JPEG-Compatibility Steganalysis Using Block-Histogram of Recompression Artifacts

Jan Kodovský, Jessica Fridrich

May 16, 2012 / IH Conference

BINGHAMTON
UNIVERSITY
STATE UNIVERSITY OF NEW YORK

What is JPEG-compatibility steganalysis?

- Detects embedding changes in the spatial domain using the fact that the cover image was previously JPEG compressed
- All pixels in a cover image must be obtainable by decompressing the corresponding quantized DCT coefficients

Relevancy

- Photographs are commonly stored in the JPEG format
- JPEG format may not allow sufficient capacity
- Vast majority of publicly available steganographic tools hide messages in raster formats
- Most steganographic algorithms do not take JPEG compatibility into account

Overview of the process

Step 1

- Estimate JPEG compression parameters θ from image $\mathbf{Y}$
- $\mathbf{Y}$ ... (stego) image in raster format
- θ ... JPEG quality factor (quantization table), DCT specifics, chrominance tables, color subsampling, etc.

Step 2

- Detect embedding changes using the recompressed image $\hat{\mathbf{Y}}$
- $\hat{\mathbf{Y}} = \mathrm{JPEG}_\theta^{-1}\left(\mathrm{JPEG}_\theta(\mathbf{Y})\right)$ ... recompressed image (predictor)
- JPEG_θ ... JPEG compression (many-to-one mapping)
- $\mathrm{JPEG}_\theta^{-1}$ ... JPEG decompression

Prior art

Fridrich, Goljan, Du (SPIE 2001)

- Mathematical guarantee of JPEG incompatibility
- Brute-force search, growing complexity for higher quality factors

Böhme (IH 2007)

- Weighted Stego-image for decompressed images (WSJPG)
- Targeted to LSB replacement (LSBR)
- Uniform weights, predictor = recompressed image $\hat{\mathbf{Y}}$

$$\hat{\beta}_{\mathrm{WS}} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y}_i)(y_i - \hat{y}_i)$$

y_i . . . pixel value
$\bar{y}_i$ . . . value of the pixel with flipped LSB
$\hat{y}_i$ . . . cover pixel predictor (image recompression)

Prior art, cont'd

Luo, Wang, Huang (IEEE SPL 2011)

- The same recompression predictor as WSJPG
- Decision based on the number of mismatched pixels

$$\hat{\beta}_{\mathrm{LUO}} = \frac{1}{n} \left| \{ i | y_i \neq \hat{y}_i \} \right|$$

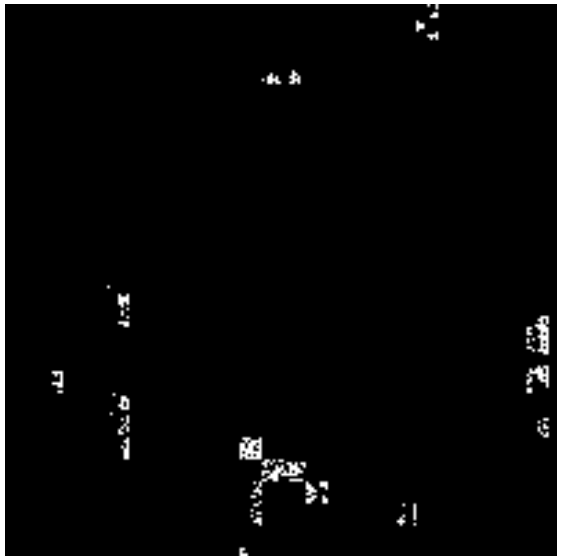
y_i . . . pixel value
$\hat{y}_i$ . . . cover pixel predictor (image recompression)

- Ability to detect embedding operations other than LSBR
- Much less robust to inaccurate estimate of θ
- Does not distinguish between embedding changes and natural recompression artifacts

Image recompression

- BOSSbase image 7347.pgm, JPEG quality factor 80

Decompressed image $\mathbf{Y}$

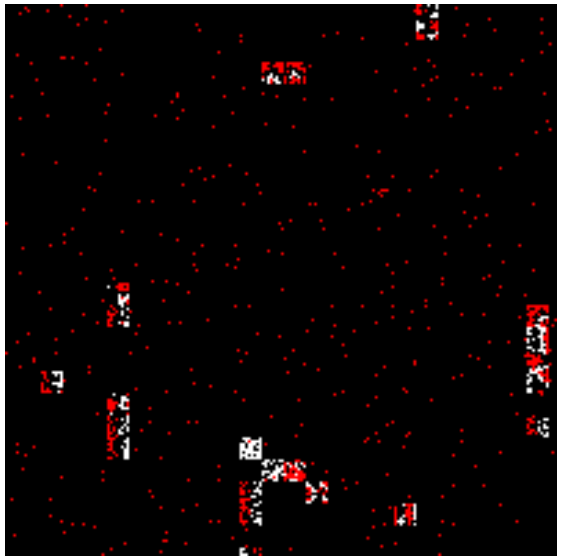


Residual $\mathbf{R} = \mathbf{Y} - \hat{\mathbf{Y}}$

Image recompression

- BOSSbase image 7347.pgm, JPEG quality factor 80

Decompressed image $\mathbf{Y}$



Residual $\mathbf{R} = \mathbf{Y} - \hat{\mathbf{Y}}$

$$\hat{\beta}_{\mathrm{LUO}} = \frac{1}{n}\mathrm{nnz}(\mathbf{R})$$

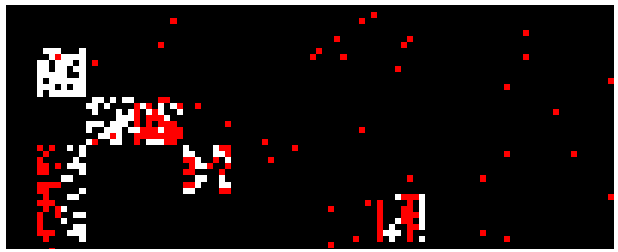
LSBR @ change rate $\beta = 0.01$

Introducing a simple feature vector

- Goal: distinguish between emb. changes and recomp. artifacts
- Recompression artifacts

  . . . patterns over 8×8 pixel blocks
- Embedding changes

  . . . individual changed pixels



Simple pattern descriptor

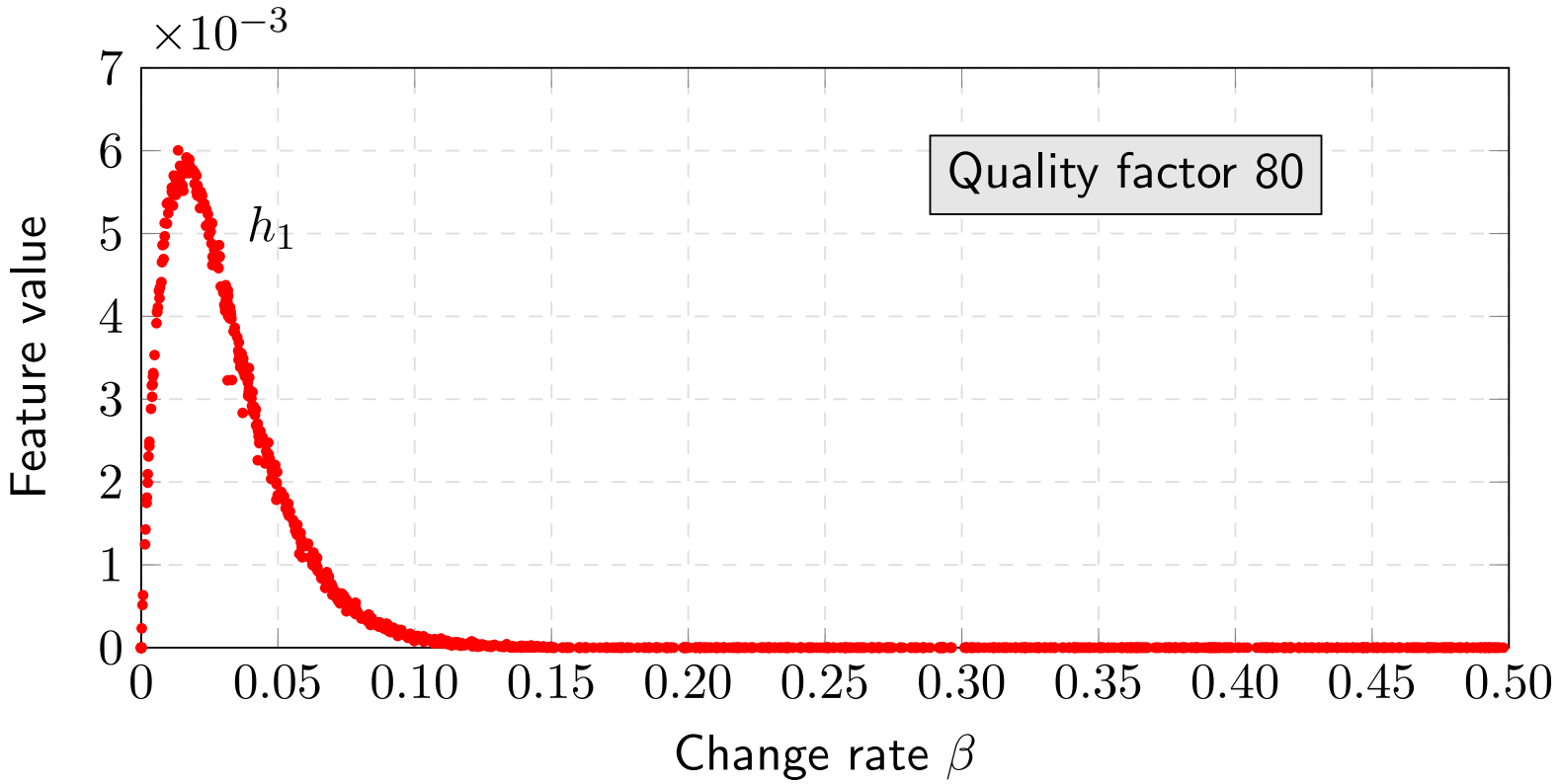
- $\rho^{(k)}$ . . . number of mismatched pixels in the kth block
- $\mathbf{h} = (h_m)$ . . . histogram of $\rho^{(k)}$ over the image

$$h_m = \frac{64}{n} \left| \{k | \rho^{(k)} = m\} \right|, \quad m = 0, \ldots, 64$$

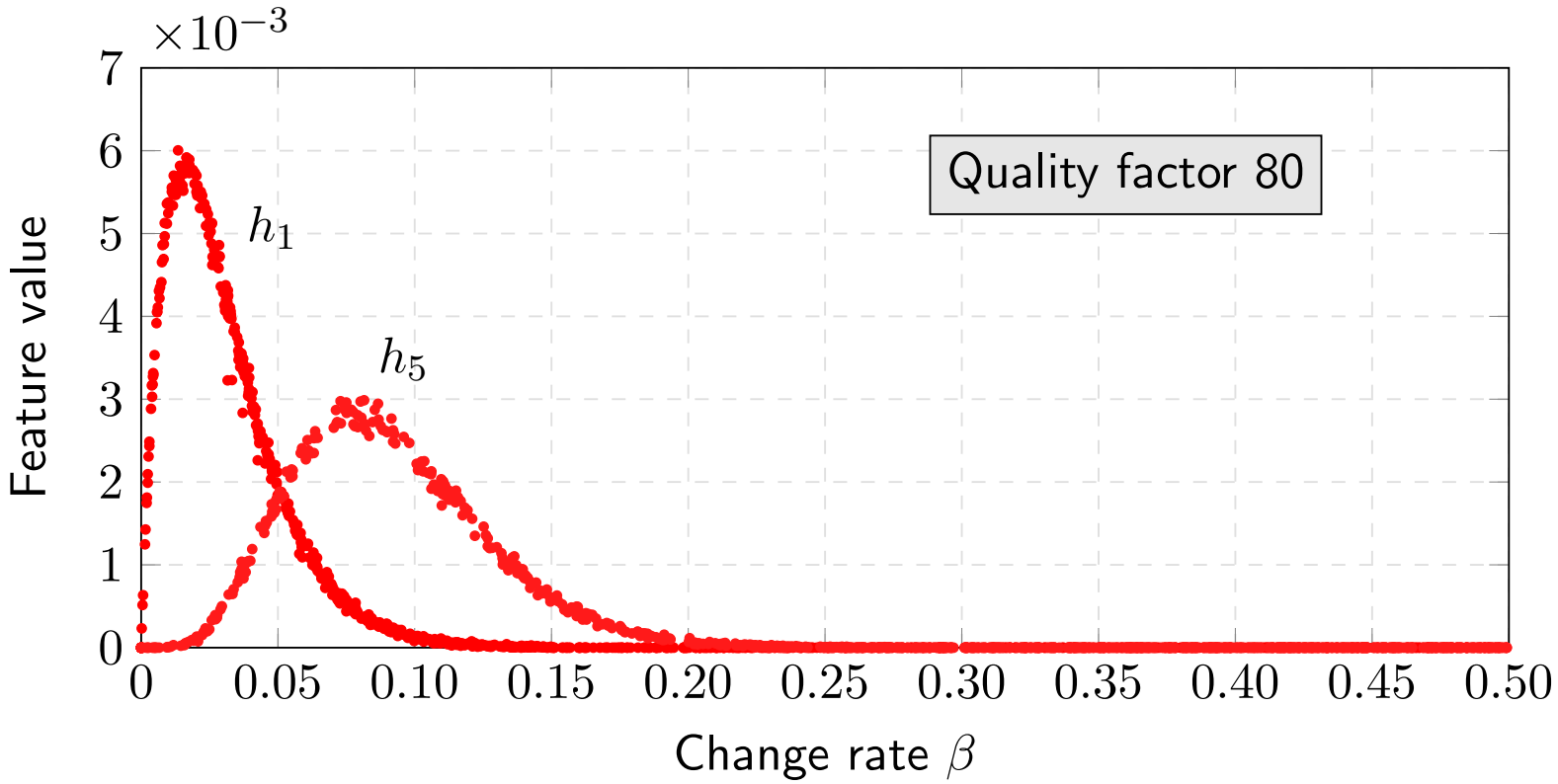
Relationship to the predictor of Luo et al.

$$\hat{\beta}_{\mathrm{LUO}} = \frac{1}{n} \left| \{i | y_i \neq \hat{y}_i\} \right| = \frac{1}{64} \sum_{m=0}^{64} m \cdot h_m$$

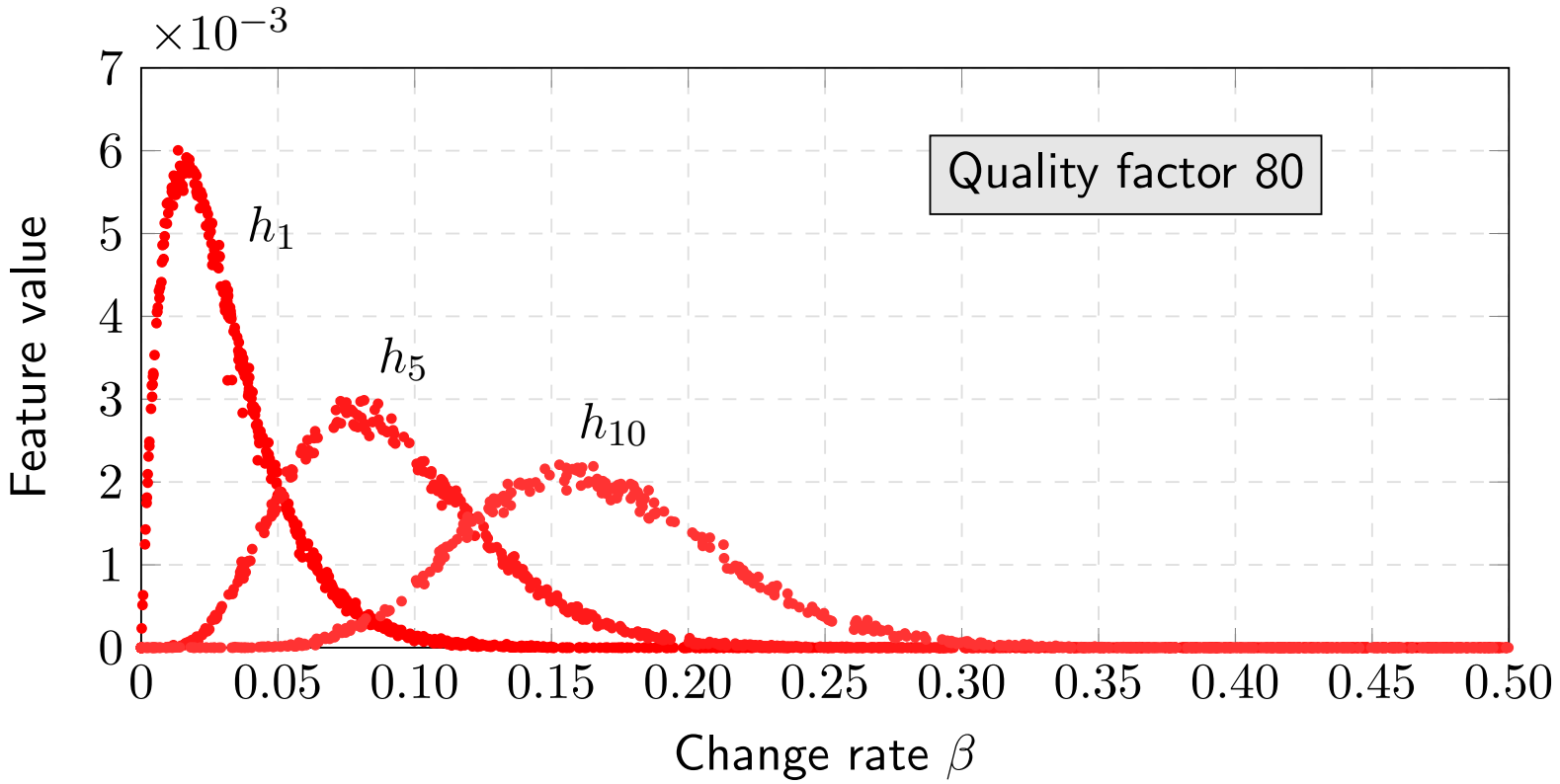
Properties of the proposed features



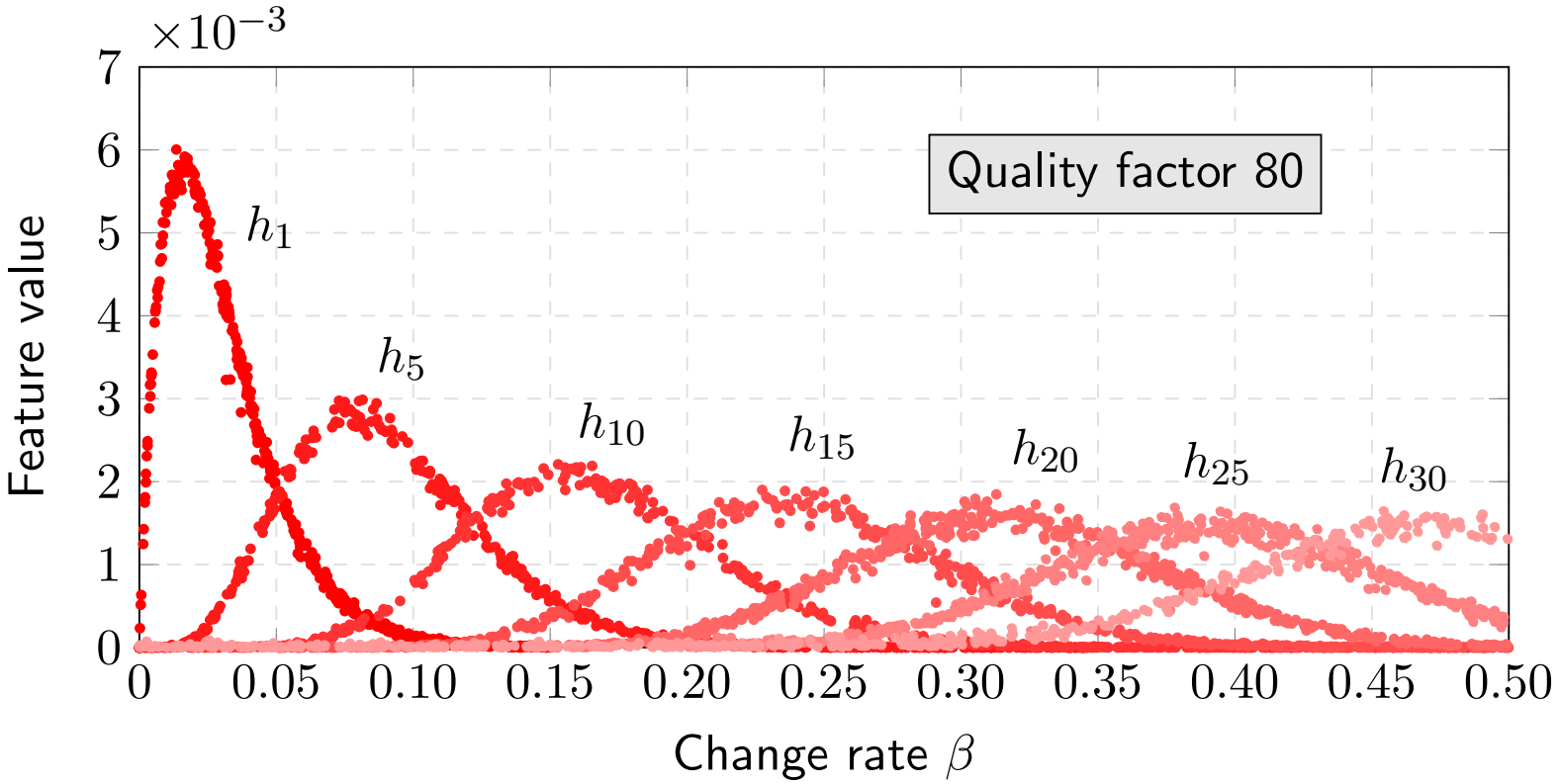
Properties of the proposed features



Properties of the proposed features

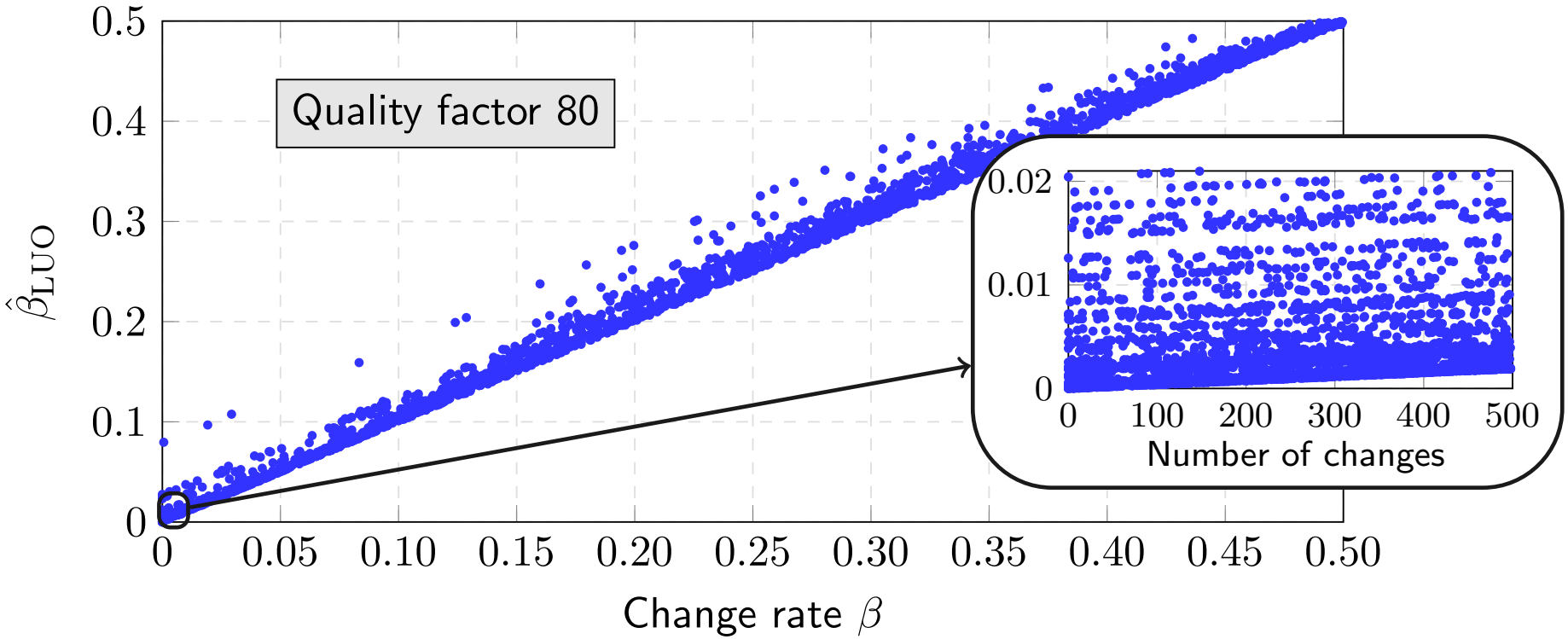


Properties of the proposed features



- Feature vector $\mathbf{h}$ covers all change rates β
- Next step: supply $\mathbf{h}$ to a machine-learning engine

Comparison to Luo et al.



- Large variance due to recompression artifacts
- Inaccurate for detecting small change rates (our focus)

Clairvoyant detector

- Distinguish between cover images and stego images of **known** β
- Tested over a range of different quality factors and change rates
- Our focus: very small change rates (even a single change)
- Binary classifier trained for every tested change rate β
- Ensemble classifier with FLD as a base learner
- Image database: BOSSbase
- Threshold set to minimize total average error under equal priors

$$P_{\mathrm{E}} = \min_{P_{\mathrm{FA}}} \frac{1}{2} \left(P_{\mathrm{FA}} + P_{\mathrm{MD}} \right)$$

Clairvoyant detector, cont'd

A single changed pixel ($\beta = 0.0000038$) LSBR

JPEG quality	70	80	85	90	92	94	96	98
WSJPG	.387	.425	.470	.490	.491	.497	.498	.498
Proposed	0	0	.010	.085	.089	.489	.497	.498

100 changed pixels ($\beta = 0.00038$)

JPEG quality	70	80	85	90	92	94	96	98
WSJPG	.157	.147	.166	.219	.197	.329	.378	.391
Proposed	0	0	0	0	0	.012	.251	.301

262 changed pixels ($\beta = 0.001$)

JPEG quality	70	80	85	90	92	94	96	98
WSJPG	.076	.063	.064	.088	.072	.148	.215	.245
Proposed	0	0	0	0	0	0	.049	.174

Note: For such small values of β, LUO performed consistently worse than WSJPG

Detection of schemes other than LSBR

- LSB matching (± 1 embedding) – similar results as LSBR
- HUGO (adaptive algorithm) – less detectable

	Number of changed pixels				Change rate β (cpp)		
QF	1	10	25	100	0.001	0.005	0.01
80	.0213	.0017	.0022	.0018	.0017	.0007	.0006
	0	0	0	0	0	0	0
90	.1235	.0160	.0065	.0049	.0035	.0024	.0024
	.0852	.0046	.0007	0	0	0	0
95	.4953	.4627	.3974	.2415	.0859	.0191	.0076
	.4948	.4472	.3680	.0977	.0003	0	0

(Recompression artifacts correlate with texture/edges)

Detector for unknown β

- Dropping the assumption of known change rate β
- One-sided hypothesis testing problem: $\beta = 0$ vs $\beta > 0$
- We construct a single classifier and set a threshold according to the predefined value of FA rate (based on covers only)

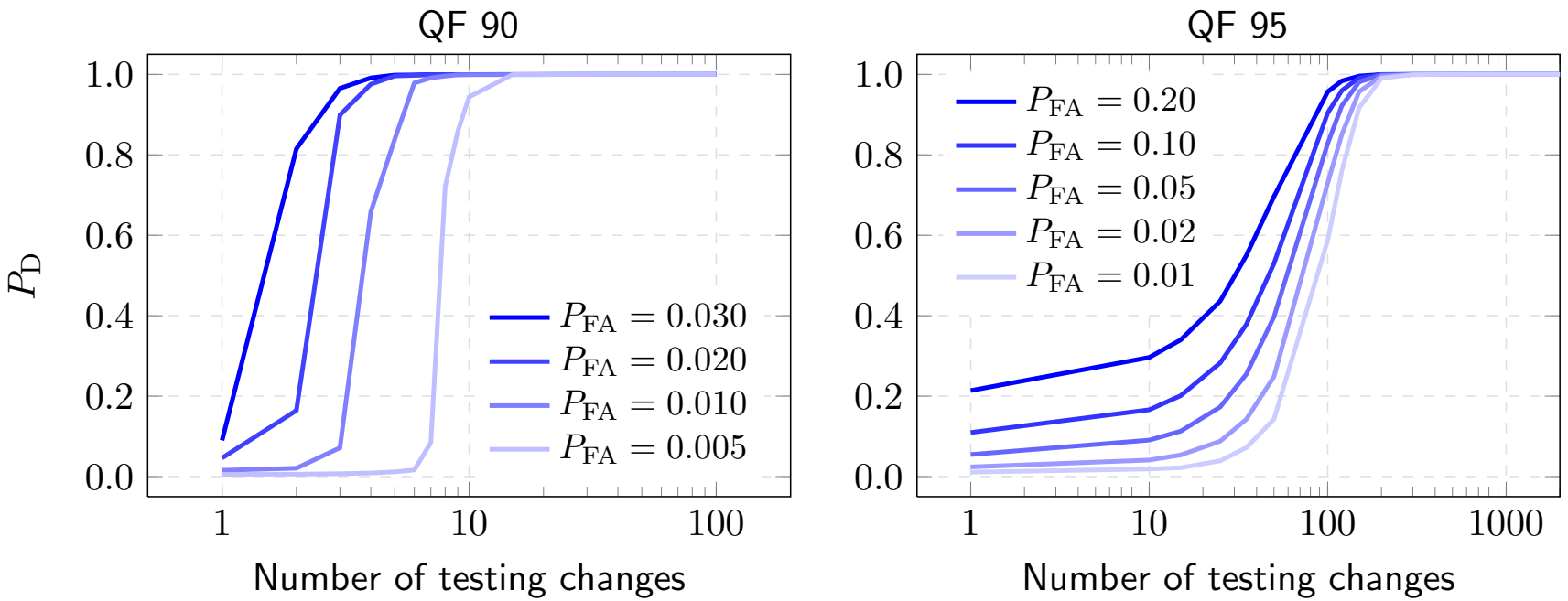
How to train a classifier?

- Pevný (SPIE 2011) – uniform mixture of change rates
- Our scenario: even a small number of changes reliably detected

  $\Rightarrow$ train on a fixed small β with a "reasonable" error rate

  - Error too high $\Rightarrow$ difficult to find optimal decision boundary
  - Error too low $\Rightarrow$ many decision boundaries equally good, but only some of them useful for smaller change rates

Detector for unknown β, cont'd



- P_{D} ... probability of detection
- QF 90: $P_{\mathrm{FA}} = 1\% \Rightarrow$ detects everything with > 10 changes
- QF 95: $P_{\mathrm{FA}} = 1\% \Rightarrow$ detects everything with > 300 changes ($\beta \approx 0.0011$)

Quantitative detector

- Quantitative detector built using Support Vector Regression (SVR)
- Methodology described by Pevný et al. (IEEE TIFS, 2012)
- ν-SVR with a Gaussian kernel (libSVM library)
  - 3 parameters: kernel width, misclassification cost, bound on the number of support vectors $\Rightarrow$ 3D grid-search + cross-validation
- BOSSbase, training change rates chosen uniformly from $[0, b]$
- **Relative** measures of accuracy:

$$B_{\mathrm{r}}(\beta) = \frac{1}{\beta}(\mathrm{median}(\hat{\beta}) - \beta) \times 100\%$$

$$M_{\mathrm{r}}(\beta) = \frac{1}{\beta}\mathrm{median}(|\hat{\beta} - \mathrm{median}(\hat{\beta})|) \times 100\%$$

(More informative for change rates of very different magnitudes)

Quantitative detector, cont'd

β	$b = 0.5$	$b = 0.05$	$b = 0.005$	$b = 0.0005$
$10/n$	$\times$	$\times$	$\times$	-2.78 ± 4.84
$50/n$	$\times$	$\times$	-9.04 ± 8.06	$+0.64 \pm 2.34$
$100/n$	$\times$	-15.6 ± 28.5	-3.36 ± 4.13	-0.22 ± 2.00
0.001	$\times$	-5.326 ± 10.9	-0.19 ± 1.75	-3.83 ± 1.72
0.0035	$\times$	-0.47 ± 3.06	$+0.11 \pm 0.71$	-16.4 ± 1.37
0.01	-16.3 ± 17.2	-0.00 ± 1.06	-0.90 ± 0.80	-43.7 ± 1.07
0.035	-3.74 ± 4.68	$+0.05 \pm 0.40$	$\times$	$\times$
0.1	-1.17 ± 1.74	-21.1 ± 1.17	$\times$	$\times$
0.2	-0.57 ± 0.94	$\times$	$\times$	$\times$
0.3	-0.26 ± 0.79	$\times$	$\times$	$\times$
0.4	$+0.02 \pm 0.51$	$\times$	$\times$	$\times$
0.5	-0.90 ± 1.52	$\times$	$\times$	$\times$

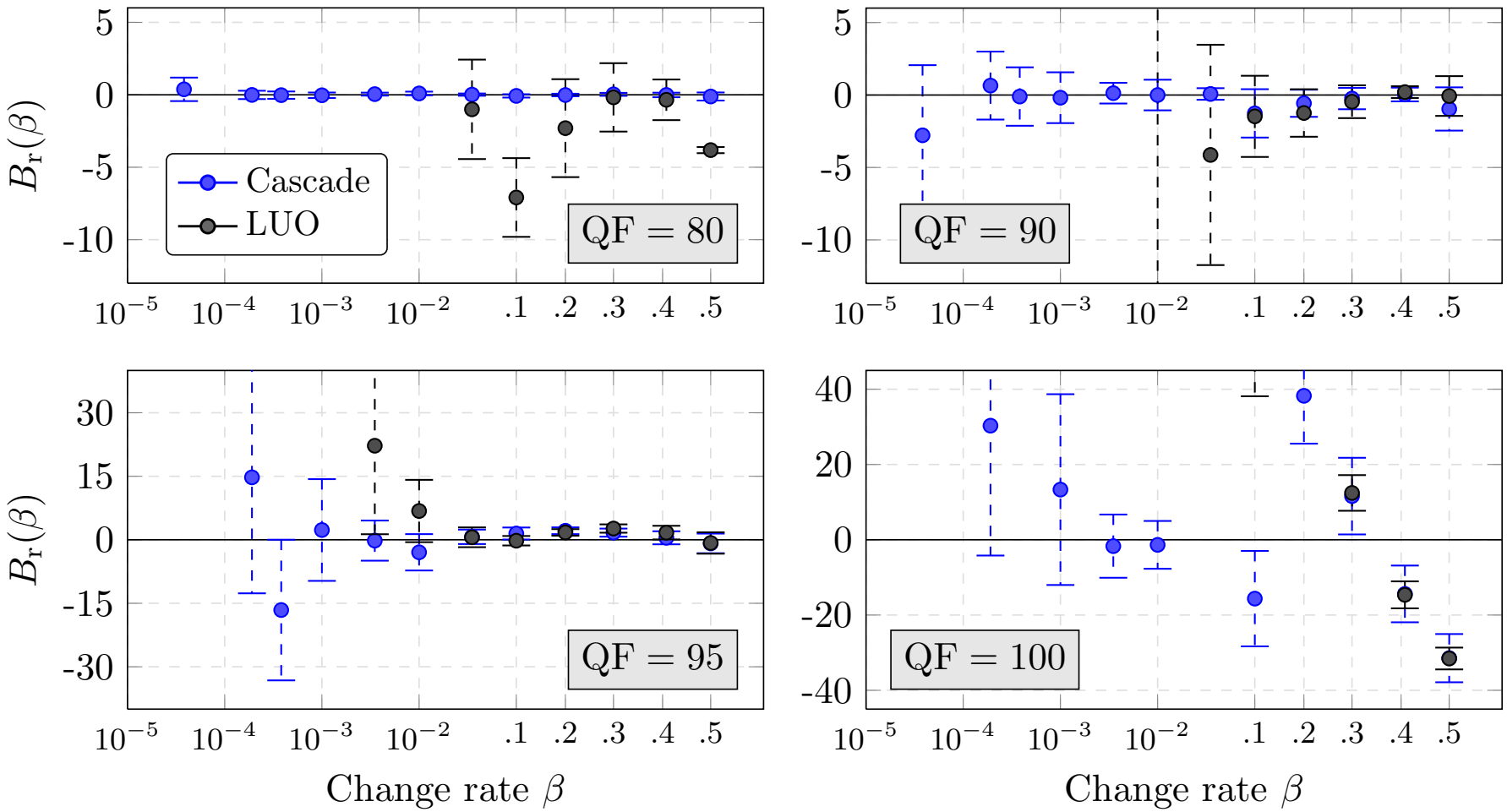
- Numbers ... $B_r(\beta) \pm M_r(\beta)$. Crosses correspond to failures (either B_r or M_r is larger than 50%). JPEG quality fixed to 90. Trained on $[0, b]$.
- Very different magnitudes of testing change rates
  - Training on $[0, 0.5] \Rightarrow$ only 4 training samples with < 100 changes
  - Smaller $b \Rightarrow$ higher training-sample density $\Rightarrow$ more accurate on $[0, b]$

Quantitative detector, cont'd

- Heuristic cascading algorithm to cover **all change rates**:
  1. Set $\mathbf{b} = (b_1, \ldots, b_k)$, $b_i > b_{i+1}$, $b_i \in [0, 0.5]$, initialize $i = 1$.
  2. Compute $\hat{\beta}_i$ by training on $[0, b_i]$. If $i = k$, terminate and output $\hat{\beta}_i$.
  3. If $\hat{\beta}_i \leq b_{i+1}$, increment $i = i + 1$, go to Step 2.
  4. Output $\hat{\beta}_i$.

	$b_1 = 0.5$	$b_2 = 0.05$	$b_3 = 0.005$	$b_4 = 0.0005$	Cascade
$10/n$	×	×	×	-2.78 ± 4.84	-2.78 ± 4.84
$50/n$	×	×	-9.04 ± 8.06	$+0.64 \pm 2.34$	$+0.65 \pm 2.35$
$100/n$	×	-15.6 ± 28.5	-3.36 ± 4.13	-0.22 ± 2.00	-0.10 ± 2.02
0.001	×	-5.326 ± 10.9	-0.19 ± 1.75	-3.83 ± 1.72	-0.19 ± 1.75
0.0035	×	-0.47 ± 3.06	$+0.11 \pm 0.71$	-16.4 ± 1.37	$+0.13 \pm 0.71$
0.01	-16.3 ± 17.2	-0.00 ± 1.06	-0.90 ± 0.80	-43.7 ± 1.07	-0.00 ± 1.06
0.035	-3.74 ± 4.68	$+0.05 \pm 0.40$	×	×	$+0.07 \pm 0.40$
0.1	-1.17 ± 1.74	-21.1 ± 1.17	×	×	-1.27 ± 1.67
0.2	-0.57 ± 0.94	×	×	×	-0.57 ± 0.94
0.3	-0.26 ± 0.79	×	×	×	-0.24 ± 0.74
0.4	$+0.02 \pm 0.51$	×	×	×	$+0.04 \pm 0.47$
0.5	-0.90 ± 1.52	×	×	×	-0.96 ± 1.49

Cascade vs. LUO (LSBR)



Summary

- Accurate JPEG-compatibility steganalysis using block-histogram of the number of mismatched pixels after recompression
- Limitations of prior art
  - WSJPG – limited to LSB replacement
  - LUO – does not distinguish between emb. changes and recompression artifacts
- Three types of detectors constructed
  - Clairvoyant – known β, binary classification
  - Unknown β – one-sided hypothesis testing, CFAR detector
  - Quantitative – outputs estimate of β
- Accurate detection of fewer than 100 changes for QF up to 94
- Proposed method requires images for classifier training
- Trade off between robustness w.r.t. the estimate of θ and the ability to detect embedding schemes other than LSBR