Application of Vision Transformers in Spatial Steganography

Alireza Nikpoosh, Supervisor: Dr. Samaneh Mashhadi

Department of Mathematics and Computer Science, Iran University of Science and Technology, Tehran, Iran

Abstract

Steganography is the technique of concealing information within an object to ensure it remains undetected and to minimize the risk of compromising the communication channel. In contrast, steganalysis involves detecting and extracting hidden messages from digital media. This dissertation reviews and implements the WOW algorithm alongside ZhuNet, a leading steganalysis algorithm designed to counteract such algorithms using CNNs. Additionally, the dissertation explores Vision Transformers, comparing their effectiveness in steganalysis with that of ZhuNet.

Keywords: Steganography, Vision Transformers, ZhuNet, WOW Algorithm, Steganalysis

1. Introduction

Steganography involves concealing information within objects to ensure undetected communication, while steganalysis aims to detect hidden messages. Various algorithms, including the WOW algorithm and ZhuNet steganalysis model, have been developed to address this. This paper explores the application of Vision Transformers in this field.

2. Vision Transformers

The Vision Transformer model divides images into patches, linearly transforms them, and uses multi-head attention mechanisms to allow communication between patches.

Since Transformers were first used in NLP, this model borrows a trick from this field and adds another vector to the linearly projected patches that

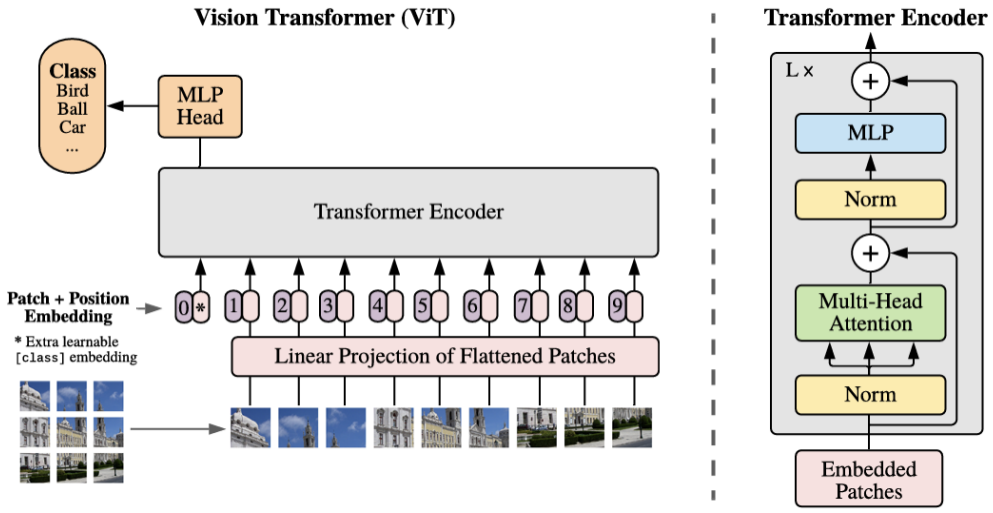


Figure 1: Overview of the Vision Transformer (ViT) architecture. The image is split into patches, embedded, and passed through a Transformer encoder for classification.

doesn't correspond to any parts of the image. This extra embedding, often called the class token, tries to summarize the whole learning process into just one vector that can help predict a class label.

The beating heart of transformers are what is called the Multi-Head Attention block. Roughly speaking, this block allows the different patch embeddings to communicate with each other.

$$Y = Softmax(\frac{QK^T}{\sqrt{d_k}})V \in R^{N \times d_v}$$

This equation summarizes the attention mechanism in a scaled dot product format. To compute attention in a transformer model, the dot product between the query matrix Q and the transpose of the key matrix K^T is calculated to form a score matrix that represents the similarity between each query and key pair. This score matrix is then scaled by dividing it by the square root of the key dimension d_k to stabilize the gradients during training and improve model performance. The scaled scores are passed through a softmax function to obtain normalized attention scores for each key. Finally, these attention scores are used to perform a weighted sum with the value matrix V, resulting in the final output Y, which effectively extracts relevant information from the values based on the attention scores. All of

these matrices are learned through the training process.

3. Spatial Domain Steganography

The main goal of a passive-warden steganographic channel(stego system) between Alice and Bob is to transmit a secret message hidden in an innocuous looking object without any possibility for the warden Eve to detect such communication. A stego system is called perfectly secure if the cover distribution exactly matches the stego distribution. Although this problem has been solved by the so-called "cover generation", this solution requires exact knowledge of the probability distribution on cover objects, which is hard (if possible at all) to obtain for real digital media in practice. The most common practical solution is to hide the message by making small perturbations with the hope that these perturbations will be covered by image noise. One of the most popular embedding methods used with digital images is the Least Significant Bit (LSB) replacement, where the LSBs of individual cover elements are replaced with message bits. It has been quickly realized that the asymmetry in the embedding operation1 is a potential weakness opening doors to the development of highly accurate targeted steganalyzers pushing the secure payload almost to zero. A trivial modification of the LSB replacement method is LSB matching (often called ± 1 embedding). This algorithm randomly modulates pixel values by ± 1 so that the LSBs of pixels match the communicated message. Despite the similarity to LSB replacement, LSB matching is much harder to detect, because the embedding operation is no longer unbalanced. In fact, LSB matching has been shown to be near optimal when only information from a single pixel can be utilized. The biggest weakness of LSB matching is the assumption that image noise is independent from pixel to pixel. It has been shown that this is not true in natural images, which was in different ways exploited by LSB matching detectors. From the short overview of spatial domain steganography above, it is clearly seen that the embedding algorithms are not secure.

4. WOW Algorithm

All of today's most secure steganographic algorithms for digital images use distortion functions that focus the embedding changes to those parts of the image that are difficult to model (e.g., complex textures or "noisy" areas). A distortion function in steganography is a tool that measures the risk of

detection for modifications to a cover image. The goal of distortion function-based steganography is to produce a stego image with optimal insertion and minimal distortion. A natural way to define the distortion function in the spatial domain is to assign pixel costs by measuring the impact of changing each pixel in a feature (model) space using a weighted norm. Making the weights dependent on the pixel's local neighborhood introduces desirable content adaptivity. An example of this approach is the embedding algorithm HUGO , which employs the SPAM feature model. In this paper, the authors approach the task of building distortion functions in the spatial domain using a different strategy. Instead of using a weighted norm in some steganalytic model to compute the pixel costs, they employed a bank of directional high-pass filters to obtain the so-called directional residuals, which are related to the predictability of the pixel in a certain direction. By measuring the impact of embedding on every directional residual and by suitably aggregating these impacts, they forced the embedding cost to be high where the content is predictable in at least one direction (smooth areas and along edges) and low where the content is unpredictable in every direction (e.g., in textured or noisy areas). The resulting algorithm thus becomes highly adaptive and better resists steganalysis using rich models. Virtually all practical steganographic algorithms for digital media strive to minimize an ad hoc embedding impact, which, if properly defined, is correlated with detectability. In its simplest form, embedding impact is simply the number of changes (known as matrix embedding). However, more general ways, as already suggested by Crandal, should be considered. In general, the embedding impact is captured by a non-negative distortion measure $D : X \times X \to [0, \infty]$. During embedding, the algorithm should find a stego image Y, which (a) communicates a given message and (b) achieves minimal value of $D(X, Y)$.

Capital and lower-case boldface symbols stand for matrices and vectors, respectively. The symbols $X = (X_{ij})$, $Y = (V_{ij}) \in \{0, ..., 255\}^{n_1 \times n_2}$ will always be used an 8-bit grayscale cover (and the corresponding stego) image with $n_1 \times n_2$ pixels. For matrix X, X^T is its transpose, $X^{\curvearrowleft}$ is X rotated by 180 degrees, and $|X|$ is the matrix of absolute values. They have restricted their design to additive distortion in the form:

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

where ρ_{ij} are the costs of changing pixel X_{ij} to Y_{ij}. The additivity means

that They have not consider the effects of individual embedding changes influencing each other.

A concept we need to familiarize ourselves with to go any further is padding convolutions. Mirror padding in convolutions is a technique used to extend the borders of an image by mirroring its edge pixels. This approach is commonly used in image processing to maintain the size of the output feature map. By reflecting the pixels at the boundary, mirror padding effectively reduces edge artifacts and ensures that every pixel in the input image contributes to the convolution operation, thereby preserving important features and spatial information at the edges. This is particularly useful when the goal is to maintain the input and output dimensions the same, allowing for consistent feature extraction across the image without losing information near the borders.

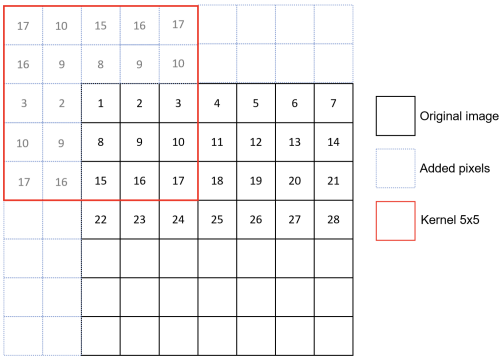


Figure 2: how mirror-padding works

the distortion function of HUGO concentrates the embedding changes primarily in textures and edges. However, the content along an edge can usually be well modeled using locally polynomial models, which aids the detection. Thus, whenever possible the embedding algorithm should embed into textured/noisy areas that are not easily modellable in any direction. To this end, They have evaluated the smoothness in multiple directions using a filter bank $B_n = \{K^{(1)}, \ldots, K^{(n)}\}$ consisting of n multiple directional high-pass filters represented by their kernels normalized so that all L_2-norms $||\mathbf{K}^{(k)}||_2$ are the same. The k-th residual $R^{(k)}$, $k = 1, \ldots, n$, is computed as

$$R^{(k)} = K^{(k)} \star X$$

, where $\star$ is a convolution mirror-padded so that $R^{(k)}$ has again $n_1 \times n_2$ elements. (The mirror-padding prevents introducing embedding artifacts at image boundary.) If the residual values $R_{ij}^{(k)}$ are large for some ij and for all k, it means that the local content at pixel X_{ij} is not smooth in any direction and thus difficult to model. Since we want to detect edges in all directions, it is natural to use established edge detectors for the filter banks (see Table 1). The non-directional 'KB' filter is often used in steganalysis, while the Sobel operator is a common edge detector. Wavelet-based Directional Filter Banks 'WDFB-H' and 'WDFB-D' use the Haar and Daubechies 8-tap wavelets. The computation of the residual coincides with the first-level wavelet decomposition with no decimation. The wavelet banks consist of three filters, $K^{(1)}, K^{(2)}, K^{(3)}$, using which the LH, HL, and HH directional residuals are obtained. Given the wavelet's 1-D low-pass decomposition filter h and a high-pass decomposition filter g, the 2-D directional filters are computed as shown in Table 1.

The embedding should prefer changing large values of directional residuals, where the textures and edges are, and preserve the small values, where the content is predictable. One way to achieve this is to weigh the difference between $R^{(k)}$ and the same residual after changing only one pixel at ij (denoted $R_{[ij]}^{(k)}$) by the wavelet coefficient itself:

$$\zeta_{ij}^{(k)} = |R^{(k)}| \star |R^{(k)} - R_{[ij]}^{(k)}|^{\backsim} \overset{(a)}{=} |R^{(k)}| \star |K^{(k)}|^{\backsim}$$

.

The quantity $\zeta_{ij}^{(k)}$ which we call embedding "suitability," is formally a correlation between the absolute value of the cover residual with the absolute value of the residual change. Since $|R^{(k)} - R_{[ij]}^{(k)}|$ is the spatially shifted directional filter $K^{(k)}$, $\zeta_{ij}^{(k)}$ can be computed for all pixels at once (equality (a)).

Next, we compute the embedding costs ρ_{ij} by aggregating all suitabilities $\zeta_{ij}^{(k)}$, $k = 1, \ldots, n$. Since They have wish to restrict the embedding changes to those pixels with complex content in every direction, the aggregation rule $\rho : R^n \to R_0^+$, $\rho_{ij} = \rho(\zeta_{ij}^{(1)}), \ldots, \zeta_{ij}^{(n)}$ is required to have the following properties:

- The larger the values of $|\zeta_{ij}^{(k)}|$, the smaller the ρ_{ij} should be.
- If there exists $k \in \{1, \ldots, n\}$ such that $\zeta_{ij}^{(k)} = 0$, then $\rho_{ij} = +\infty$.

KB $\mathbf{K}^{(1)} = \begin{pmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{pmatrix}$

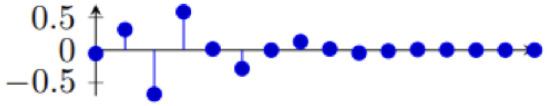
Sobel $\mathbf{K}^{(1)} = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$, $\mathbf{K}^{(2)} = (\mathbf{K}^{(1)})^{\mathrm{T}}$

WDFB-H $\mathbf{h}$ = Haar wavelet decomp. low-pass

$\mathbf{g}$ = Haar wavelet decomp. high-pass

WDFB-D $\mathbf{h}$ = Daubechies 8 wavelet decomp. low-pass

$\mathbf{g}$ = Daubechies 8 wavelet decomp. high-pass



$\mathbf{K}^{(1)} = \mathbf{h} \cdot \mathbf{g}^{\mathrm{T}}$, $\mathbf{K}^{(2)} = \mathbf{g} \cdot \mathbf{h}^{\mathrm{T}}$, $\mathbf{K}^{(3)} = \mathbf{g} \cdot \mathbf{g}^{\mathrm{T}}$

Figure 3: Filter banks used in the WOW paper

A simple function that meets both requirements is the reciprocal Hölder norm with $p > 0$:

$$\rho_{ij}^{(p)} = (\sum_{1}^{n} |\zeta_{ij}^{(k)}|^{p})^{-\frac{1}{p}}$$

They have restrict the embedding changes to ± 1, $|X_{ij} - Y_{ij}| = 1$.

5. Zhu-Net

Steganography is traditionally divided into two stages. Stage one consists on the manual extraction of features where the best results have been achieved using Rich Models(RM). Stage Two Is Based On a binary classifier

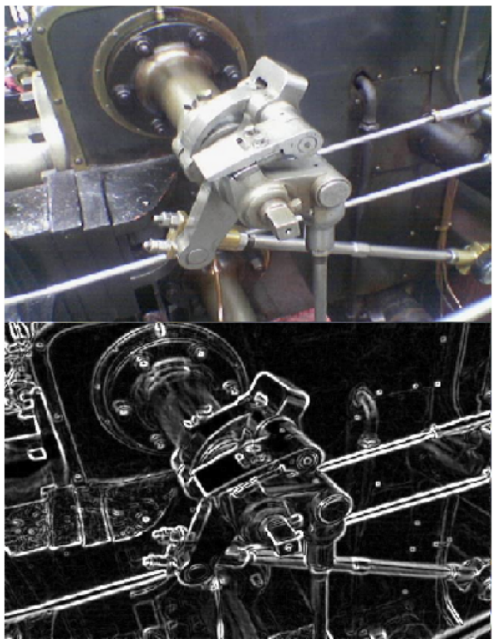


Figure 4: An example of the Sobel Filter applied to an image

(an image is steganographic or not) where Ensemble Classifiers (EC), Support Vector Machines (SVM) or perceptrons [20] are typically used. Thanks to advances in Deep Learning (DL) and Graphic Pro cessing Units (GPUs), researchers have begun to apply these techniques in steganography and steganalysis obtaining better detection percentages of steganographic images. When DL is employed in steganalysis, the feature extraction stage and classification are unified under the same architecture, and the parameters are optimized simultaneously, allowing the complexity and dimensionality introduced by manual feature extraction to be reduced. According to this survey * Zhu-Net performs the best among other proposed CNN architectures for steganalysis like Qian-net, Xu-net, Ye-net and Yedroudj-net. Here we offer a brief explanation of this network.

The framework of the proposed CNN-based steganalysis is illustrated in the above figure. The CNN takes a 256×256 input image and outputs two class labels (stego and cover). The architecture includes an image preprocessing layer, two separable convolution (sepconv) blocks, four basic blocks for feature extraction, a spatial pyramid pooling (SPP) module, and three fully connected layers followed by a softmax layer. The CNN's basic blocks

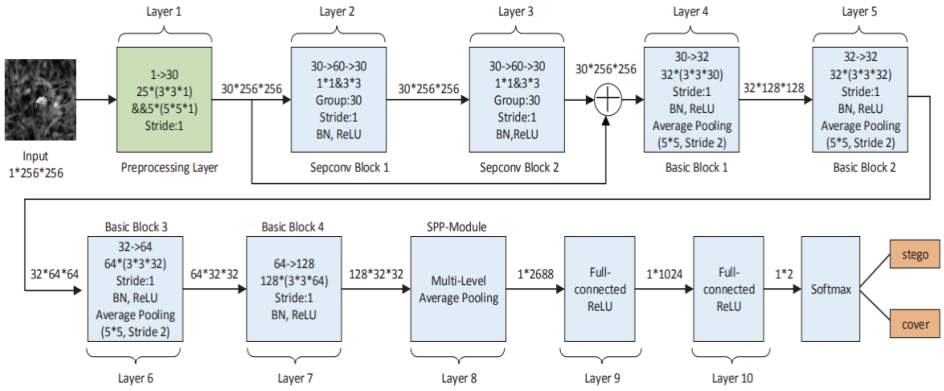


Figure 5: ZhuNet Architecture

consist of the following components:

1. **Convolution Layer**: The authors use small convolution kernels (3×3) to reduce the number of parameters and effectively extract local features. Basic Block 1 to Basic Block 4 have 32, 32, 64, and 128 channels, respectively. Stride and padding sizes are detailed in Fig1.
2. **Batch Normalization (BN) Layer**: BN normalizes each mini-batch to zero-mean and unit-variance, preventing gradient issues and overfitting, and allows larger learning rates for faster convergence.
3. **Non-linear Activation Function**: The authors employ ReLU across all blocks to prevent gradient vanishing/exploding, accelerate convergence, and produce efficient features. A comparison with the Truncated Linear Unit (TLU) showed that ReLU offers lower error rates and better performance.
4. **Average Pooling Layer**: Used in Basic Block 1 to Basic Block 3 to down-sample feature maps, abstract image features, and enhance generalization. The first block omits pooling to avoid information loss.

Separable convolution blocks (Sepconv 1 and 2) enhance the signal-to-noise ratio and handle spatial and channel correlations effectively. The SPP module, as we will explain further on, improves feature extraction through multi-level pooling. The network concludes with three fully connected layers (2688, 1024, and 2 neurons) and a softmax layer for classification into cover or stego.

The SPP module:

For some steganalysis networks, a global average pooling (GAP) layer is added after the last convolution layer for down-sampling, which can greatly reduce the feature dimension. For image classification, GAP is generally used to replace full connected layer to prevent overfitting and reduce computational complexity. This global averaging operation equals to modeling the entire feature map, which leads to information loss of local features. However, modeling local information is very important for steganalysis networks. In this network, the authors use spatial pyramid pooling (SPP) to model local feature map, as shown. SPP has the following properties:

1. SPP outputs a fixed-length feature for any size input.
2. SPP uses multi-level pooling to effectively detect object deformation.
3. Since input is of arbitrary size, SPP can perform feature aggregation for any scale or size images.

As is common with SPP modules, the authors divide the feature maps into several bins. In each spatial bin, they pool the responses of each feature map (average pooling). The output of the spatial pyramid pool is a fixed $k \times M$ dimensional vector, where M is the number of bins and k is the number of filters in the final convolution layer.

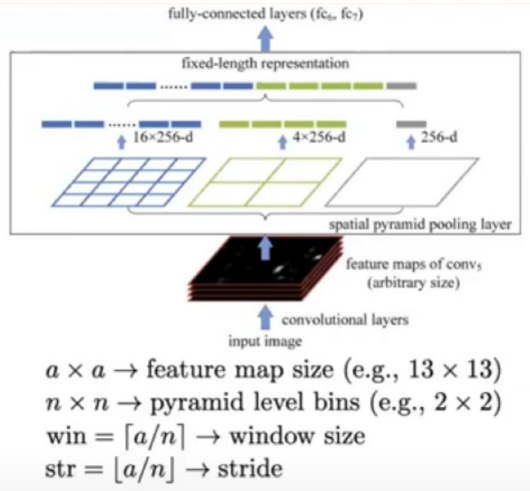


Figure 6: A high-level overview of spatial pyramid pooling

6. Comparative Analysis

As is common with other papers in this field, I used the BOSSBase dataset for comparing the ZhuNet model with Vision Transformers that contains 10000 grayscale images. The WoW algorithm was applied to all 10000 images. In the testing stage, a stego or cover image is randomly fed to the model as input to see the results. three different metrics(accuracy, precision and F1 Score) that are common for benchmarking classification models are used to compare both models. The results show a 7% increase in accuracy, a 6% increase in precision and a 8% increase in the F1 score when testing Vision Transformers. These numbers are an average for 4 different datasets each with 0.1, 0.2, 0.3 and 0.4 payloads.

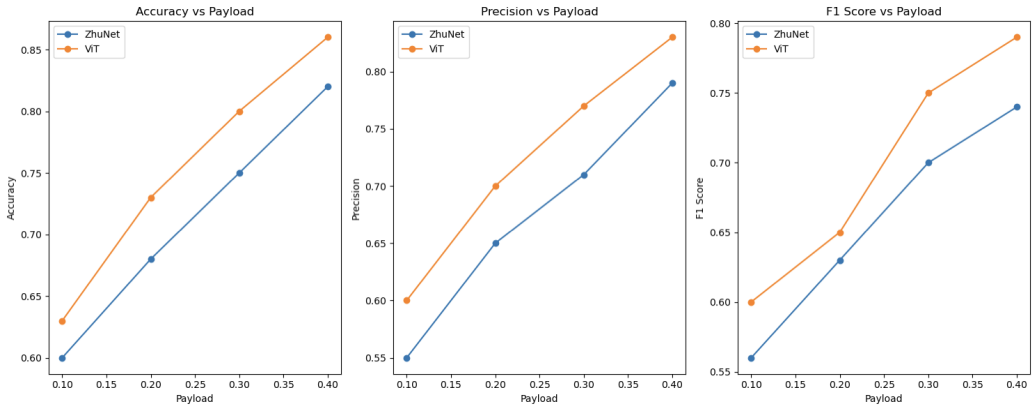


Figure 7: Compared Results of ViT and ZhuNet with different payloads

7. Conclusions

This dissertation has explored and implemented the WOW steganography algorithm alongside ZhuNet, a prominent steganalysis technique designed to counteract WOW. Extensive experimentation using the BOSSBase dataset evaluated ZhuNet's performance in detecting hidden messages and compared it to Vision Transformers. The results demonstrate that Vision Transformers outperform ZhuNet in accuracy and effectiveness for steganalysis. These findings underscore the potential of Vision Transformers as a superior alternative to traditional steganalysis methods, revealing their enhanced capability in uncovering hidden information within digital images. Future research

should investigate the application of transformers and attention mechanisms in steganalysis across various digital media. Additionally, exploring new and specialized transformer architectures could further advance the field of steganalysis.

8. References

Rich Models for Steganalysis of Digital Images - J. Fridrich, Jan Kodovský 2012, IEEE Transactions on Information Forensics and Security
A Survey on Deep Convolutional Neural Networks for Image - I. Hussain, Jishen Zeng, Xinhong, Shunquan Tan 2020, KSII Transactions on Internet and Information Systems Steganography and Steganalysis
Deep Learning Applied to Steganalysis of Digital Images: A Systematic Review Tabares-Soto Reinel; Ramos-Pollán Raúl; Isaza Gustavo; IEEE
Ruan, F., Zhang, X., Zhu, D. et al. Deep learning for real-time image steganalysis: a survey. J Real-Time Image Proc 17, 149–160 (2020)
V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," 2012 IEEE International Workshop on Information Forensics and Security (WIFS), Costa Adeje, Spain, 2012
Attention Is All You Need ; Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin
An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale; A Dosovitskiy
R. Zhang, F. Zhu, J. Liu and G. Liu, "Depth-Wise Separable Convolutions and Multi-Level Pooling for an Efficient Spatial CNN-Based Steganalysis," in IEEE Transactions on Information Forensics and Security, Filler, Tomás Fridrich, Jessica. (2011).
Design of Adaptive Steganographic Schemes for Digital Images. Proceedings of SPIE - The International Society for Optical Engineering.