Digital Image Steganography Using Universal Distortion

Vojtěch Holub, Jessica Fridrich

BINGHAMTON
UNIVERSITY
STATE UNIVERSITY OF NEW YORK

Steganography by Minimizing Distortion

1. Define distortion function $D(\mathbf{X}, \mathbf{Y})$ that measures the statistical impact of changing $\mathbf{X}$ to $\mathbf{Y}$.
2. Payload-limited sender: Given cover image $\mathbf{X} \in \mathcal{I}^{n_1 \times n_2}$, message $\mathbf{m} \in \{0,1\}^k$, and a linear code parity-check matrix $\mathbf{H} \in \mathbb{R}^{k \times n_1 n_2}$:

$$\mathbf{Y} = \arg \min_{\mathbf{HY}=\mathbf{m}} D(\mathbf{X}, \mathbf{Y})$$

3. With syndrome trellis codes (STCs) [Filler et al. TIFS 2011], $E_{P(\mathbf{m})}[D(\mathbf{X}, \mathbf{Y})]$ is close to the minimum distortion determined by the rate–distortion bound.

Distortion function

- **Additive**
  - embedding changes do not interact
  - D is sum of pixel costs:

$$D(\mathbf{X}, \mathbf{Y}) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \rho_{ij}(\mathbf{X}, Y_{ij})$$

  - All bits embedded at once using STCs
- **Non-additive**
  - embedding changes interact
  - D is not sum of costs (there are no pixel costs)
  - Message embedded by applying STCs iteratively on interleaved sublattices (Gibbs construction [Filler et al. TIFS 2011])

Additive approximation

- Given an arbitrary (non-additive) distortion $D(\mathbf{X}, \mathbf{Y})$, one can embed with its **additive approximation** by defining the cost of changing pixel i, j as the distortion between $\mathbf{X}$ and $\mathbf{X}$ in which the only changed pixel is $X_{ij} \to Y_{ij}$:

$$\rho_{ij} \triangleq D(\mathbf{X}, Y_{ij}\mathbf{X}_{\sim ij})$$

- In our work, we introduce a general non-additive D but work with their additive approximations in all embedding domains.

Embedding Domains of Interest

- **Spatial domain** – modifying pixel values
  - HUGO [Pevný et al. IH 2010]
  - WOW [Holub et al. WIFS 2012]
- **JPEG domain** – modifying quantized DCT coefficients
  - nsF5 [Westfeld IH 2001, Fridrich et al. ACM MMSec 2007]
  - UED [Guo et al. WIFS 2012]
- **JPEG domain with side information** – raw image (unquantized DCT coef.) available
  - NPQ [Huang et al. TIFS 2012]
  - EBS [Wang et al. ICASSP 2012]

Goal

Design a single universal distortion function that works in ALL domains

UNIWARD – **UNI**versal **WA**velet **R**elative **D**istortion

UNIWARD (non side-informed)

- **Main ingredient:** Filter bank $\mathcal{B} = \left\{\mathbf{K}^{(1)}, \mathbf{K}^{(2)}, \mathbf{K}^{(3)}\right\}$ consisting of the LH, HL, and HH directional high-pass filters (Daubechies wavelets)

UNIWARD (non side-informed)

- **Main ingredient:** Filter bank $\mathcal{B} = \left\{\mathbf{K}^{(1)}, \mathbf{K}^{(2)}, \mathbf{K}^{(3)}\right\}$ consisting of the LH, HL, and HH directional high-pass filters (Daubechies wavelets)
- k-th directional residual (undecimated wavelet sub-band): $W^{(k)}(\mathbf{X}) = \mathbf{X} * \mathbf{K}^{(k)}$, $W^{(k)}(\mathbf{Y}) = \mathbf{Y} * \mathbf{K}^{(k)}$

UNIWARD (non side-informed)

- **Main ingredient:** Filter bank $\mathcal{B} = \left\{\mathbf{K}^{(1)}, \mathbf{K}^{(2)}, \mathbf{K}^{(3)}\right\}$ consisting of the LH, HL, and HH directional high-pass filters (Daubechies wavelets)
- k-th directional residual (undecimated wavelet sub-band): $W^{(k)}(\mathbf{X}) = \mathbf{X} * \mathbf{K}^{(k)}$, $W^{(k)}(\mathbf{Y}) = \mathbf{Y} * \mathbf{K}^{(k)}$
- UNIWARD distortion function defined as the relative distortion

$$D(\mathbf{X}, \mathbf{Y}) = \sum_{k=1}^{3} \sum_{u,v} \frac{|W_{uv}^{(k)}(\mathbf{X}) - W_{uv}^{(k)}(\mathbf{Y})|}{\varepsilon + |W_{uv}^{(k)}(\mathbf{X})|}$$

where the sum over uv is taken over all subband coefficients, $\varepsilon > 0$ is a stabilizing constant to avoid dividing by zero.

UNIWARD (non side-informed)

- **Main ingredient:** Filter bank $\mathcal{B} = \left\{\mathbf{K}^{(1)}, \mathbf{K}^{(2)}, \mathbf{K}^{(3)}\right\}$ consisting of the LH, HL, and HH directional high-pass filters (Daubechies wavelets)
- k-th directional residual (undecimated wavelet sub-band): $W^{(k)}(\mathbf{X}) = \mathbf{X} * \mathbf{K}^{(k)}$, $W^{(k)}(\mathbf{Y}) = \mathbf{Y} * \mathbf{K}^{(k)}$
- UNIWARD distortion function defined as the relative distortion

$$D(\mathbf{X}, \mathbf{Y}) = \sum_{k=1}^{3} \sum_{u,v} \frac{|W_{uv}^{(k)}(\mathbf{X}) - W_{uv}^{(k)}(\mathbf{Y})|}{\varepsilon + |W_{uv}^{(k)}(\mathbf{X})|}$$

  where the sum over uv is taken over all subband coefficients, $\varepsilon > 0$ is a stabilizing constant to avoid dividing by zero.
- D is non-additive $\Rightarrow$ we use its additive approximation

UNIWARD (non side-informed)

- **Main ingredient:** Filter bank $\mathcal{B} = \left\{\mathbf{K}^{(1)}, \mathbf{K}^{(2)}, \mathbf{K}^{(3)}\right\}$ consisting of the LH, HL, and HH directional high-pass filters (Daubechies wavelets)
- k-th directional residual (undecimated wavelet sub-band): $W^{(k)}(\mathbf{X}) = \mathbf{X} * \mathbf{K}^{(k)}$, $W^{(k)}(\mathbf{Y}) = \mathbf{Y} * \mathbf{K}^{(k)}$
- UNIWARD distortion function defined as the relative distortion

$$D(\mathbf{X}, \mathbf{Y}) = \sum_{k=1}^{3} \sum_{u,v} \frac{|W_{uv}^{(k)}(\mathbf{X}) - W_{uv}^{(k)}(\mathbf{Y})|}{\varepsilon + |W_{uv}^{(k)}(\mathbf{X})|}$$

  where the sum over uv is taken over all subband coefficients, $\varepsilon > 0$ is a stabilizing constant to avoid dividing by zero.
- D is non-additive $\Rightarrow$ we use its additive approximation
- Changing the value of any element by $+1$ or -1 causes the same distortion $\Rightarrow$ ternary STCs can be used

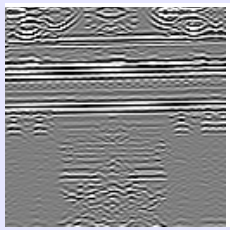
Properties

- Uses a bank of directional filters (Daubechies 8-tap) to obtain multiple *directional residuals*
- Measures the embedding impact for every direction
- The embedding cost is small only if the local content is unpredictable in every direction – embedding avoids smooth edges, embeds mostly to highly textured areas

cover image $\mathbf{X}$

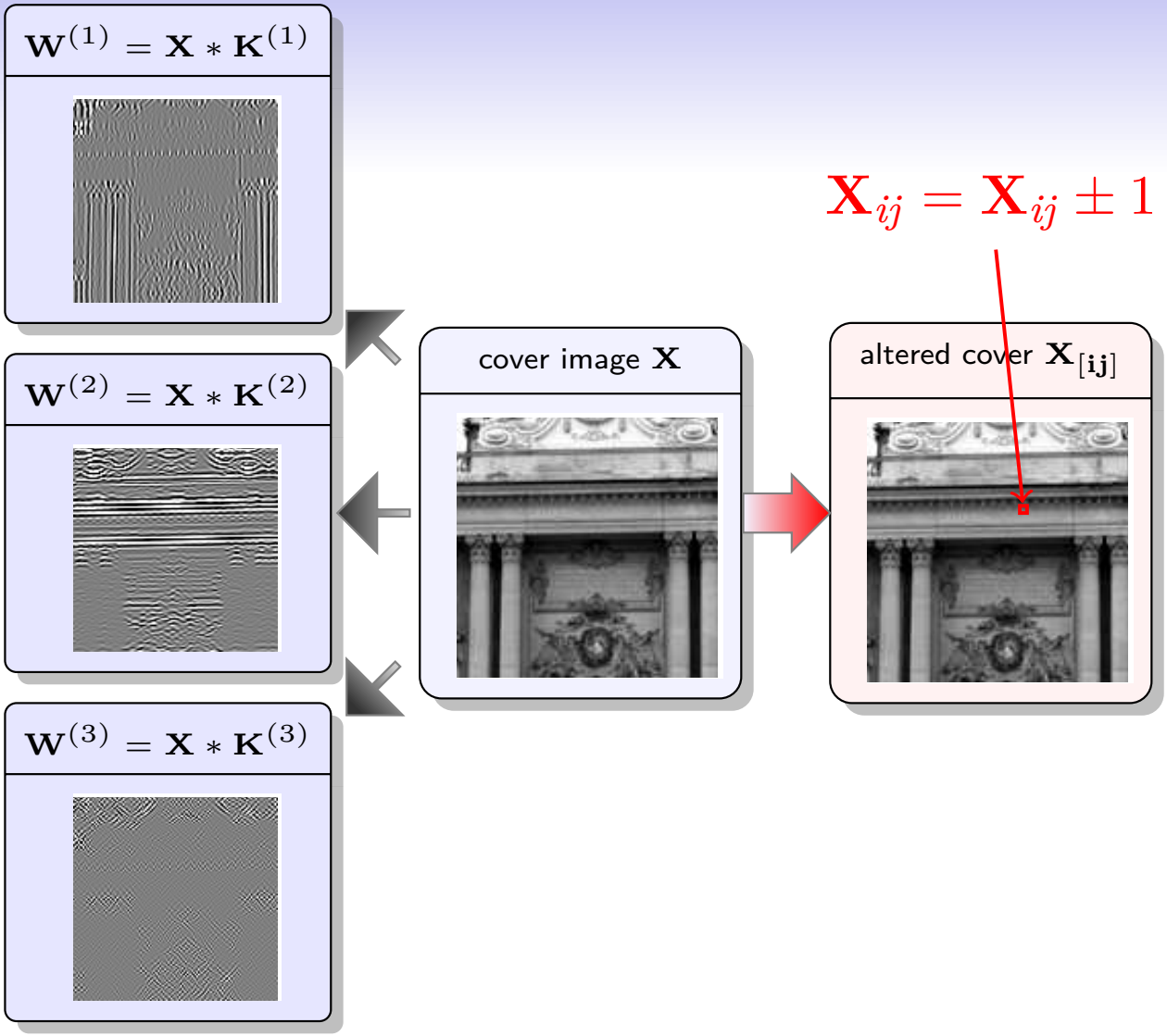
$\mathbf{W}^{(1)} = \mathbf{X} * \mathbf{K}^{(1)}$

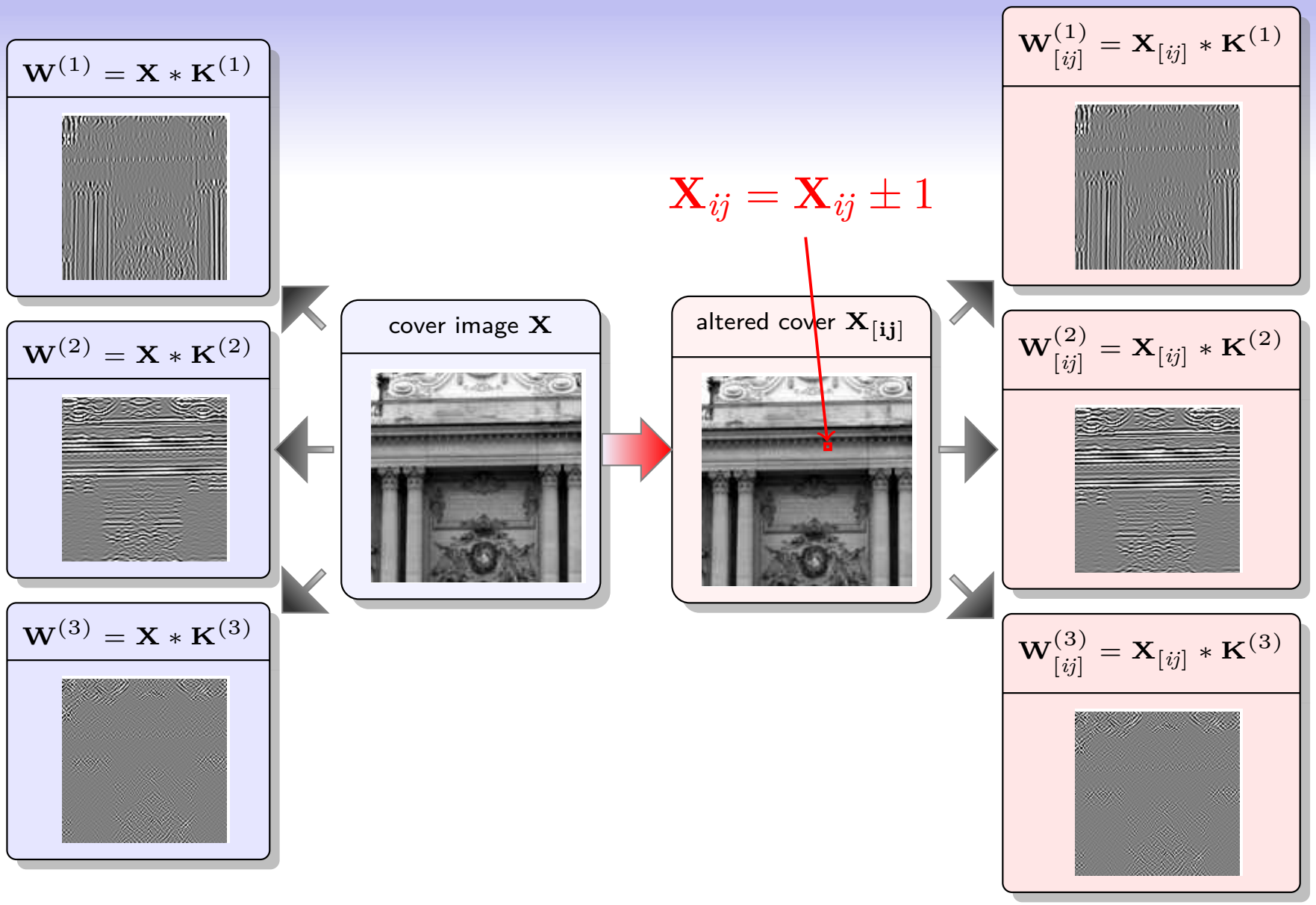
$\mathbf{W}^{(2)} = \mathbf{X} * \mathbf{K}^{(2)}$

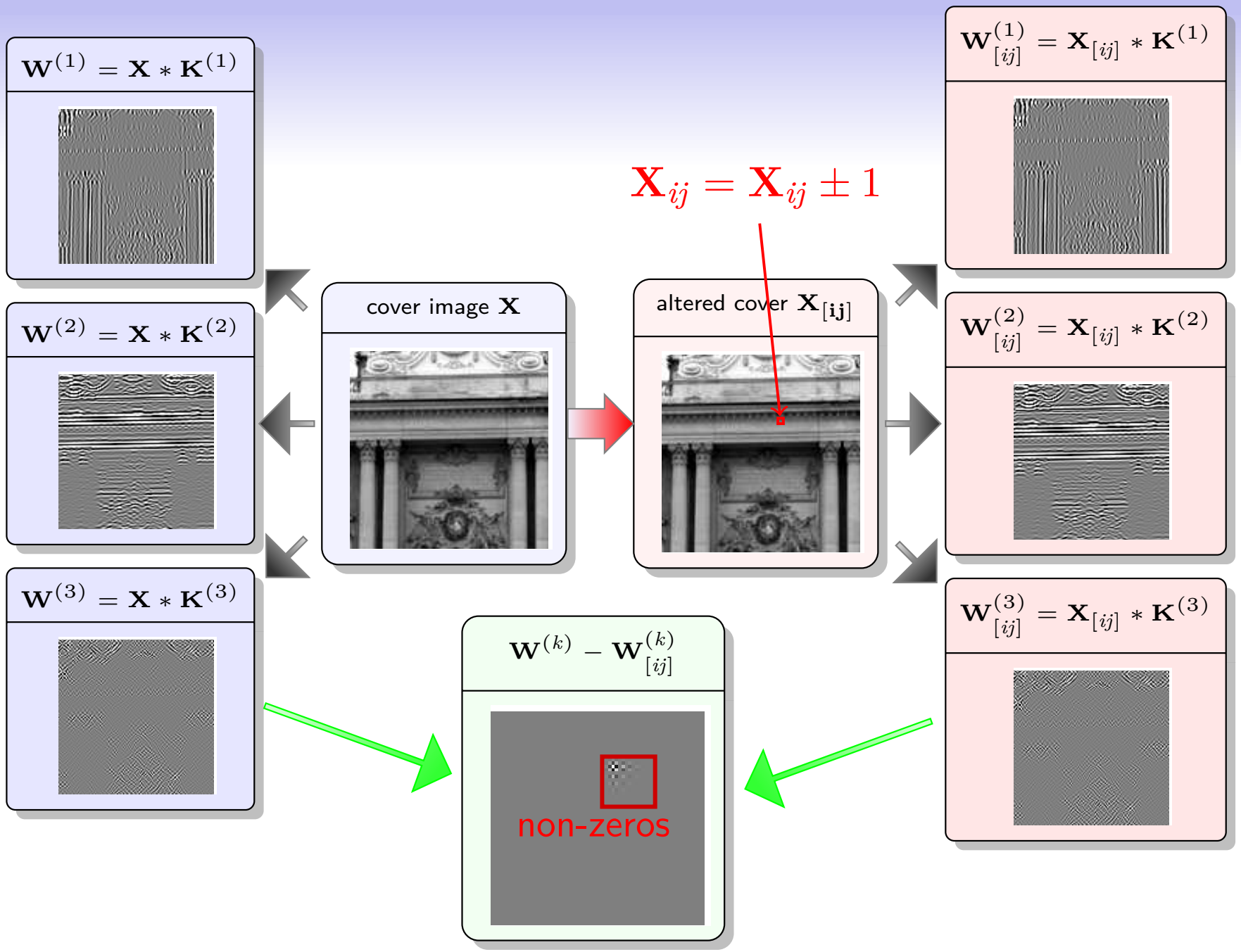


$\mathbf{W}^{(3)} = \mathbf{X} * \mathbf{K}^{(3)}$

cover image $\mathbf{X}$


$\mathbf{W}^{(1)} = \mathbf{X} * \mathbf{K}^{(1)}$
$\mathbf{W}^{(2)} = \mathbf{X} * \mathbf{K}^{(2)}$
$\mathbf{W}^{(3)} = \mathbf{X} * \mathbf{K}^{(3)}$
cover image $\mathbf{X}$
$\mathbf{X}_{ij} = \mathbf{X}_{ij} \pm 1$
altered cover $\mathbf{X}_{[ij]}$


$\mathbf{W}^{(1)} = \mathbf{X} * \mathbf{K}^{(1)}$
$\mathbf{W}^{(2)} = \mathbf{X} * \mathbf{K}^{(2)}$
$\mathbf{W}^{(3)} = \mathbf{X} * \mathbf{K}^{(3)}$
cover image $\mathbf{X}$
$\mathbf{X}_{ij} = \mathbf{X}_{ij} \pm 1$
altered cover $\mathbf{X}_{[ij]}$
$\mathbf{W}^{(1)}_{[ij]} = \mathbf{X}_{[ij]} * \mathbf{K}^{(1)}$
$\mathbf{W}^{(2)}_{[ij]} = \mathbf{X}_{[ij]} * \mathbf{K}^{(2)}$
$\mathbf{W}^{(3)}_{[ij]} = \mathbf{X}_{[ij]} * \mathbf{K}^{(3)}$


$\mathbf{W}^{(1)} = \mathbf{X} * \mathbf{K}^{(1)}$
$\mathbf{W}^{(2)} = \mathbf{X} * \mathbf{K}^{(2)}$
$\mathbf{W}^{(3)} = \mathbf{X} * \mathbf{K}^{(3)}$
cover image $\mathbf{X}$
$\mathbf{X}_{ij} = \mathbf{X}_{ij} \pm 1$
altered cover $\mathbf{X}_{[ij]}$
$\mathbf{W}^{(1)}_{[ij]} = \mathbf{X}_{[ij]} * \mathbf{K}^{(1)}$
$\mathbf{W}^{(2)}_{[ij]} = \mathbf{X}_{[ij]} * \mathbf{K}^{(2)}$
$\mathbf{W}^{(3)}_{[ij]} = \mathbf{X}_{[ij]} * \mathbf{K}^{(3)}$
$\mathbf{W}^{(k)} - \mathbf{W}^{(k)}_{[ij]}$
non-zeros

Implementation

$$\rho_{ij} = \sum_{k=1}^{3} \sum_{u,v} \frac{|W_{uv}^{(k)} - W_{[ij]uv}^{(k)}|}{\varepsilon + |W_{uv}^{(k)}|} = \sum_{k=1}^{3} \sum_{(u,v)\in\mathcal{N}_{ij}} \frac{|W_{uv}^{(k)} - W_{[ij]uv}^{(k)}|}{\varepsilon + |W_{uv}^{(k)}|}$$

- **Spatial domain**
  - UNIWARD applied directly
  - Changing one pixel affects its 16×16 neighborhood $\mathcal{N}_{ij}$ (the support of Daubechies 8-tap 2-D filters)
  - $\Rightarrow$ embedding costs can be computed using convolution
- **JPEG domain**
  - $\mathbf{X}$, $\mathbf{Y}$ decompressed to spatial domain
  - then wavelet coefficients computed
  - Chaning one DCT coefficient affects a 23×23 neighborhood $\mathcal{N}_{ij}$ $(23 = 16 + 8 - 1)$

Experiments

Common setup:

- BOSSbase image database (10,000 512×512 grayscale images)
- Ensemble classifier, reported error is its out-of-bag estimate E_{OOB}

Spatial domain setup:

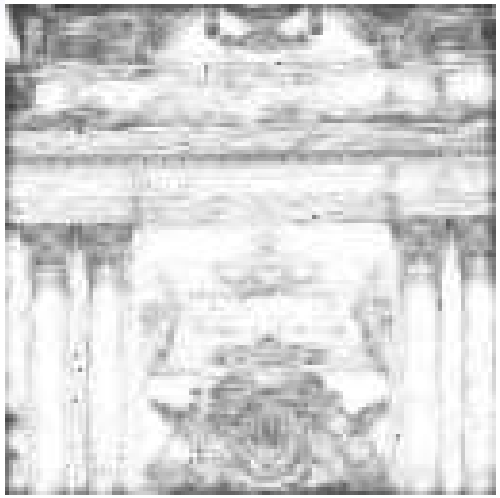
- Spatial Rich Model feature set (SRM), dim $=$ 34,671

JPEG setup:

- Quality factors 75, 85, 95
- JPEG Rich Model + SRMQ1 (JSRM), dim $=$ 35,263

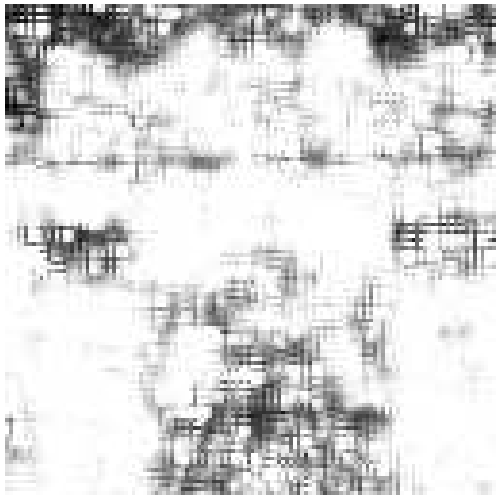
Content adaptivity of S-UNIWARD

Cover

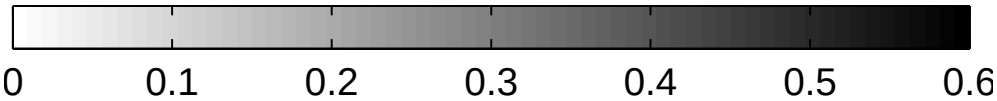


HUGO

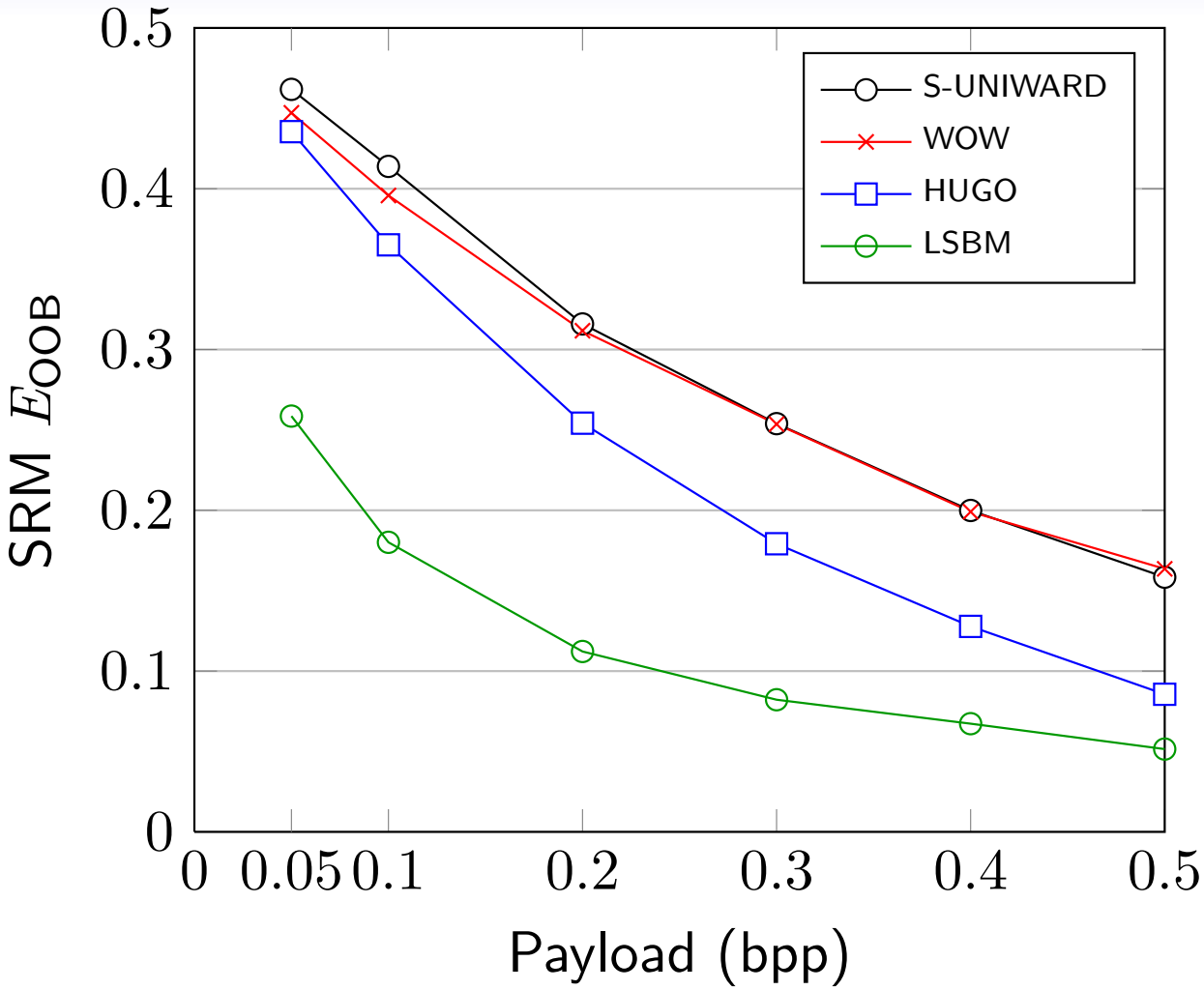
WOW



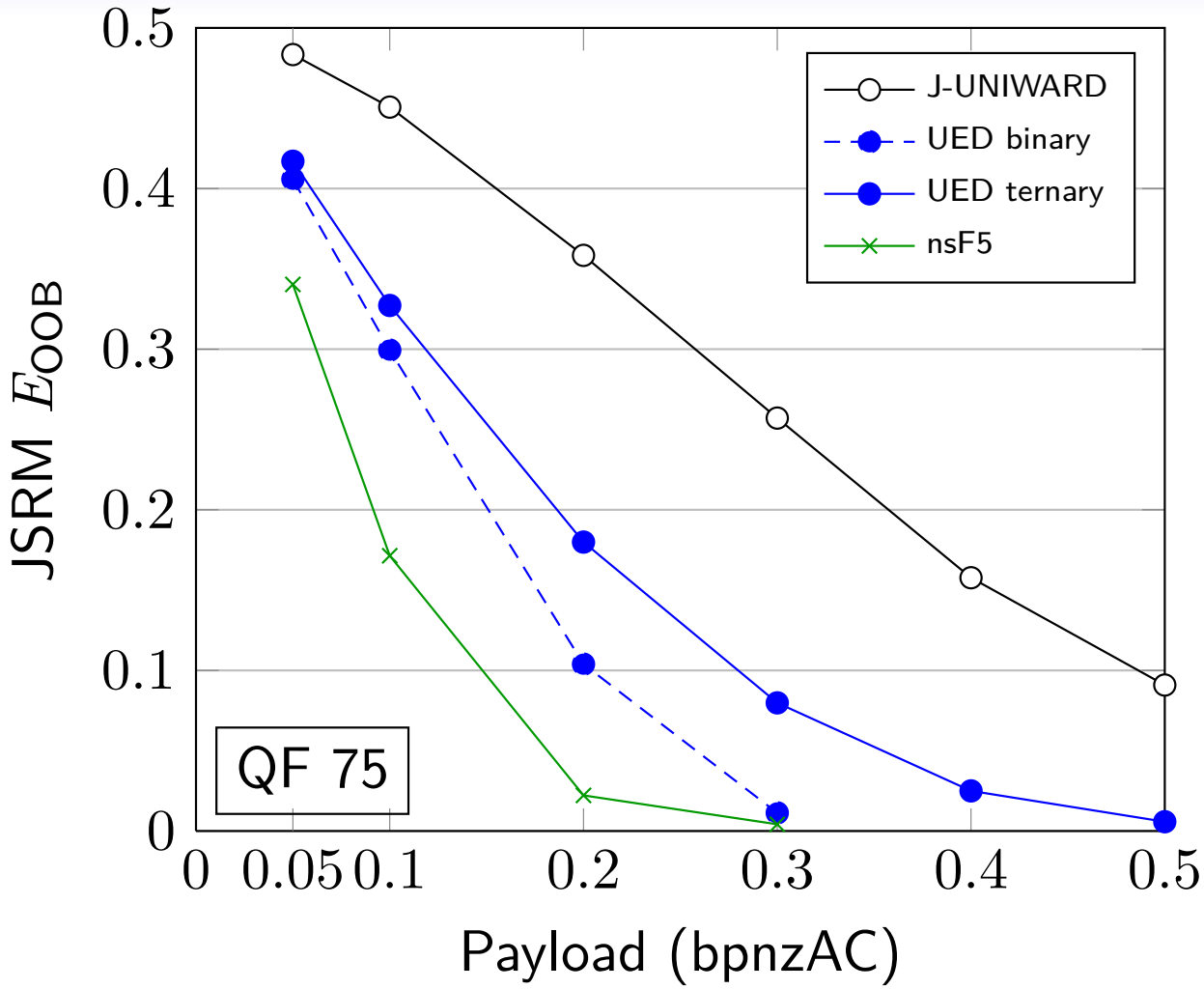
S-UNIWARD



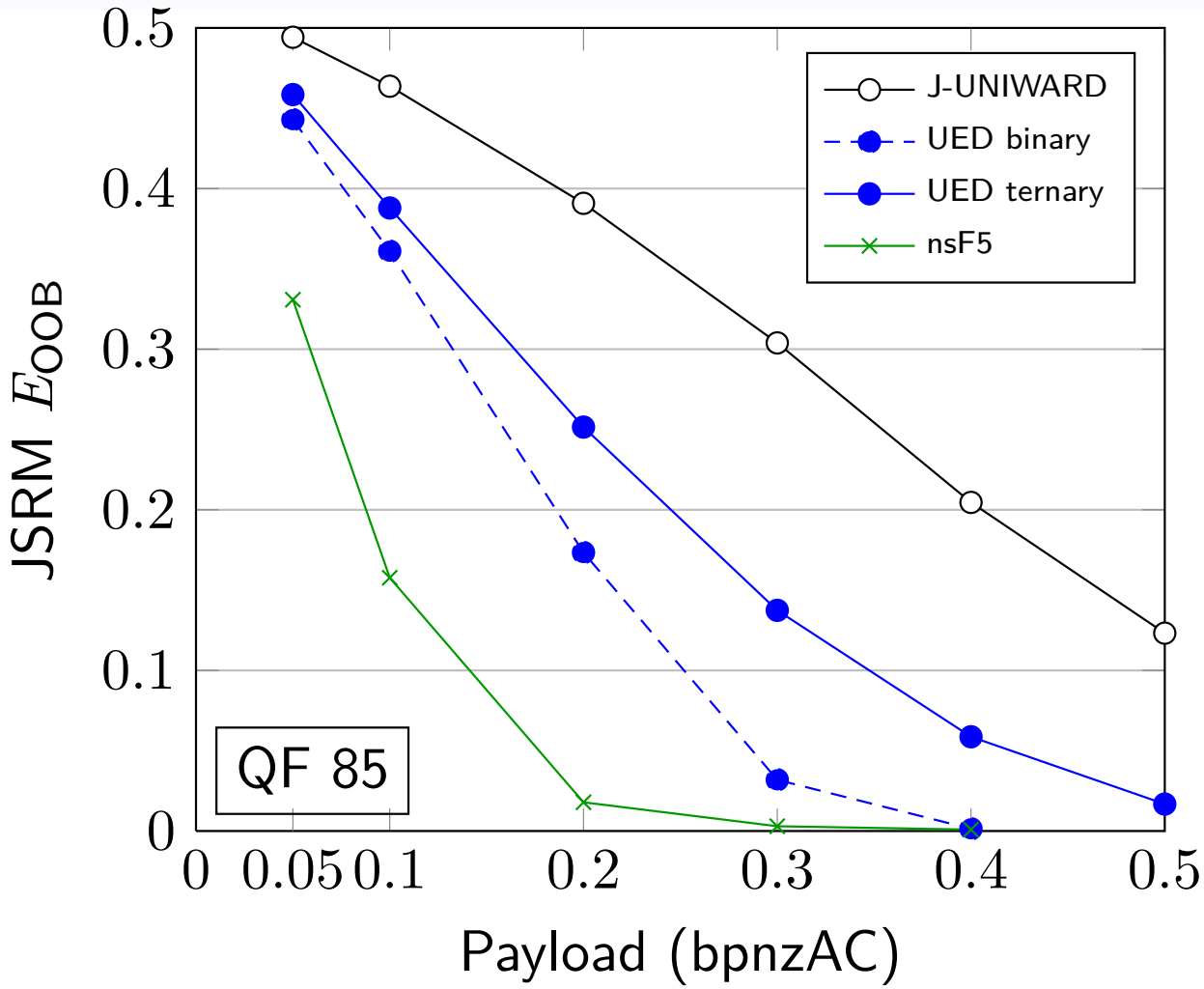
Performance of S-UNIWARD



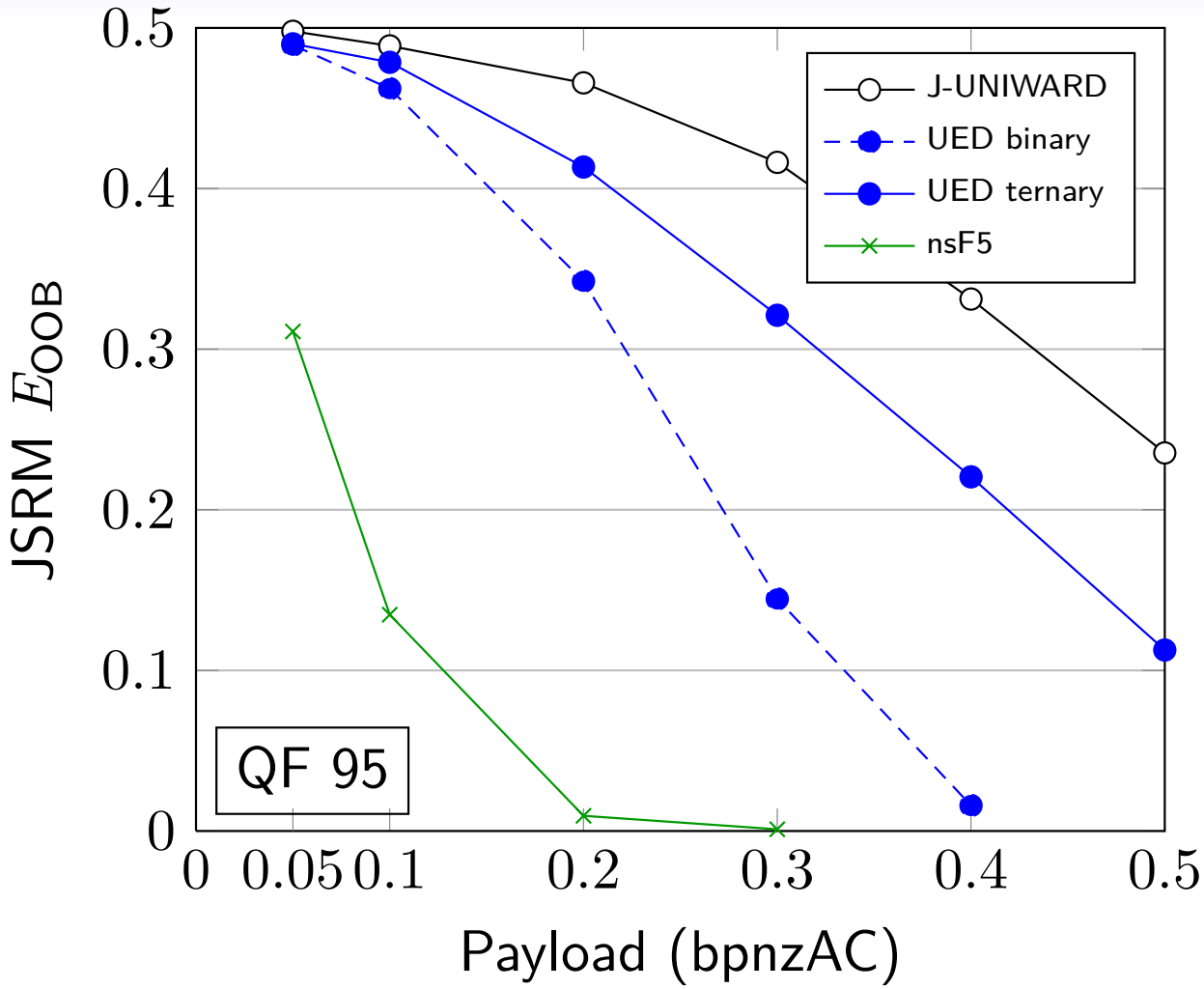
Performance of J-UNIWARD



Performance of J-UNIWARD



Performance of J-UNIWARD



Side-informed JPEG Steganography

General embedding principle:

- $D_{ij} =$ raw DCT coefficients from uncompressed precover $\mathbf{P}$

Side-informed JPEG Steganography

General embedding principle:

- $D_{ij} =$ raw DCT coefficients from uncompressed precover $\mathbf{P}$
- Embedder rounds D_{ij} **up** or **down** to modulate its parity

Side-informed JPEG Steganography

General embedding principle:

- $D_{ij} =$ raw DCT coefficients from uncompressed precover $\mathbf{P}$
- Embedder rounds D_{ij} **up** or **down** to modulate its parity
- $e_{ij} = |D_{ij} - X_{ij}|$, $e_{ij} \in [0, 0.5] =$ rounding error when JPEG compressing $\mathbf{X}$

Side-informed JPEG Steganography

General embedding principle:

- $D_{ij} =$ raw DCT coefficients from uncompressed precover $\mathbf{P}$
- Embedder rounds D_{ij} **up** or **down** to modulate its parity
- $e_{ij} = |D_{ij} - X_{ij}|$, $e_{ij} \in [0, 0.5] =$ rounding error when JPEG compressing $\mathbf{X}$
- Rounding "to the other side" leads to "rounding error" $1 - e_{ij}$

Side-informed JPEG Steganography

General embedding principle:

- $D_{ij} =$ raw DCT coefficients from uncompressed precover $\mathbf{P}$
- Embedder rounds D_{ij} **up** or **down** to modulate its parity
- $e_{ij} = |D_{ij} - X_{ij}|$, $e_{ij} \in [0, 0.5] =$ rounding error when JPEG compressing $\mathbf{X}$
- Rounding "to the other side" leads to "rounding error" $1 - e_{ij}$
- Every embedding change increases the distortion w.r.t. precover by the difference between both rounding errors:

$$|D_{ij} - Y_{ij}| - |D_{ij} - X_{ij}| = 1 - 2e_{ij}$$

UNIWARD (side-informed)

Given:

1. Precover (uncompressed) image $\mathbf{P}$
2. Compressed cover $\mathbf{X}$

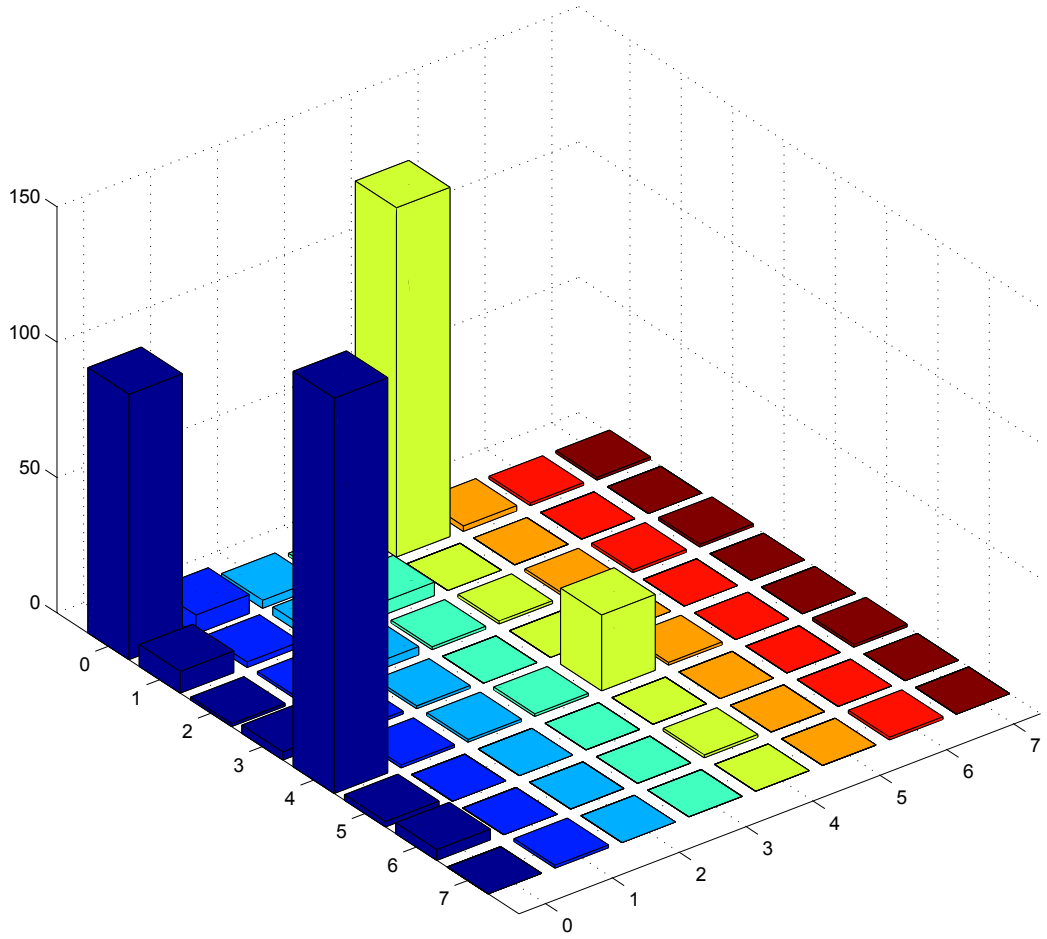
Side-informed UNIWARD (SI-UNIWARD) is defined as the difference:

$$D^{(\mathrm{SI})}(\mathbf{X},\mathbf{Y}) \triangleq D(\mathbf{P},\mathbf{Y}) - D(\mathbf{P},\mathbf{X})$$

$$= \sum_{k=1}^{3}\sum_{u,v} \frac{|W_{uv}^{(k)}(\mathbf{P}) - W_{uv}^{(k)}(\mathbf{Y})| - |W_{uv}^{(k)}(\mathbf{P}) - W_{uv}^{(k)}(\mathbf{X})|}{\varepsilon + |W_{uv}^{(k)}(\mathbf{P})|}$$

Issue With Side-informed Embedding – Observation

Histogram of embedding changes for QF 95, payload 0.01 bpnzAC



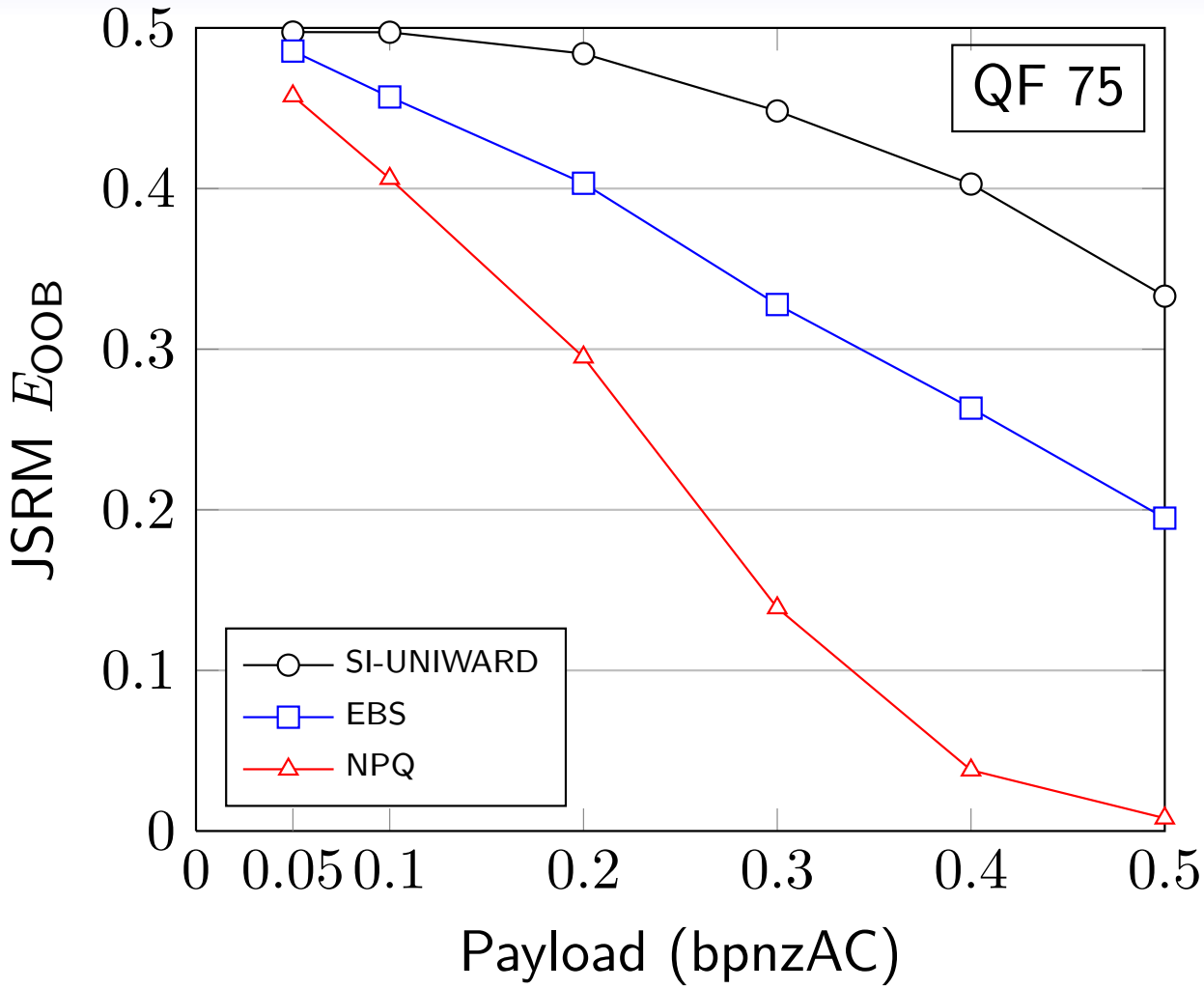
Issue With Side-informed Embedding – Cause

- When $D_{ij} = k + 0.5$, $k \in \mathbb{Z}$, the cost of embedding change is **zero**
  - Should not be zero as we are changing the cover value
  - Applies to all side-informed schemes, not just SI-UNIWARD
- The number of coefficients equal to $k + 0.5$ increases with
  - quality factor
  - fast integer DCT (as opposed to slow DCT)
- Unrounded DCT modes 00 (DC), 40, 04 and 40 are always rational numbers!
  - with smaller payload only these four modes with $e_{ij} = 0.5$ are changed – **detectable by JRM**

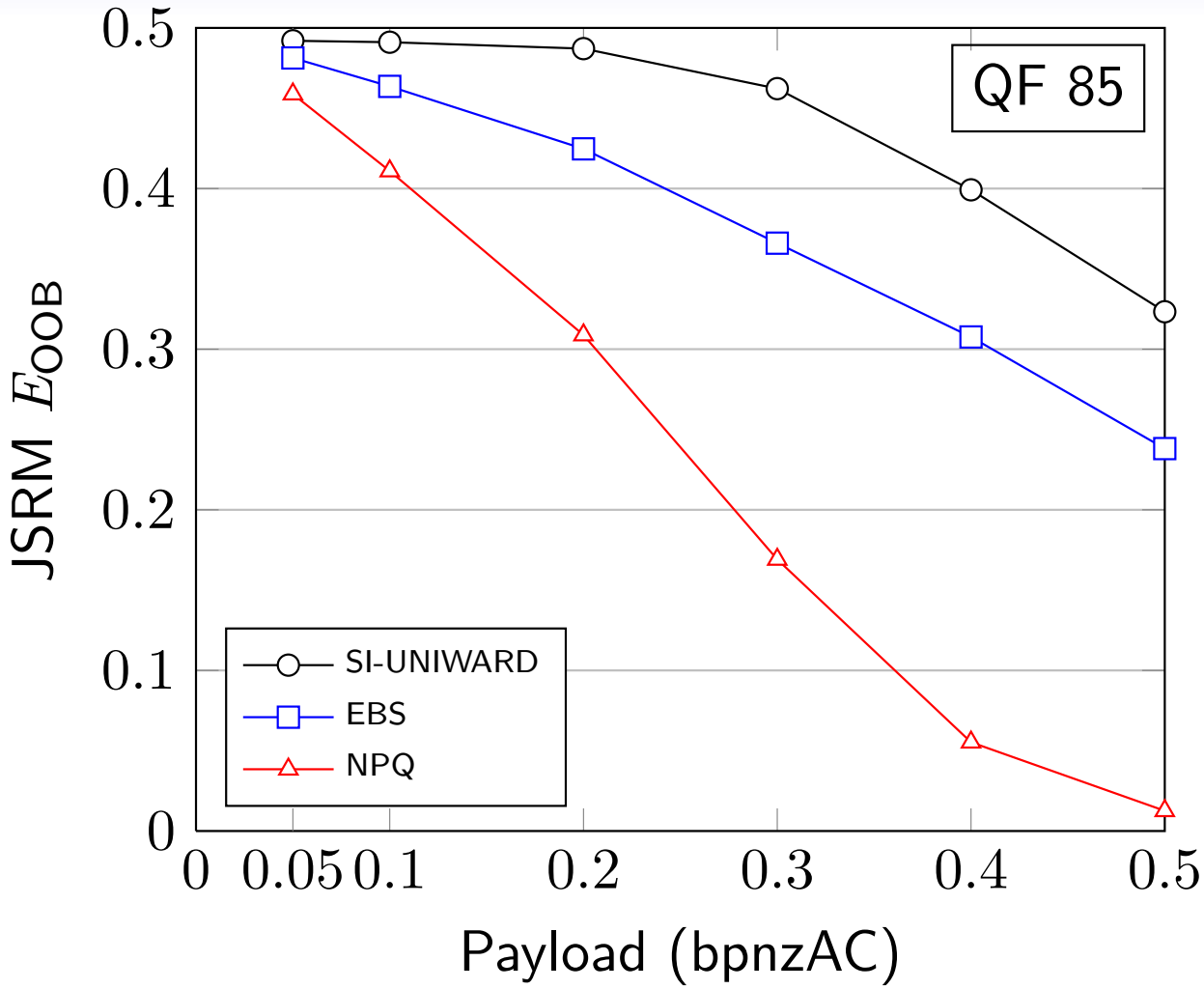
Temporary Fix

- For DCT modes 00, 04, 40, 44 with $e = 0.5$ instead of zero cost, assign a very large cost
- Clearly suboptimal but effective
- More fundamental problem: "How to use side-information in an optimal way?" (future direction)

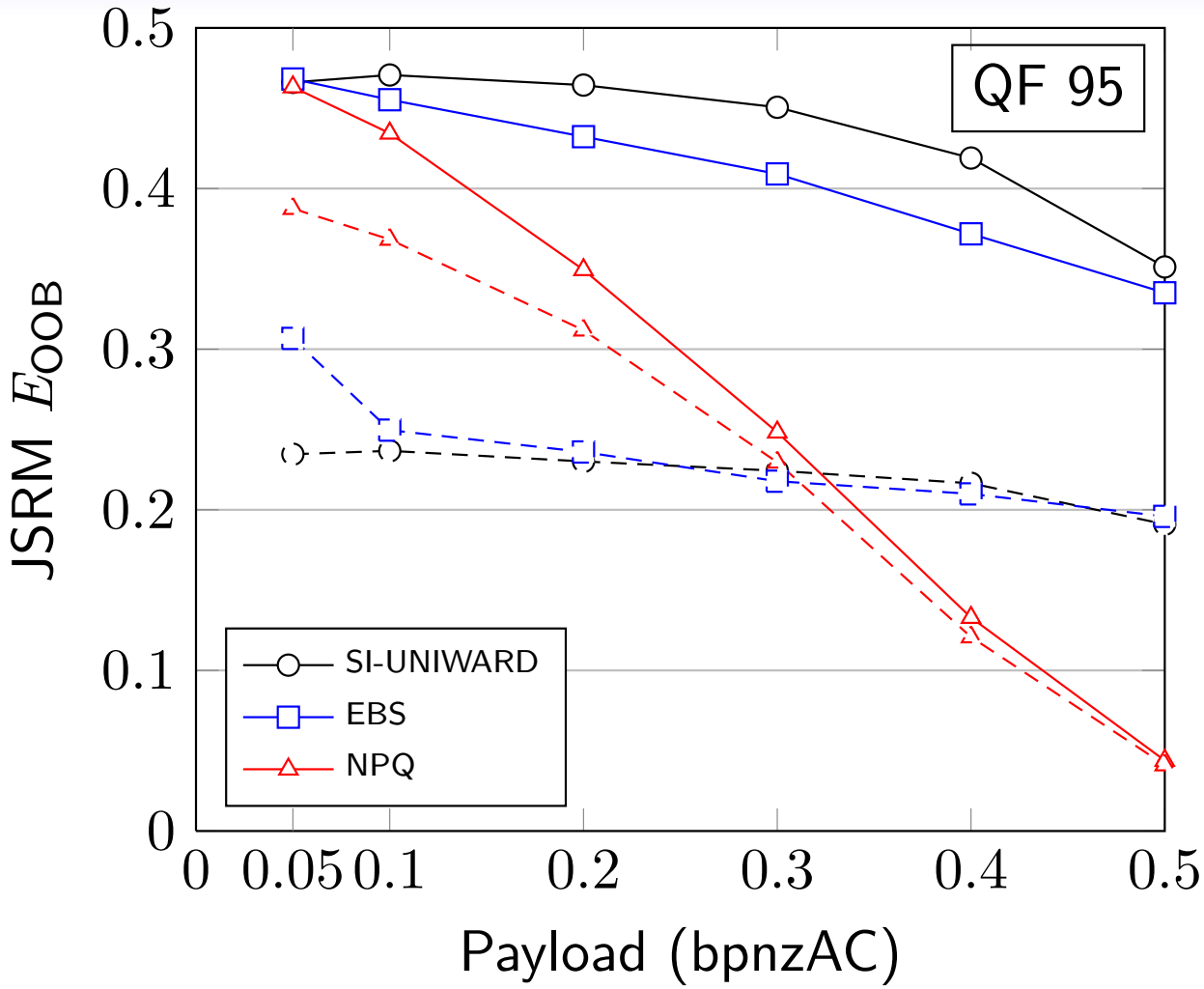
Performance of SI-UNIWARD



Performance of SI-UNIWARD



Performance of SI-UNIWARD



Summary

- UNIWARD = parameter-free, clean, and universal embedding distortion, independent of the embedding domain.
- In all three tested domains, the proposed steganographic schemes outperformed the current state-of-the-art steganographic methods when tested using classifiers with rich models.
- Discovered a new phenomenon that hampers the performance of side-informed JPEG steganography for higher quality factors (deeper problem with the role of side-info in steganography).

Matlab, MEX and C++ source code is available at
`http://dde.binghamton.edu/download/stego_algorithms/`