Defeating Statistical Steganalysis

Niels Provos

provos@citi.umich.edu

CITI - Center for Information Technology Integration

University of Michigan

Overview

1. Introduction
2. Background and Related Work
3. Embedding Process
4. JPEG Image Format
5. Statistical Tests
6. Correcting Statistical Deviations
7. Analysis
8. Conclusion

Introduction

- Steganography hides the fact that communication is happening.
- Steganographic systems
  - *classical*: relies on keeping encoding system secret
  - *modern*: detectable only if secret information known
- Invasive nature leaves detectable traces in medium's characteristic.
- Secrecy of the information is not degraded, but its hidden nature is revealed, defeating purpose of steganography.

Introduction

- Information hiding process:
  - Identify redundant bits in cover medium.
  - Embedding process selects a subset to be replaced with data from hidden message.
  - Insert into cover medium to create stego medium.
- Modifying redundant bits changes statistical properties of cover medium.

Introduction

- New method:
  - Preserve statistical properties of cover medium by applying correcting transforms.
  - *A priori* estimate for the amount of data that can be hidden safely.
- Specific transforms depend on data format of stego medium.
- Illustrate statistical tests for JPEG, and demonstrate how they can be defeated.

Background and Related Work

- Two conditions for secure steganography:
  - Secret key used to embed hidden message unknown to adversary.
  - Actual cover medium is unknown to adversary.
- **Easy** in practice:
  - Create cover medium with digital camera, and do not publish original.

Background and Related Work

- In “Attacks on Steganographic Systems”, Westfeld and Pfitzmann describe visual and statistical attacks.
- To detect hidden information, they use a χ^2-test.
- It doesn’t detect information embedded by OutGuess, but we describe an extended χ^2 test that does.
- However, the **correcting transform** prevents detection by both of them.

Embedding Process

- Specific transforms to correct statistics depend on embedding methods.
- Divide embedding into two steps:
  - Identification of redundant data; these can be modified without detectably degrading cover medium.
  - Selection of bits into which hidden information should be placed.

Embedding Process

Identification of Redundant Bits

- Depends on final data format, and might be non-deterministic.
- For JPEG, use the LSB in the quantized discrete cosine transform coefficients.

Embedding Process

Selection of Bits

- RC4 stream cipher is initialized with user-chosen secret key:
  - Use it to encrypt hidden message.
  - Use it to create a pseudo-random number generator (PRNG).
- PRNG select bits to be replaced with hidden information:
  - New bit is determined by $b_i = b_{i-1} + R_i(x)$.
  - $R_i(x)$ is i-th random number in interval $[1, x]$.

Embedding Process

- 32 state bits are embedded before hidden message:
  - 16-bit seed to reseed PRNG.
  - 16-bit hidden data length.
- Interval is adjusted so that hidden message is distributed evenly over the redundant bits.
- Without the secret key, it is not possible to find the hidden message.
- However, the recipient knows the secret key, and uses the same selection process to retrieve the hidden message.

JPEG Image Format

- While embedding methods are independent of data format, each data format has its own statistical properties.
- Restrict analysis to common data format: JPEG.
- JPEG uses a discrete cosine transform (DCT) to map successive 8×8-pixel blocks into 64 DCT coefficients.
- LSB of those quantized DCT coefficients, that are $\neq 0$ and $\neq 1$, are used as redundant bits.

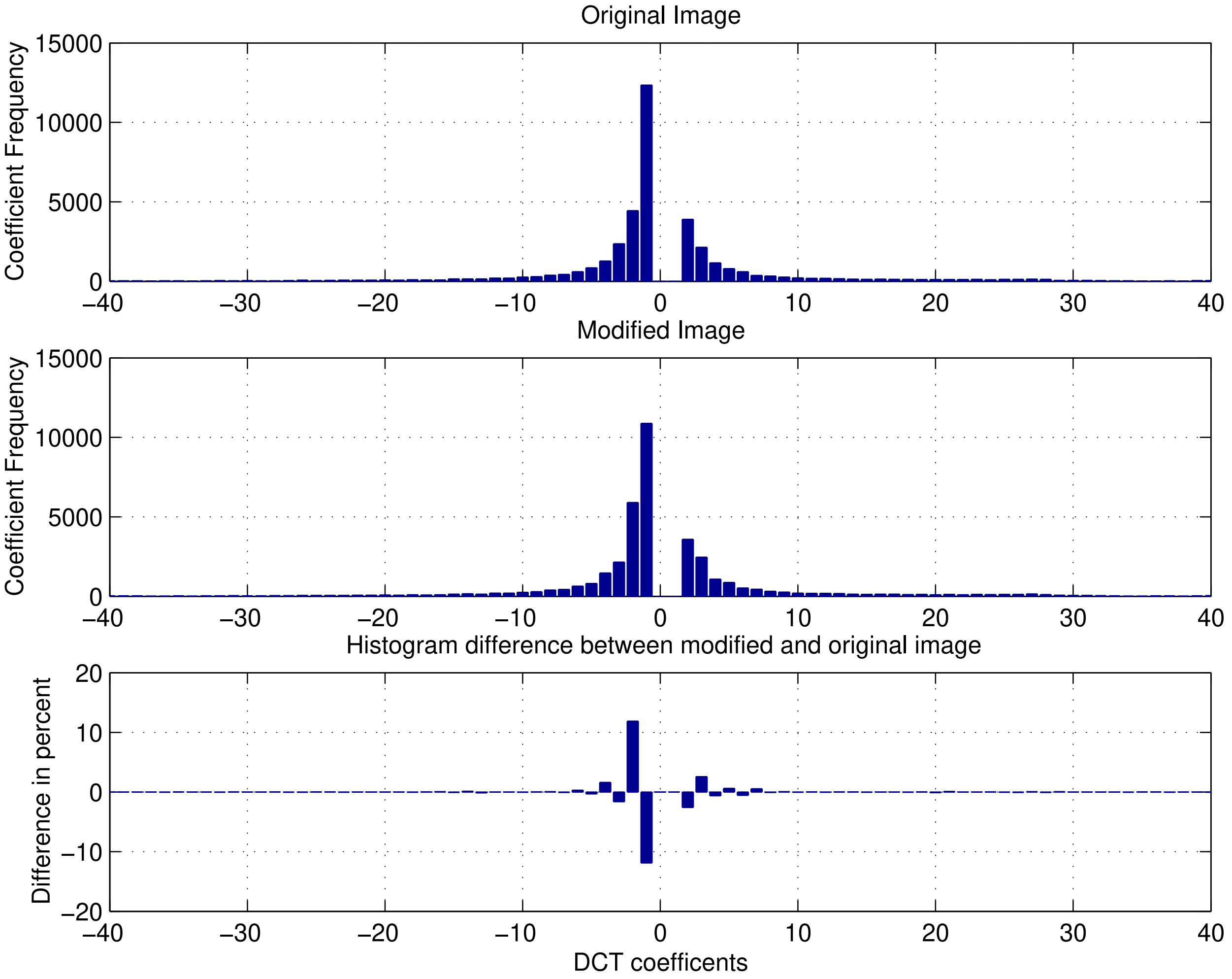
Statistical Tests

- Statistical tests can reveal if an image holds steganographic content.
- Simple tests just measure entropy of the redundant bits:
  - Correlation towards one.
  - Maurer's "Universal Statistical Test for Random Bit Generators"
    * Truly random source is 7.184 (8-bit blocks).
- We expect images with hidden data to have a higher entropy.

Statistical Tests

- Westfeld and Pfitzmann observe that embedding of encrypted data changes the histogram of color frequencies.
- Let n_i and n_i^* be color frequencies before and after embedding data in the LSB of colors.
- Then we expect: $|n_{2i} - n_{2i+1}| \geq |n^*_{2i} - n^*_{2i+1}|$.
- Same is true for JPEG. Take frequency of the DCT coefficients instead.

Statistical Tests



- Differences in DCT histogram without statistical corrections.

Statistical Tests

- Use a χ^2-test to determine whether an observed frequency distribution matches a distribution distorted from embedding hidden information.
- Determine expected distribution by computing arithmetic mean, $y_i^* = \frac{n_{2i}+n_{2i+1}}{2}$, observed distribution by $y_i = n_{2i}$.
- Difference between distributions is $\chi^2 = \sum_{i=1}^{\nu+1} \frac{(y_i - y_i^*)^2}{y_i^*}$.
- Probability p that two distributions are alike is given by the cumulative distribution function.

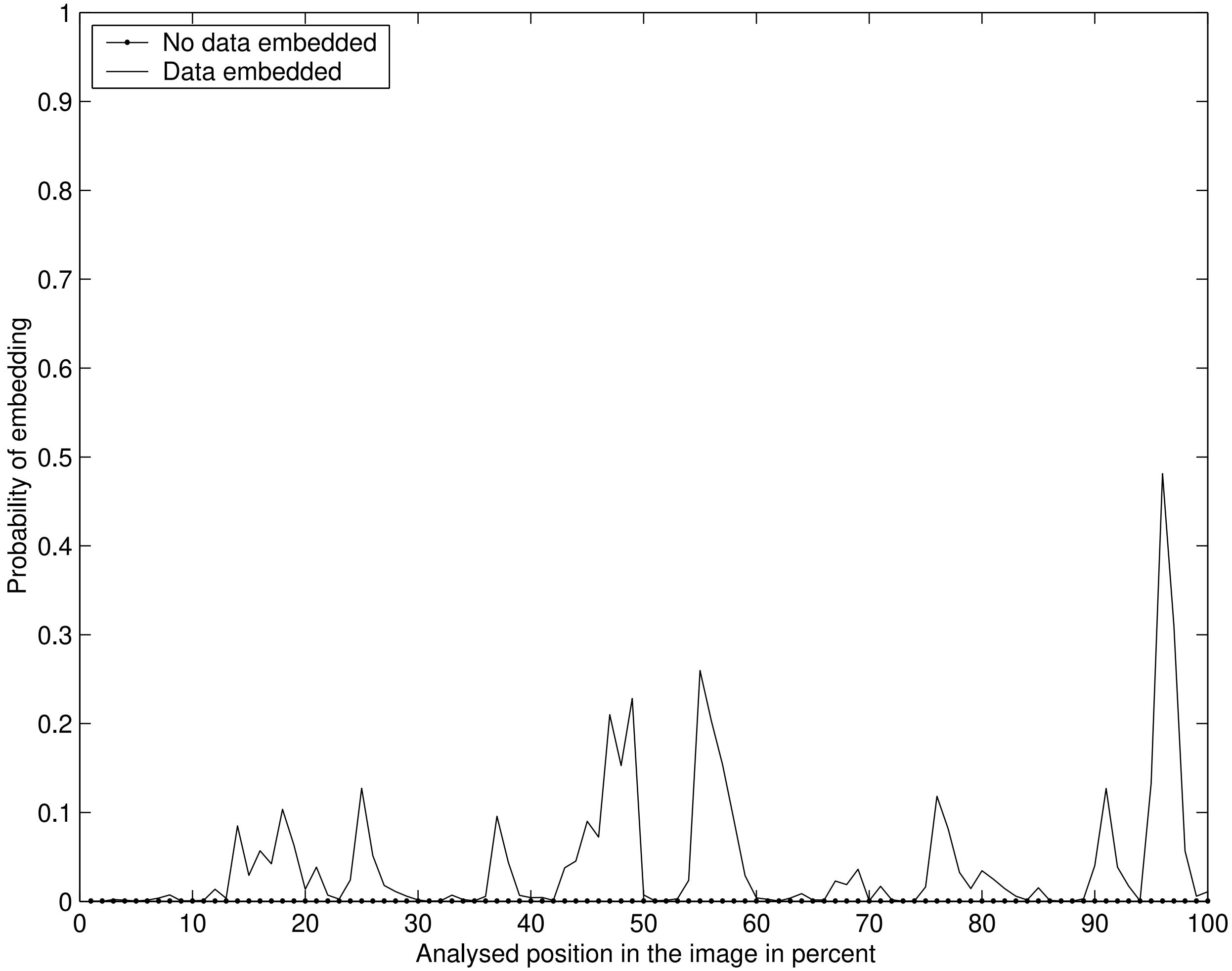
Statistical Tests

- Probability of embedding is determined by calculating p for a sample from the DCT coefficients.
- Sample starts at the beginning of the image, and for each measurement the sample size is increased.
- Increasing sample size and fixed start detects changes only if frequency histogram is distorted continuously from the beginning of the image.
- The test does not detect embedding outlined in this talk.

Statistical Tests

- Create new test that is more sensitive to partial distortions.
- Observe that two identical distributions produce about the same χ^2 values in any part of the distribution.
- Instead of increasing sample size with a fixed position, use a constant sample size but slide the position where samples are taken.
- The extended test is capable of detecting the simple embedding process.

Statistical Tests

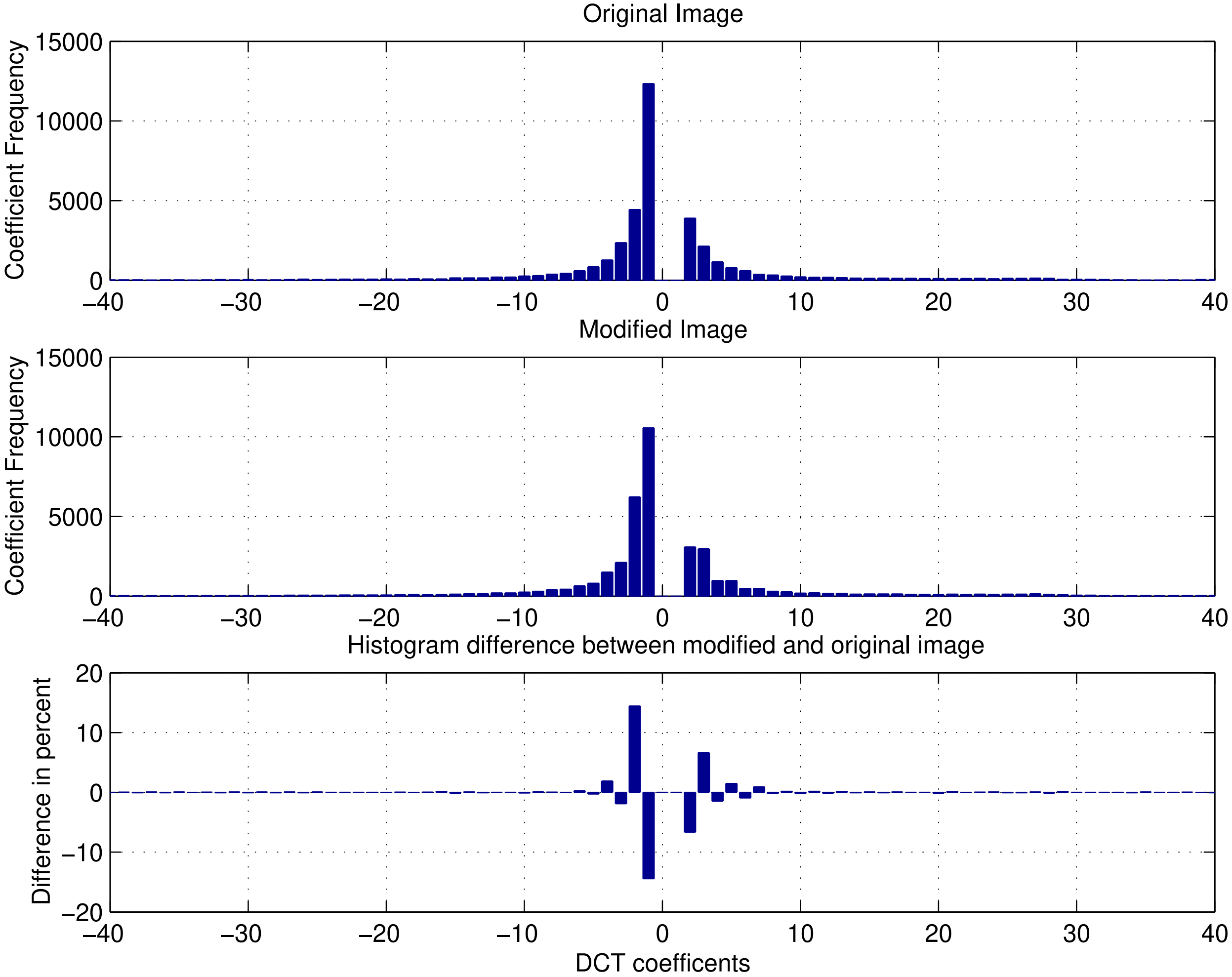


- Extended test detects modified image, but not original.

Correcting Statistical Deviations

- Not all redundant bits are used to hold hidden message bits.
- Use remaining redundant bits to correct statistical deviations for tests that we know about.
- First (naive) approach preserves correlation towards one and Maurer's test.
- When a bit is changed from zero to one, change a nearby bit from one to zero.

Correcting Statistical Deviations



- Naive corrections no good!

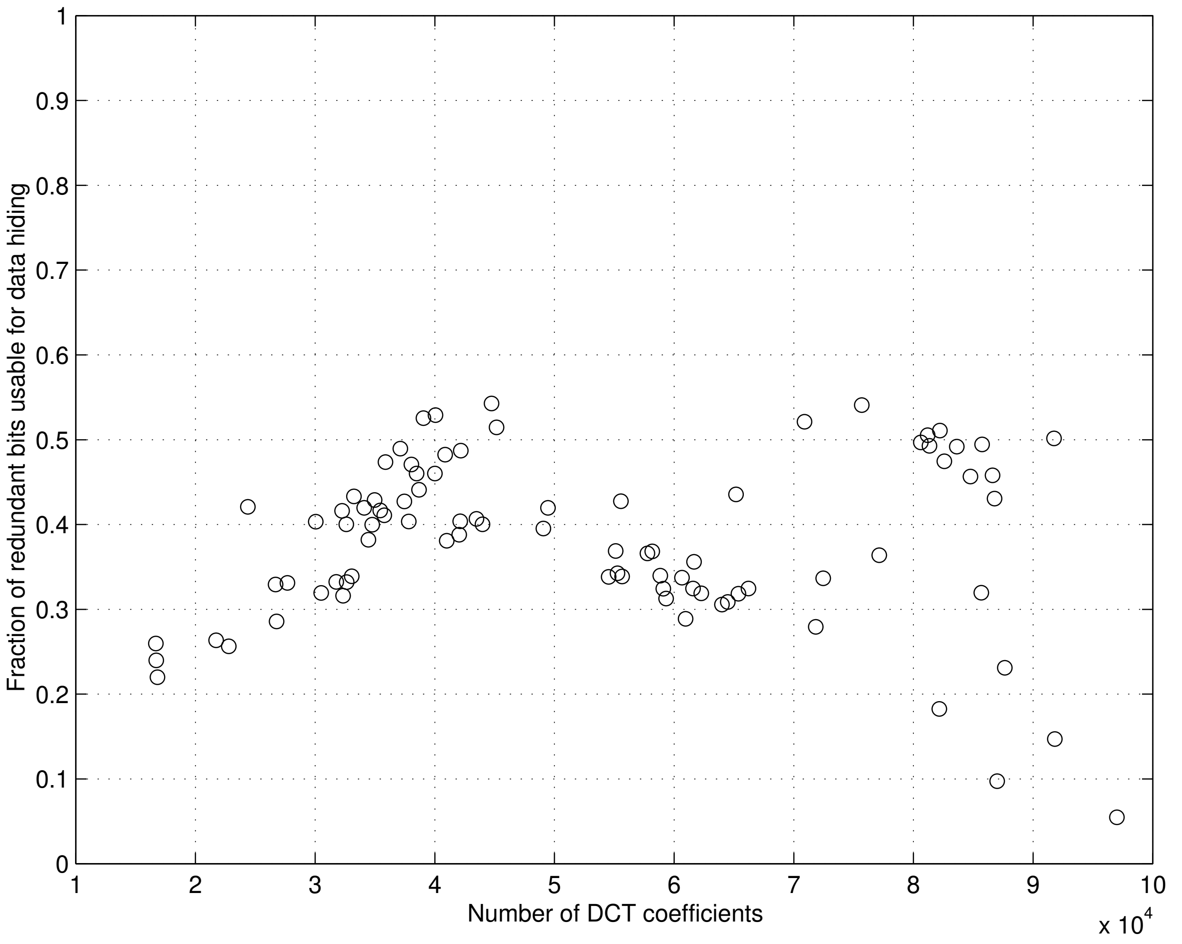
Correcting Statistical Deviations

- To avoid distortions in DCT histogram, correcting transform has to maintain DCT coefficient distribution.
- Correcting transform:
  - If $DCT(j) = 2i$ is changed to $2i + 1$, find *adjacent coefficient* $DCT(k) = 2i + 1$ and change it to $2i$.
- Such a transform keeps all frequency counts constant.
- As a result, no statistic based on frequency counts will detect a difference between original and stego medium.

Correcting Statistical Deviations

- Let f be a frequency count, $\bar{f}$ is adjacent count, and α the fraction of redundant bits used to hold hidden message.
- WLOG, $f \geq \bar{f}$.
- Derive an *a priori* estimate on the amount of data we can embed safely:
  - After embedding, $f^* = f - \frac{\alpha}{2}(f - \bar{f})$ and $\bar{f}^* = \bar{f} + \frac{\alpha}{2}(f - \bar{f})$.
  - Need enough unmodified coefficients for correcting transform to work: $(1 - \alpha)\bar{f} \geq \frac{\alpha}{2}(f - \bar{f})$.
- Deviations can be corrected, if $\alpha \leq \frac{2\bar{f}}{f + \bar{f}}$.

Correcting Statistical Deviations



- Fraction of usable DCT coefficients does not increase linearly.

Correcting Statistical Deviations

- Correcting transform has the following requirements:
  - For any part of the image, distribution of DCT coefficients is similar to unmodified image.
  - Number of corrections should be small.
- Algorithm: For every modified DCT coefficient do:
  1. Trade off against pending errors for adjacent coefficient.
  2. Increase pending errors for this coefficient if below threshold.
  3. Find a nearby adjacent coefficient to exchange this coefficient with.

Analysis

- Look at 54 pictures, 640×480 pixels, taken in Ann Arbor, Michigan.
- They were recompressed with quality factor 75 to simulate embedding without data.
- Average number of DCT coefficients is $46,000$, ranging between $30,000$ and $97,000$.
- Embed first chapter of Lewis Carroll's "The Hunting of the Snark," $14,700$ bits after compression.

Analysis

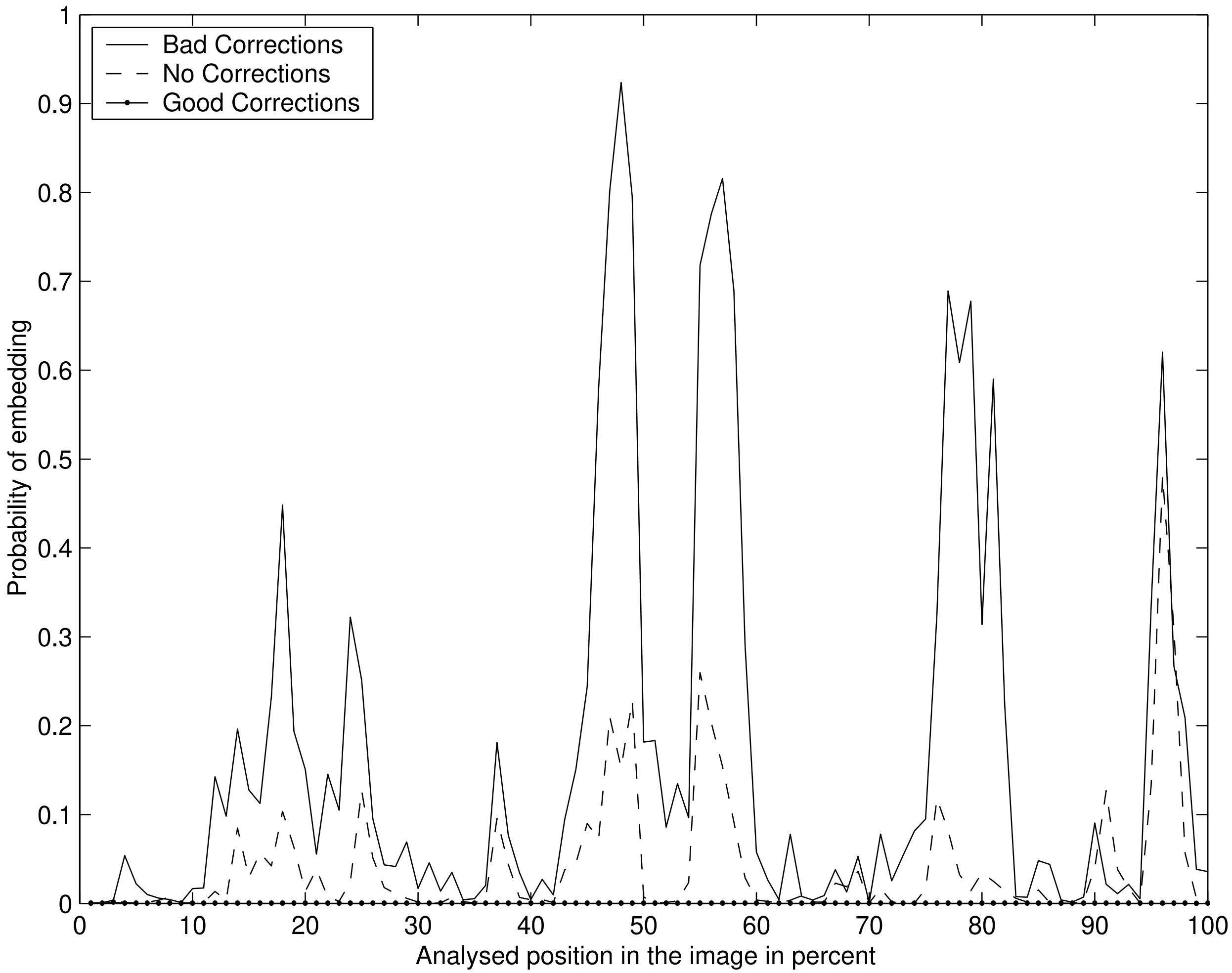
- Correcting transform causes 2967 ± 434 additional changes, approx. 20% of message size.
- Average number of bits where correction failed is 186 ± 400. Majority of corrections fail for images with message size $>$ *a priori* estimate.

Analysis

Method	One-Correlation	Maurer Test
Unmodified	$63.77\% \pm 3.37\%$	6.704 ± 0.253
No corrections	$59.10\% \pm 3.19\%$	6.976 ± 0.168
Corrections	$62.91\% \pm 3.36\%$	6.775 ± 0.231

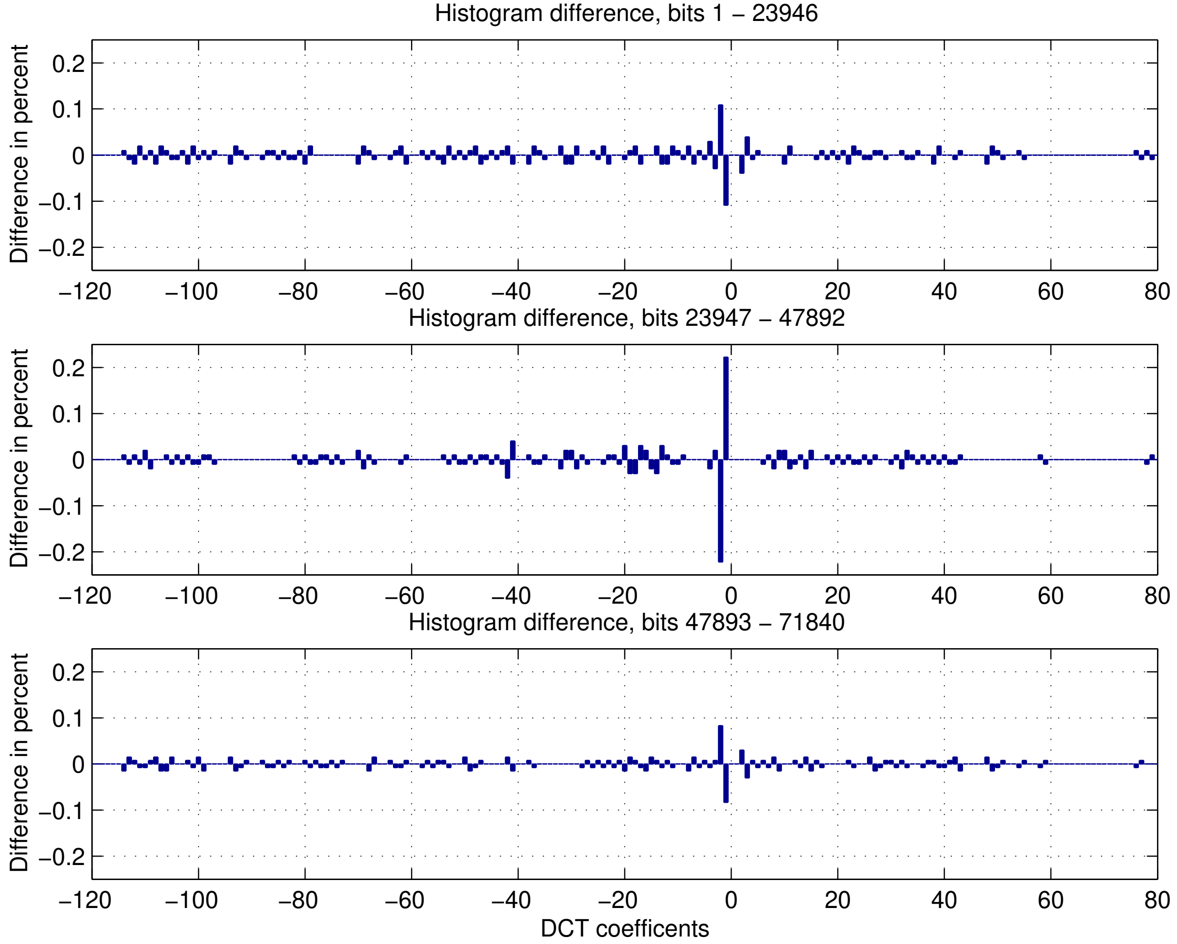
- Noticeable increase in entropy for images without correcting transform applied.
- Images corrected for statistical distortions have similar values as unmodified images.

Analysis



- DCT histogram unchanged, extended χ^2 finds nothing.

Analysis



- No noticeable differences in parts of image.

Conclusion

- Steganography is undetectable by observer's senses, but statistical steganalysis can reveal presence of hidden content.
- Commonly used χ^2-test is unable to detect the presented embedding process. However, the extended χ^2-test can.
- To counter statistical tests, introduce a new method to preserve statistical properties.
- As a result, none of the presented tests can detect the presence of steganography.

Conclusion

- An *a priori* estimate determines the amount of data that can be hidden safely.
- Given a hidden message, the estimate allows us to choose an appropriate image quickly.
- All methods have been implemented in OutGuess. It is freely available at
  `http://www.outguess.org/`

