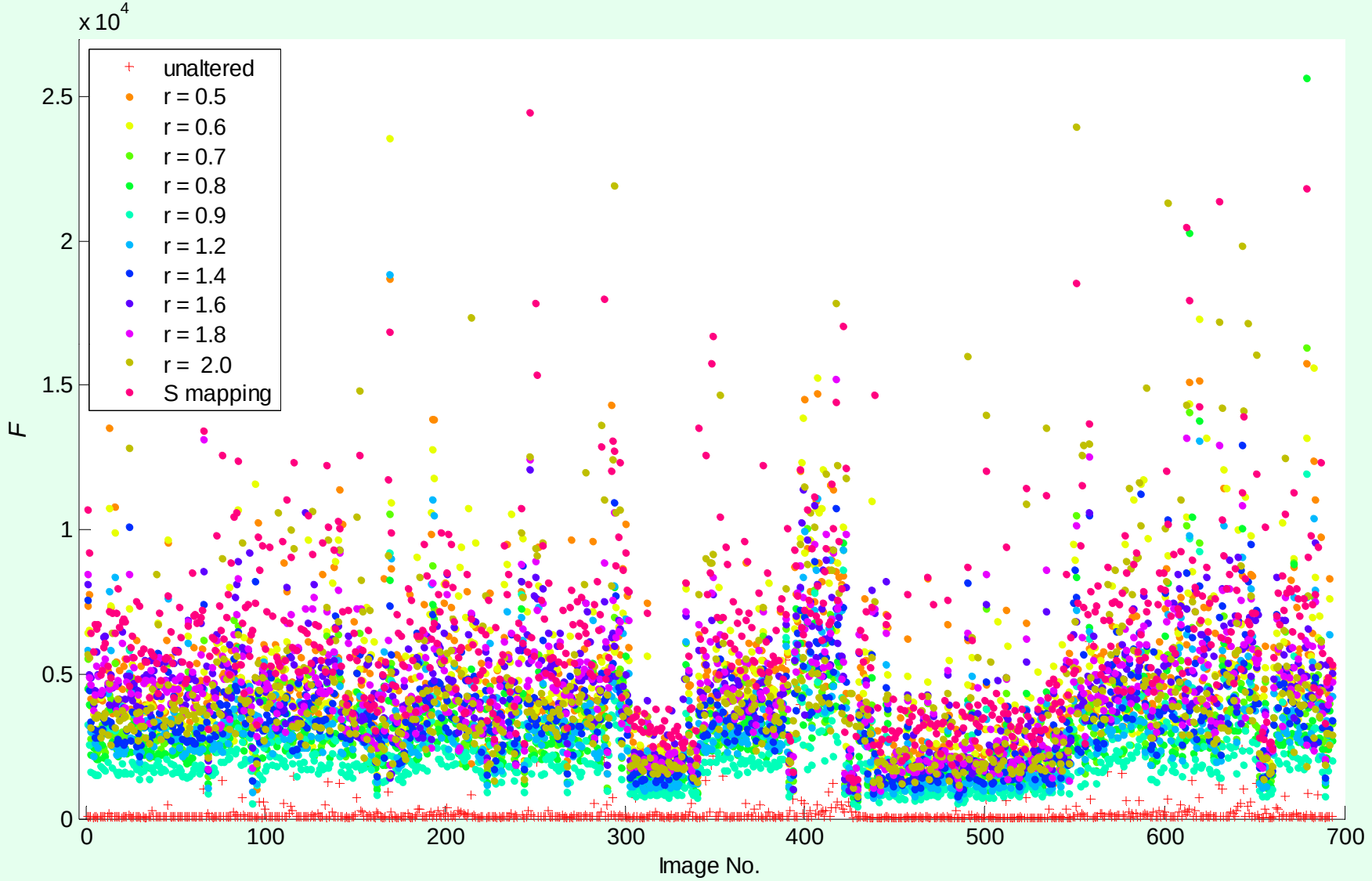



Fig. Feature value distribution of samples.

# Baseline Detection Results (2)

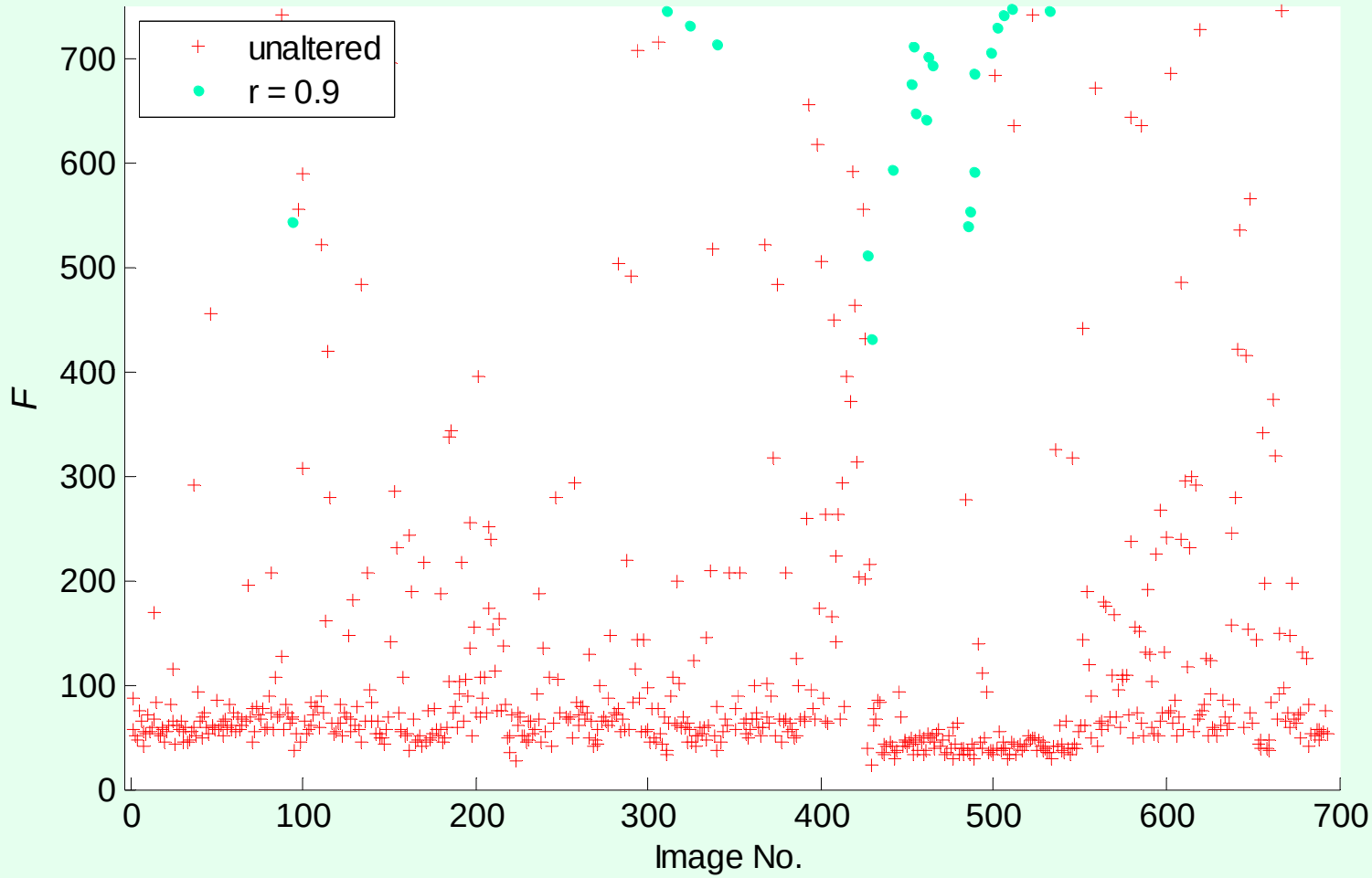



Fig. Feature value distribution of samples *(Locally magnified).*

# Baseline Detection Results (3)

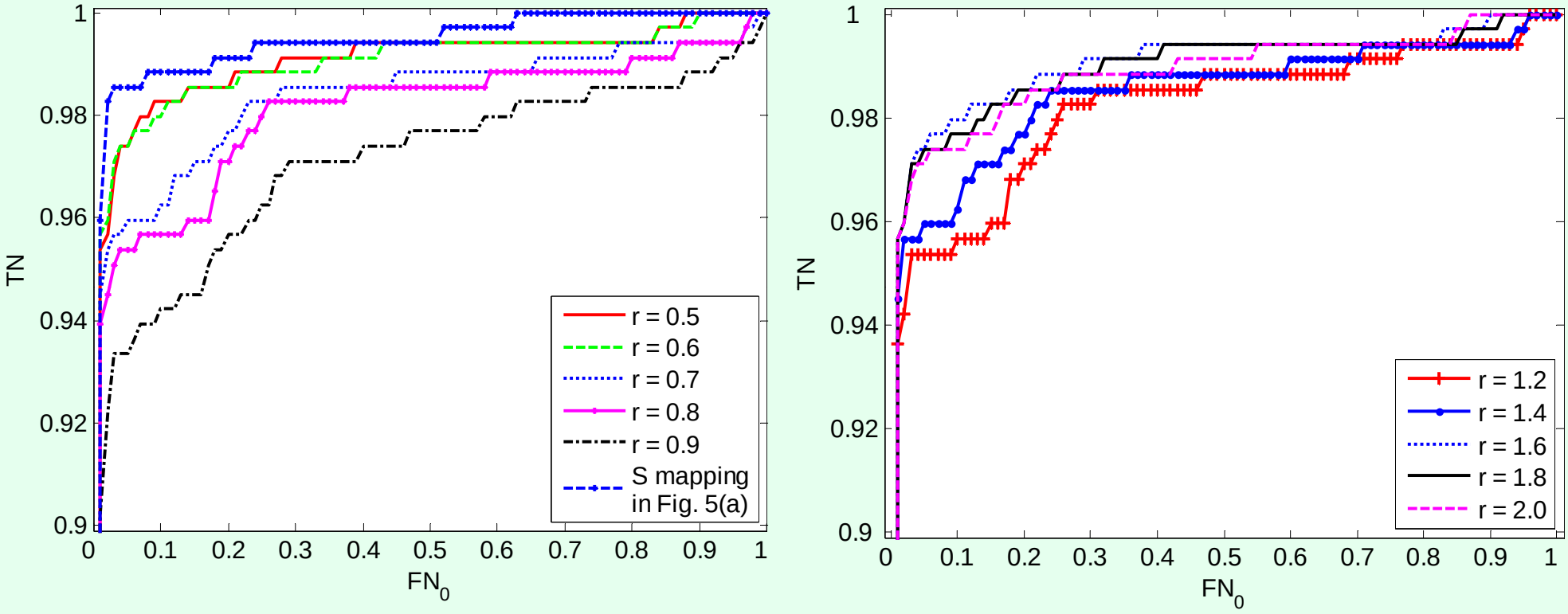



Fig. Detection rate of unaltered images (TN) under varying amounts for false negative rates ($FN_0$) .

# Baseline Detection Results (4)

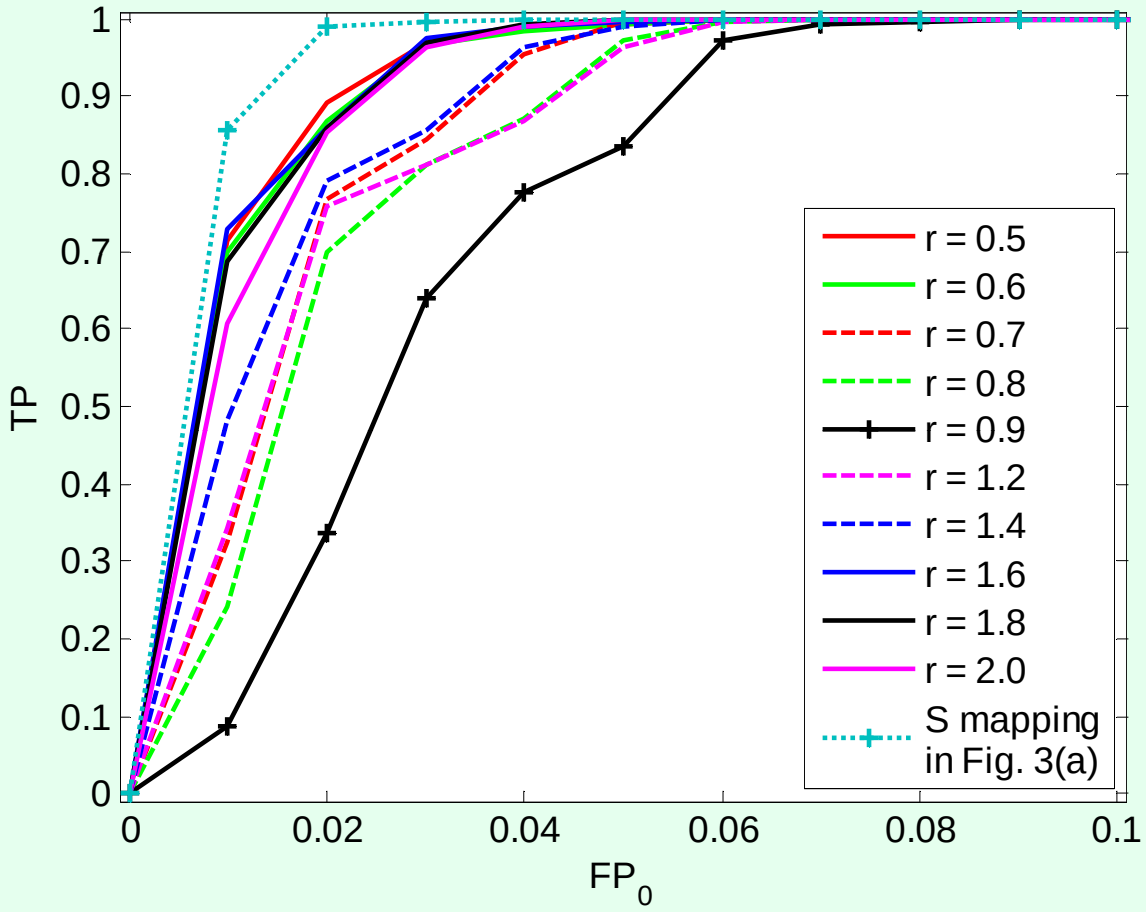



Fig. Detection rate of altered images (TP) under varying amounts for false positive rates ($FP_0$) .

# Evaluate CE Forging Attack (1)

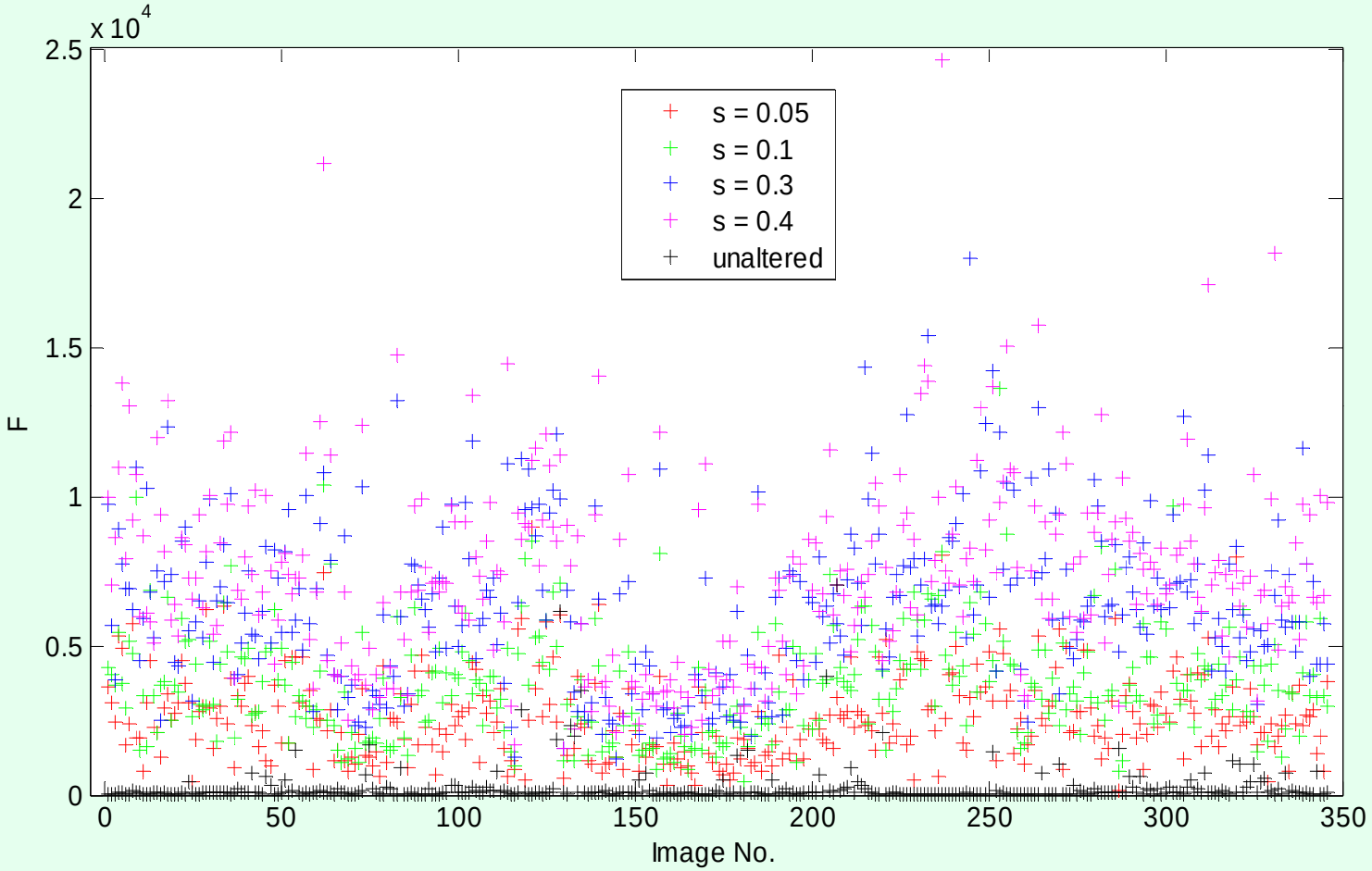



Fig. *F* – distribution for unaltered images and their *universal* CE forging attacked versions by random-LSB method.

*s: attack strength*

# Evaluate CE Forging Attack (2)

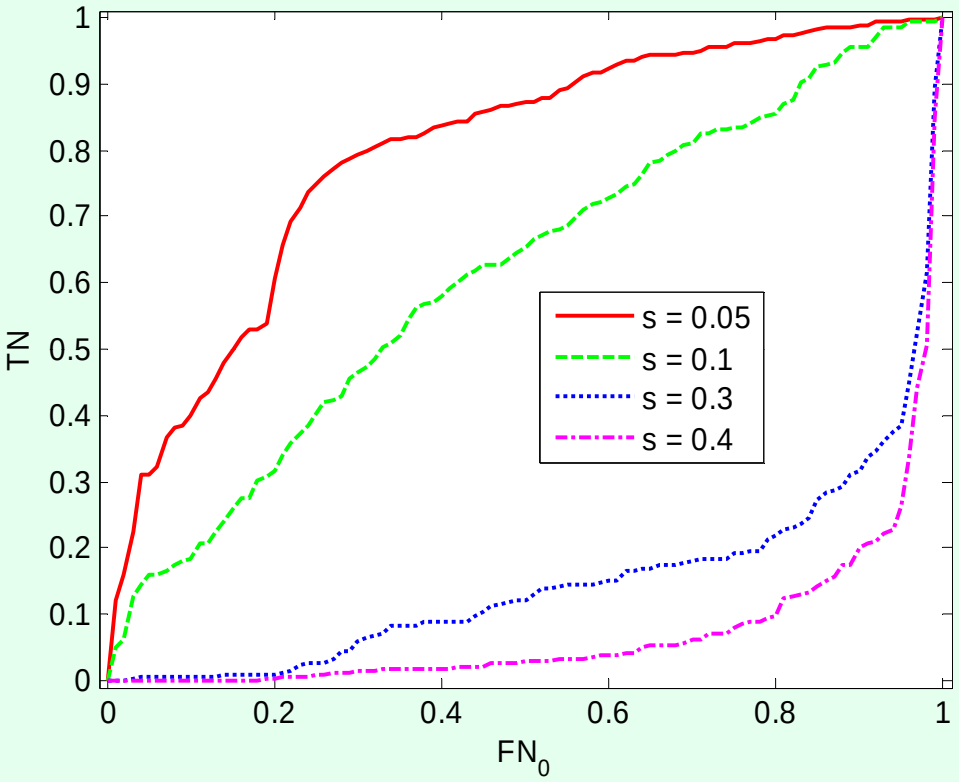



Fig. TN-$FN_0$ curve

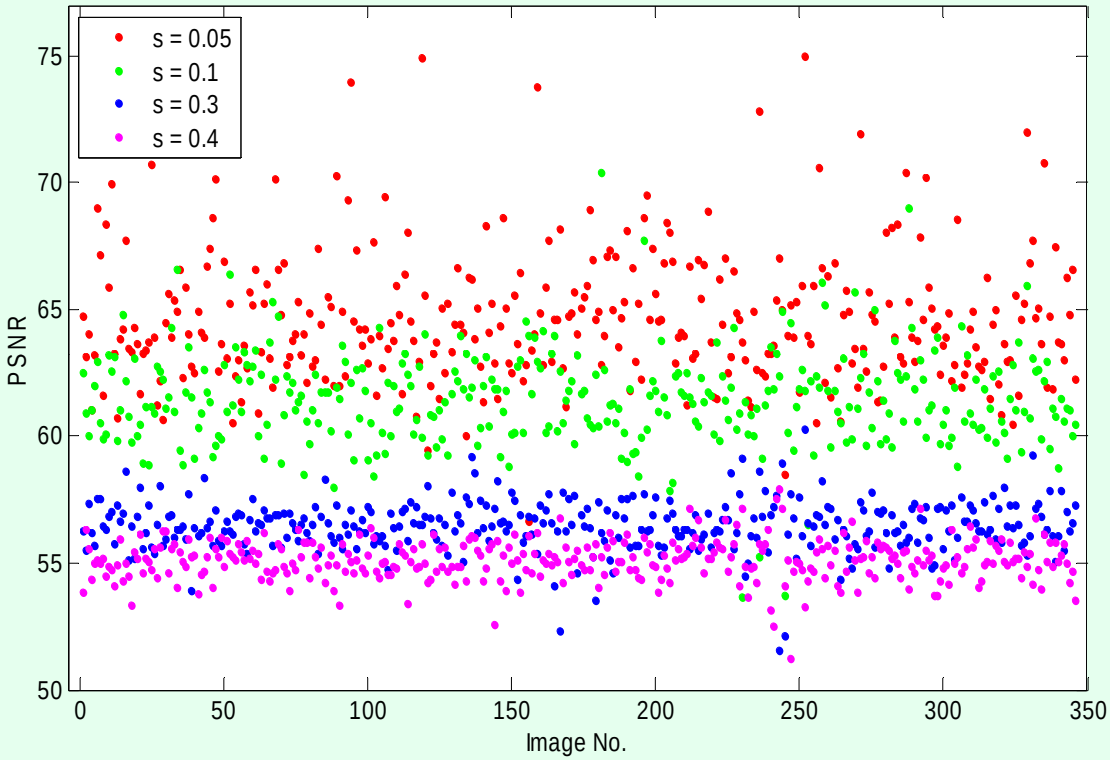



Fig. PSNR distribution

# Evaluate CE Forging Attack (3)

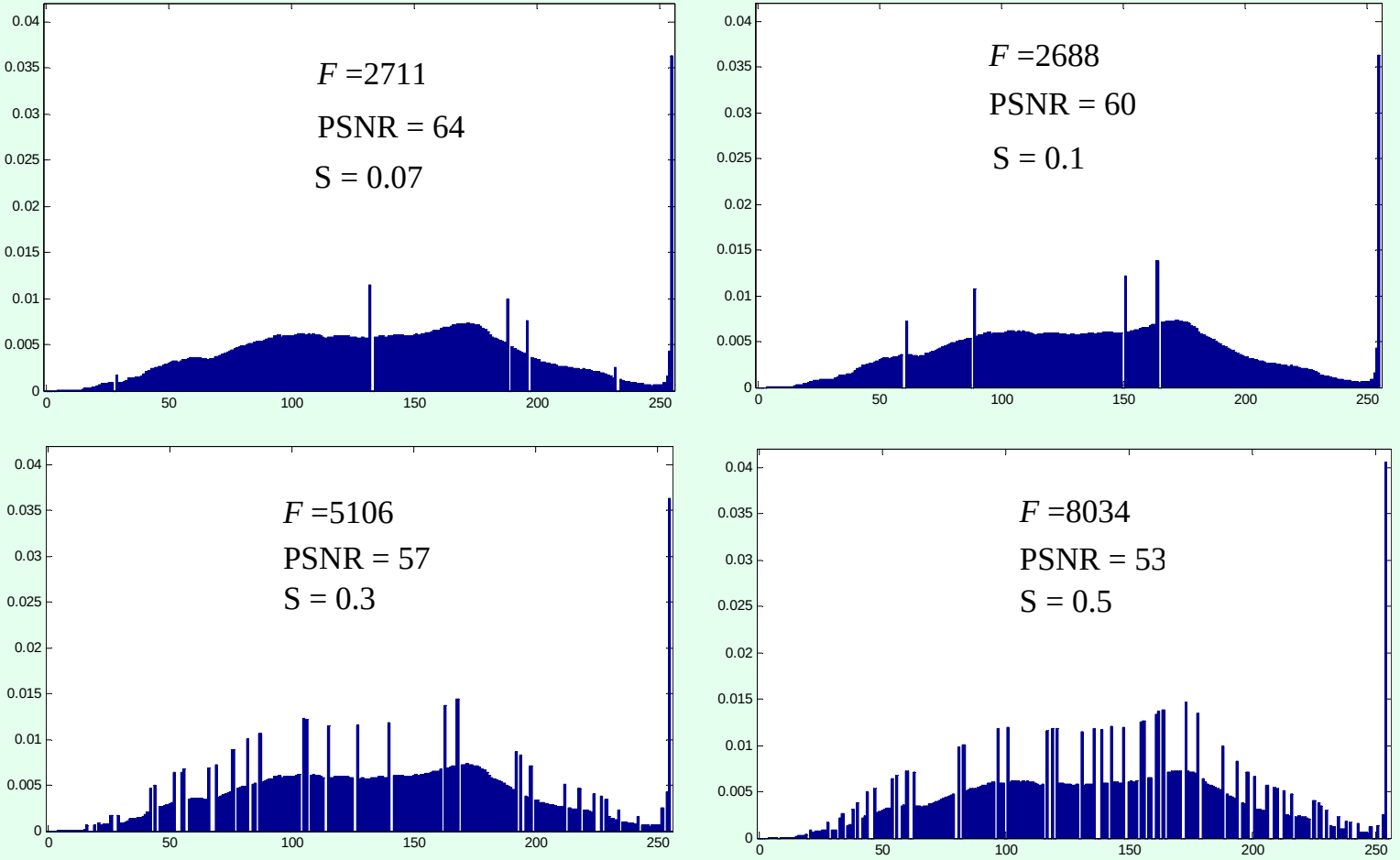



Fig. Histogram of attacked images ( random-LSB method ).

# Evaluate CE Forging Attack (4)

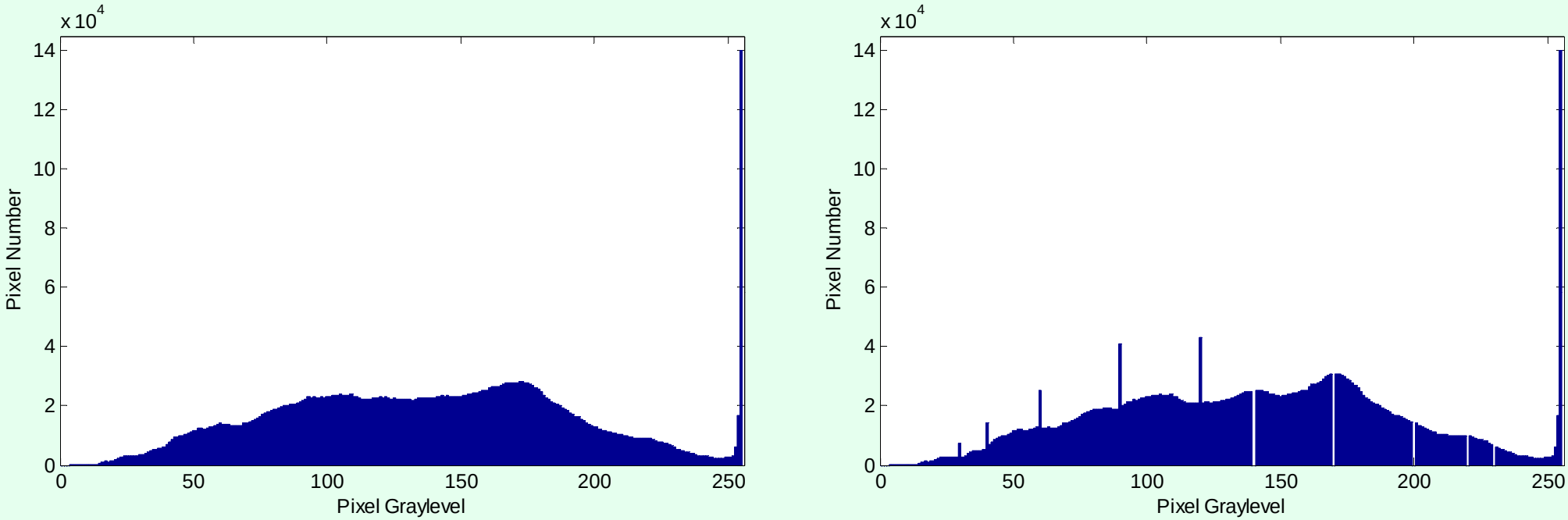



Fig. Histogram of *targeted CE forging attacked* image.

*Unaltered image:* $F = 63$

*Attacked image :* $F = 1281.7$, PSNR = 49.9

# Evaluate CE Hiding Attack (1)

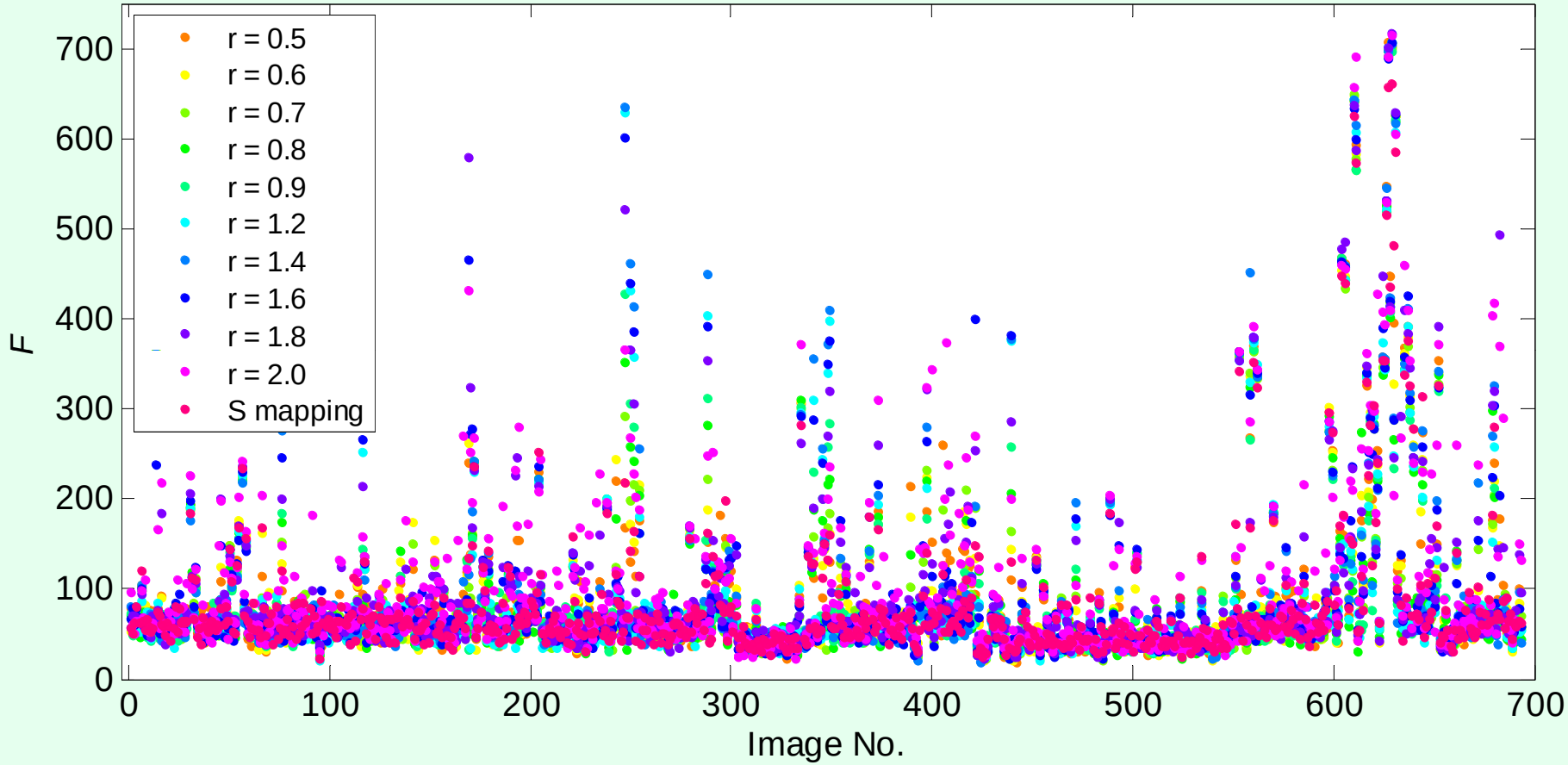



Fig. *F* distribution for *integrated CE hiding attack*.

# Evaluate CE Hiding Attack (2)

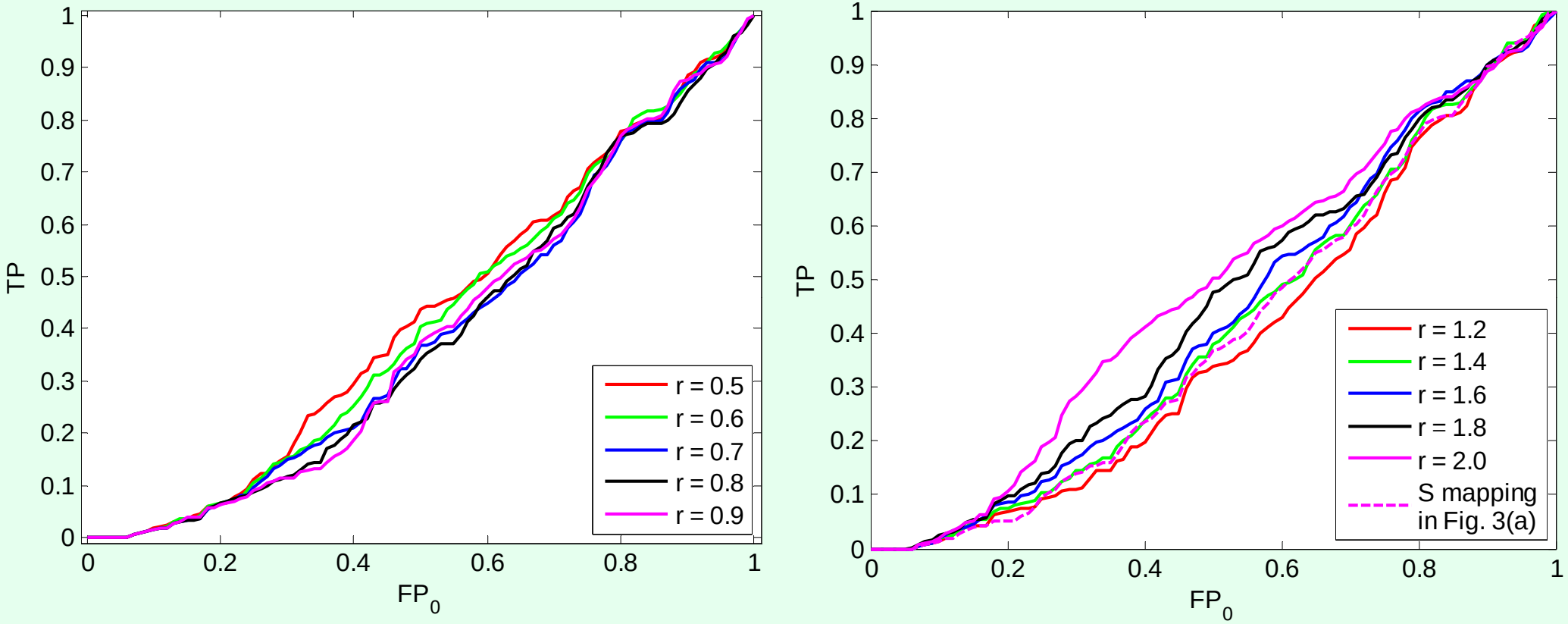



Fig. *Detection results* for *integrated CE hiding attack*.

# Evaluate CE Hiding Attack (3)

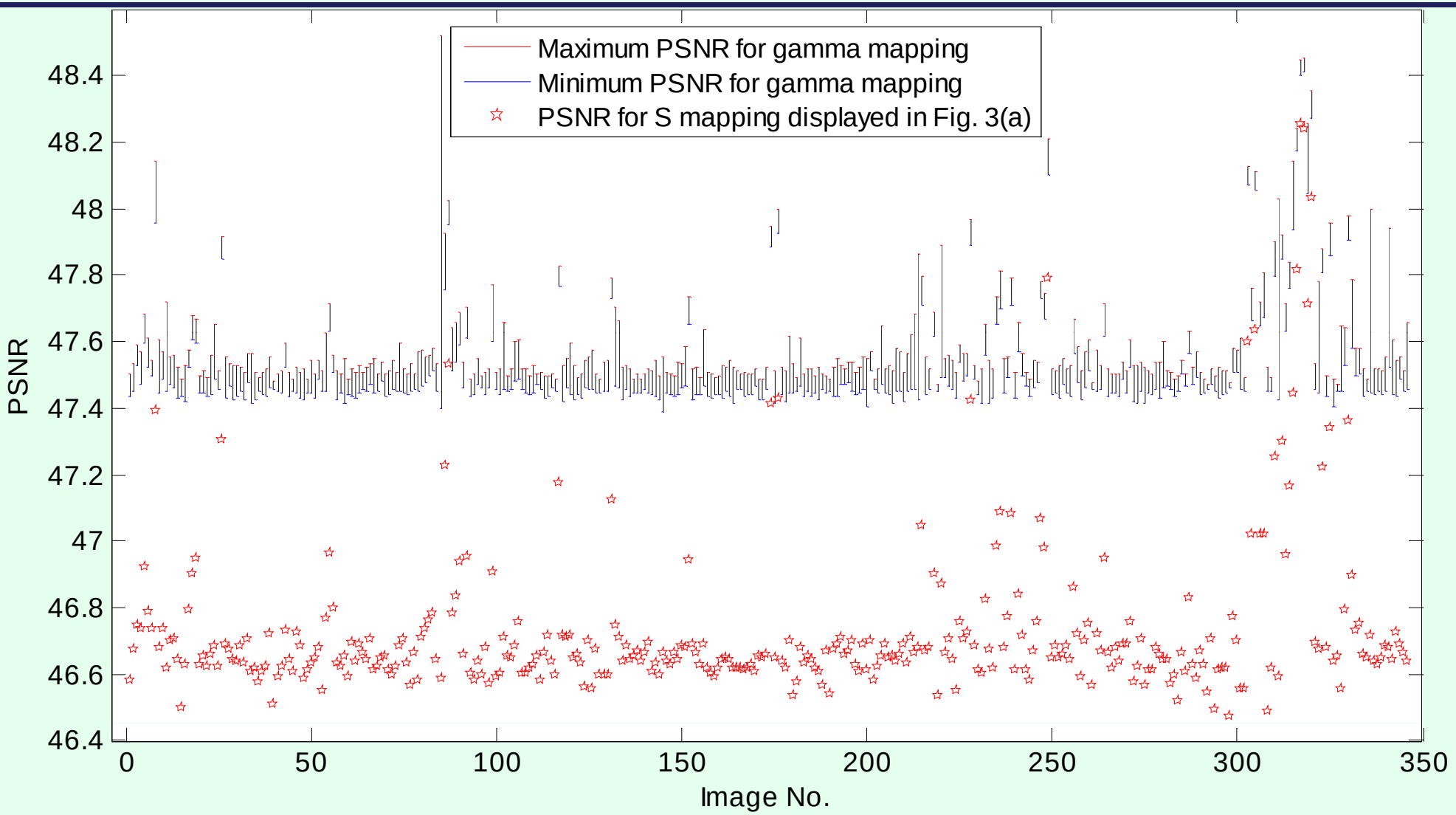



Fig. *PSNR* for *integrated CE hiding attack*.

# Evaluate CE Hiding Attack (4)

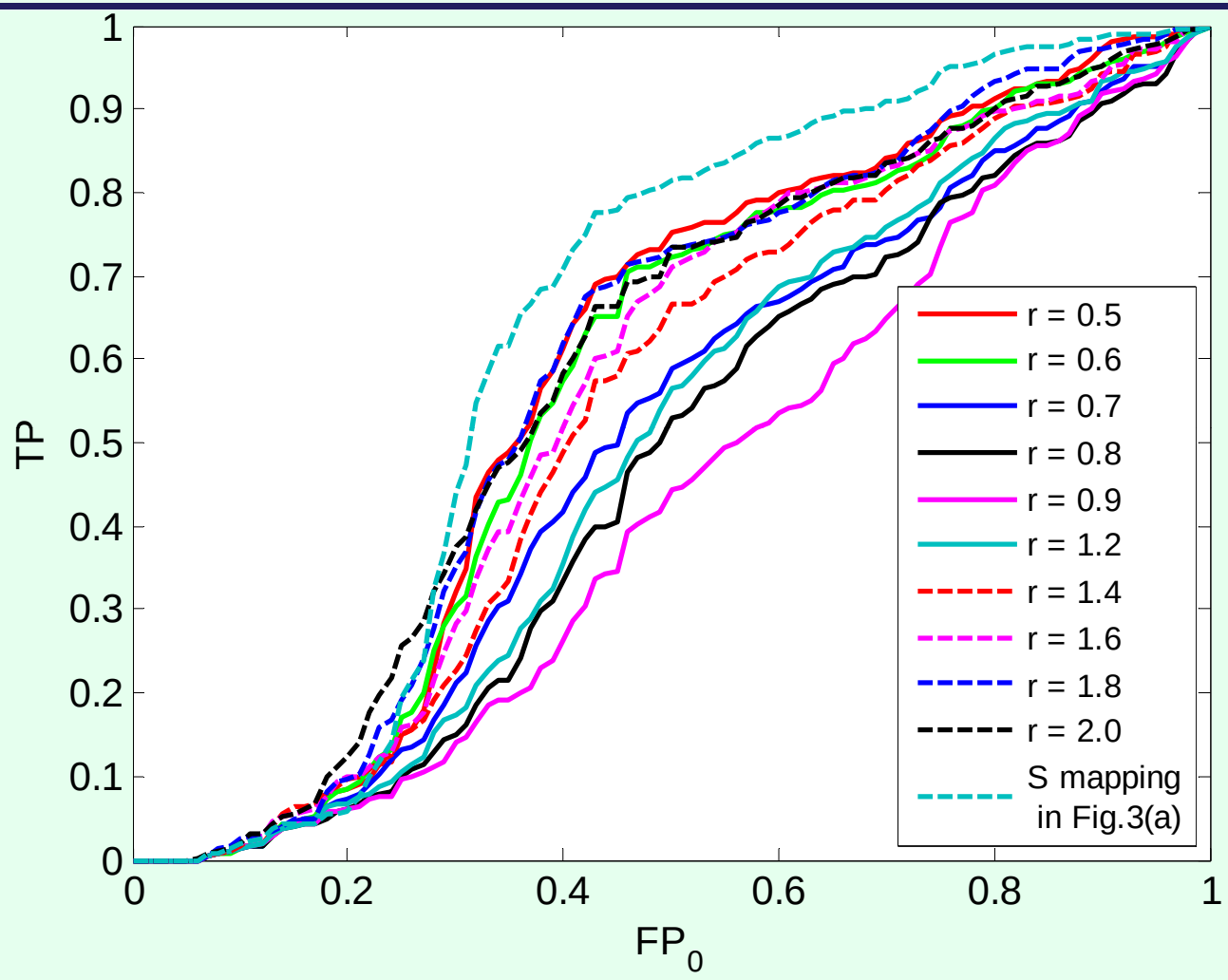



Fig. *Detection results* for *postprocessing CE hiding attack*.
(adding noise method)

# Evaluate CE Hiding Attack (5)

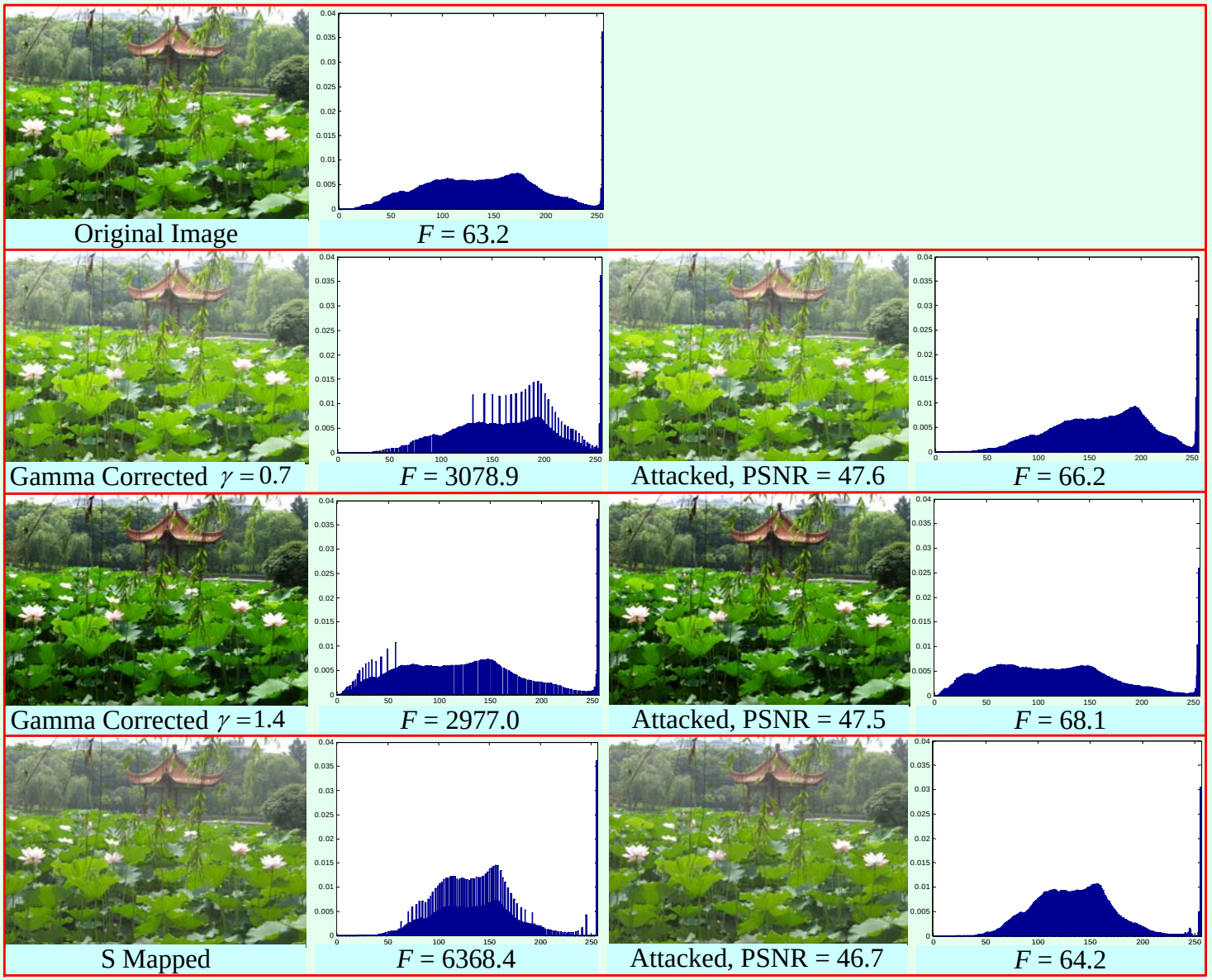



Fig. Test on *an example image* for integrated *CE hiding attack*.

# Application to Forgery Making (1)

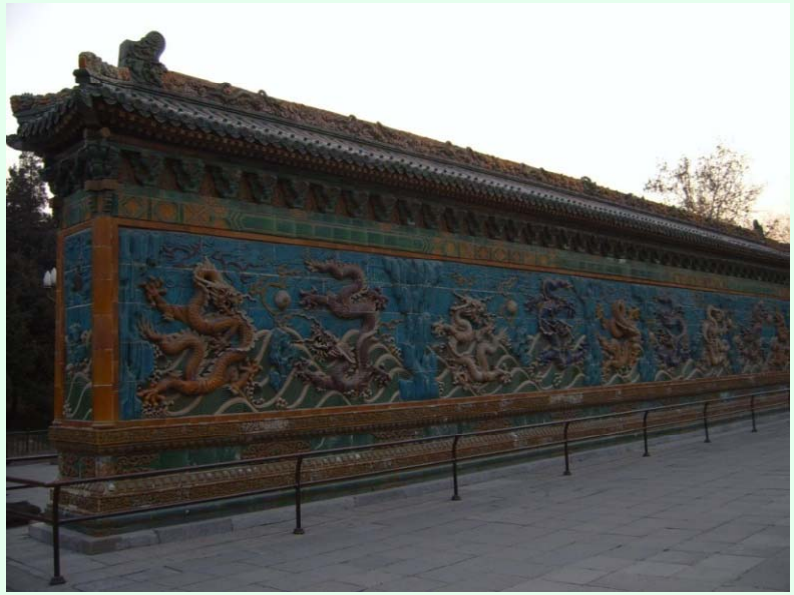
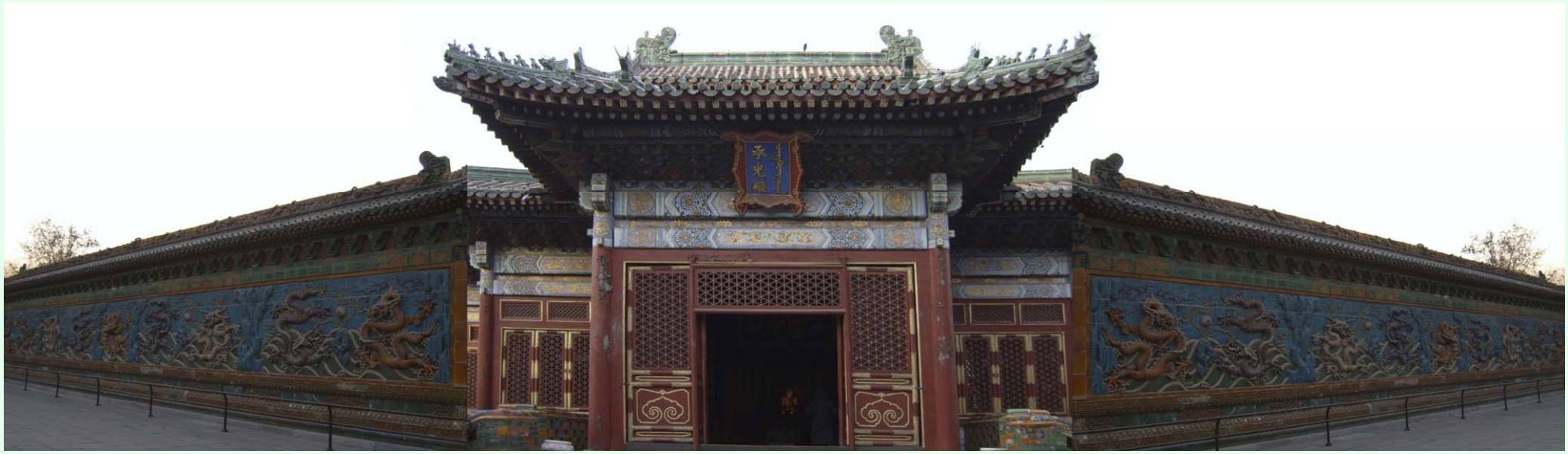


Fig. An image forgery example. without CE → Need CE ?

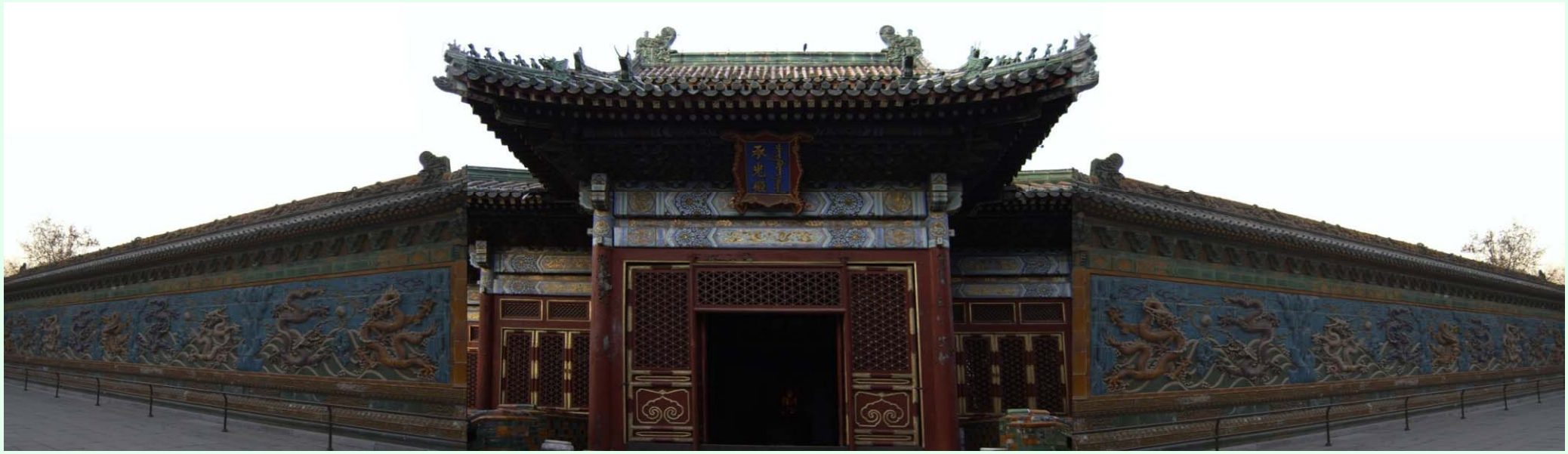
# Application to Forgery Making (2)



| | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 2 | 2 | 4 | 4 | 3 | 2 | 3 | 3 | 3 | 3 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 5 | 5 | 6 | 13 | 6 | 7 | 7 | 5 | 7 | 6 | 7 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 4 | 5 | 8 | 4 | 4 | 4 | 13 | 7 | 13 | 4 | 4 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 2 | 1 | 3 | 4 | 4 | 7 | 5 | 4 | 5 | 4 | 7 | 6 | 7 | 6 | 7 | 8 | 9 | 7 | 4 | 4 | 5 | 3 | 1 | 1 | 3 |
| 5 | 4 | 6 | 6 | 6 | 5 | 4 | 5 | 5 | 5 | 5 | 4 | 6 | 7 | 6 | 5 | 4 | 7 | 4 | 7 | 5 | 4 | 4 | 4 | 5 | 5 |
| 7 | 7 | 6 | 5 | 5 | 5 | 8 | 7 | 6 | 5 | 7 | 6 | 9 | 11 | 5 | 5 | 5 | 6 | 5 | 8 | 6 | 5 | 5 | 4 | 7 | 6 |
| 5 | 5 | 7 | 3 | 9 | 4 | 6 | 7 | 7 | 10 | 3 | 7 | 17 | 21 | 5 | 4 | 7 | 6 | 5 | 7 | 3 | 5 | 6 | 4 | 8 | 6 |

Fig. *F* -map of original forgery. (without CE) *block size : 200x200*

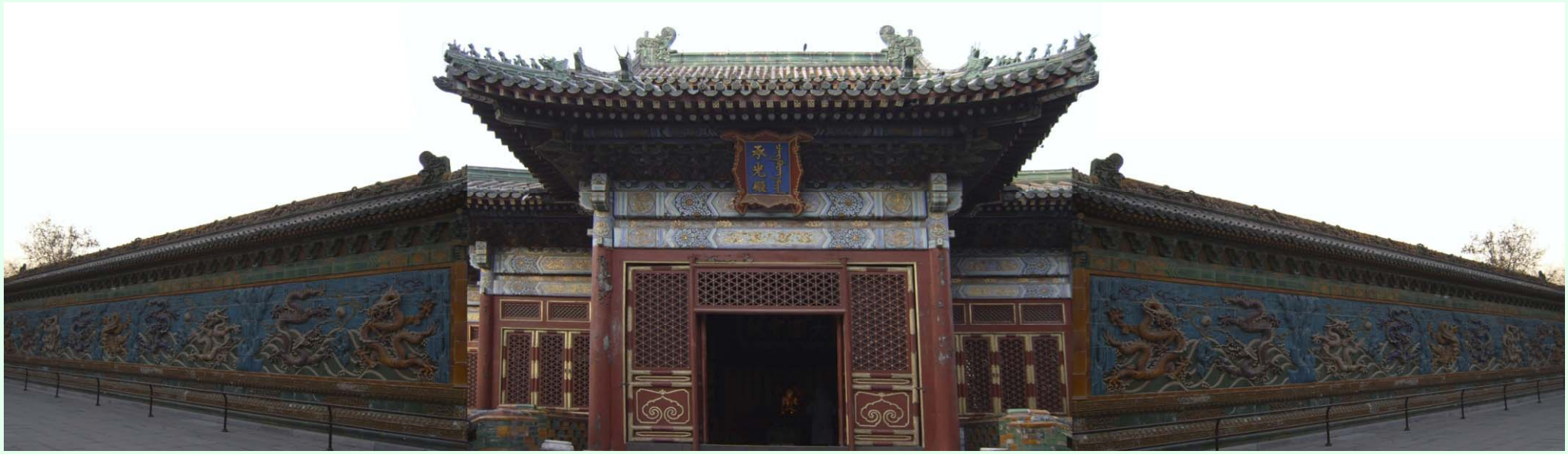
# Application to Forgery Making (3)



| | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 8 | 7 | 11 | 18 | 11 | 10 | 15 | 10 | 8 | 11 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 8 | 53 | 32 | 33 | 46 | 34 | 33 | 33 | 25 | 40 | 50 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 4 | 42 | 88 | 98 | 101 | 65 | 75 | 117 | 92 | 70 | 21 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 2 | 1 | 3 | 4 | 4 | 7 | 11 | 39 | 35 | 26 | 30 | 35 | 35 | 29 | 32 | 48 | 29 | 7 | 4 | 4 | 5 | 3 | 1 | 1 | 3 |
| 5 | 4 | 6 | 6 | 6 | 5 | 4 | 8 | 32 | 27 | 47 | 47 | 52 | 53 | 45 | 31 | 40 | 27 | 4 | 7 | 5 | 4 | 4 | 4 | 5 | 5 |
| 7 | 7 | 6 | 5 | 5 | 5 | 8 | 10 | 41 | 32 | 52 | 61 | 260 | 231 | 56 | 81 | 89 | 113 | 5 | 8 | 6 | 5 | 5 | 4 | 7 | 6 |
| 5 | 5 | 7 | 3 | 9 | 4 | 6 | 7 | 40 | 44 | 87 | 50 | 222 | 281 | 50 | 51 | 67 | 81 | 5 | 7 | 3 | 5 | 6 | 4 | 8 | 6 |

Fig. *F* -map of gamma-corrected forgery. (r = 1.5)

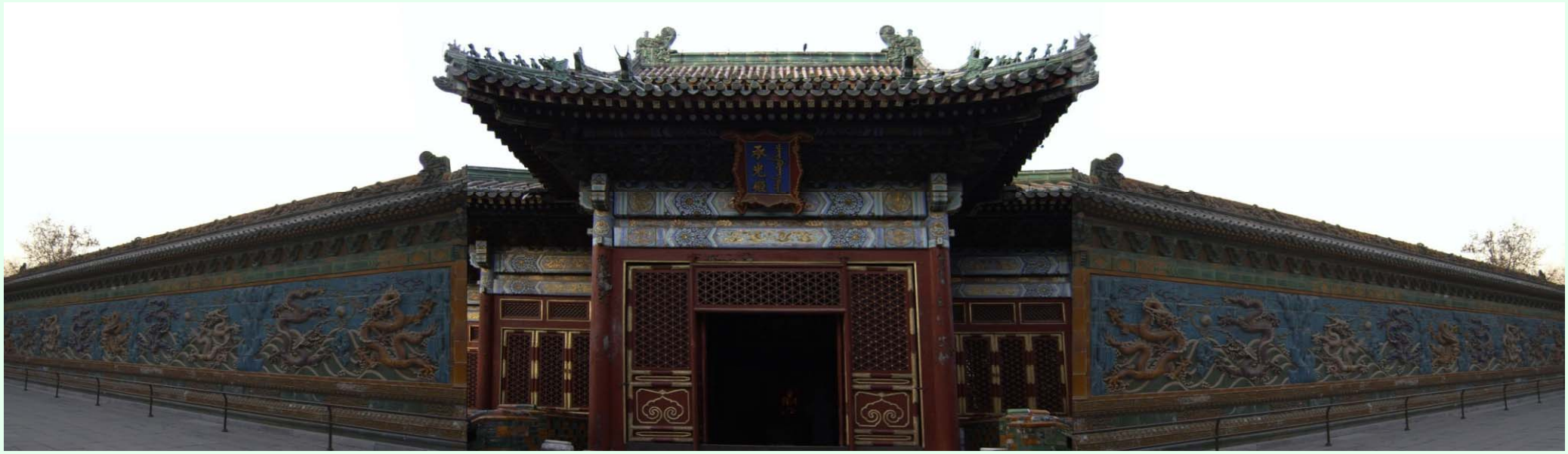
# Application to Forgery Making (3)



| | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 6 | 8 | 7 | 11 | 18 | 11 | 10 | 15 | 10 | 8 | 11 | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 13 | 53 | 32 | 33 | 46 | 34 | 33 | 33 | 25 | 40 | 50 | 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 2 | 23 | 42 | 88 | 98 | 101 | 65 | 75 | 117 | 92 | 70 | 21 | 14 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 4 | 5 | 23 | 38 | 45 | 65 | 53 | 39 | 35 | 26 | 30 | 35 | 35 | 29 | 32 | 48 | 29 | 78 | 56 | 38 | 31 | 15 | 3 | 2 | 9 |
| 46 | 77 | 100 | 126 | 135 | 184 | 203 | 132 | 32 | 27 | 47 | 47 | 52 | 53 | 45 | 31 | 40 | 27 | 36 | 41 | 57 | 86 | 83 | 68 | 64 | 43 |
| 195 | 174 | 168 | 122 | 162 | 158 | 123 | 97 | 41 | 32 | 52 | 61 | 260 | 231 | 56 | 81 | 89 | 113 | 25 | 49 | 27 | 24 | 31 | 29 | 26 | 38 |
| 52 | 71 | 108 | 119 | 116 | 99 | 96 | 89 | 40 | 44 | 87 | 50 | 222 | 281 | 50 | 51 | 67 | 81 | 51 | 63 | 54 | 47 | 44 | 35 | 42 | 63 |

Fig. Attacked forgery. (universal forging attack, random LSB, s: 0.1) PSNR: 71.0dB

# Application to Forgery Making (4)



| | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 14 | 14 | 10 | 9 | 13 | 13 | 11 | 11 | 12 | 12 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 5 | 3 | 4 | 5 | 6 | 5 | 6 | 5 | 4 | 6 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 5 | 8 | 10 | 7 | 8 | 4 | 7 | 10 | 3 | 12 | 6 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 2 | 1 | 3 | 4 | 4 | 7 | 8 | 4 | 4 | 6 | 6 | 5 | 5 | 8 | 4 | 5 | 5 | 7 | 4 | 4 | 5 | 3 | 1 | 1 | 3 |
| 5 | 4 | 6 | 6 | 6 | 5 | 4 | 6 | 6 | 4 | 7 | 9 | 7 | 5 | 6 | 5 | 7 | 4 | 4 | 7 | 5 | 4 | 4 | 4 | 5 | 5 |
| 7 | 7 | 6 | 5 | 5 | 5 | 8 | 6 | 7 | 6 | 8 | 6 | 5 | 11 | 7 | 4 | 5 | 3 | 5 | 8 | 6 | 5 | 5 | 4 | 7 | 6 |
| 5 | 5 | 7 | 3 | 9 | 4 | 6 | 7 | 8 | 4 | 4 | 8 | 13 | 5 | 7 | 12 | 4 | 6 | 5 | 7 | 3 | 5 | 6 | 4 | 8 | 6 |

Fig. Attacked forgery. (integrated hiding attack, dithering-based CE) PSNR: 56.6dB

# Conclusion & Remarks

## ➢ Main Contribution

- Forensics algorithms design for detecting Median filtering, Gamma correction, Blur/Sharpening, composition.
- Anti-forensics of contrast enhancement

## ➢ Further Work

- Algorithm Design:
  - ✓ improve reliability & security of algorithms
  - ✓ counter-counter-…. forensics (cat-mouse game)
- Theoretical Analysis: forensics behavior model & analysis (game theory).