1. Introduction:

Steganography is the art and science of hiding communication; a steganographic system thus embeds hidden content in unremarkable cover media so as not to arouse an eavesdropper’s suspicion. In the past, people used hidden tattoos or invisible ink to convey steganographic content. Today, computer and network technologies provide easy-to-use communication channels for steganography. Essentially, the information-hiding process in a steganographic system starts by identifying a cover medium’s redundant bits (those that can be modified without destroying that medium’s integrity). The embedding process creates a stego medium by replacing these redundant bits with data from the hidden message.

Modern steganography’s goal is to keep its mere presence undetectable, but steganographic systems—because of their invasive nature—leave behind detectable traces in the cover medium. Even if secret content is not revealed, the existence of it is: modifying the cover medium changes its statistical properties, so eavesdroppers can detect the distortions in the resulting stego medium’s statistical properties. The process of finding these distortions is called statistical steganalysis.

Basics of embedding:

Three different aspects in information-hiding systems contend with each other: capacity, security, and robustness. Capacity refers to the amount of information that can be hidden in the cover medium, security to an eavesdropper’s inability to detect hidden information, and robustness to the amount of modification the stego medium can withstand before an adversary can destroy hidden information. Information hiding generally relates to both watermarking and steganography. A watermarking system’s primary goal is to achieve a high level of robustness—that is, it should be impossible to remove a watermark without degrading the data object’s quality. Steganography, on the other hand, strives for high security and capacity, which often entails that the hidden information is fragile. Even trivial modifications to the stego medium can destroy it. A classical steganographic system’s security relies on the encoding system’s secrecy. An example of this type of system is a Roman general who shaved a slave’s head and tattooed a message on it. After the hair grew back, the slave was sent to deliver the now-hidden message. Although such a system might work for a time, once it is known, it is simple enough to shave the heads of all the people passing by to check for hidden messages—ultimately, such a steganographic system fails. Modern steganography attempts to be detectable only if secret information is known—namely, a secret key. This is similar to Kerckhoffs’ Principle in cryptography, which holds that a cryptographic system’s security should rely solely on the key material. For steganography to remain undetected, the unmodified cover medium must be kept secret, because if it is exposed, a comparison between the cover and stego media immediately reveals the changes. Information theory allows us to be even more specific on what it means for a system to be perfectly secure. Christian Cachin proposed an information-theoretic model for steganography that considers the security of steganographic systems against passive eavesdroppers. In this model, you assume that the adversary has complete knowledge of the encoding system but does not know the secret key. His or her task is to devise a model for the probability distribution \( P_C \) of all possible cover media and \( P_S \) of all possible stego media. The adversary can then use detection theory to decide between hypothesis \( C \) (that a message contains no hidden information) and hypothesis \( S \) (that a message carries hidden content). A system is perfectly secure if no decision rule exists that can perform better than random guessing. Essentially, steganographic communication senders and receivers agree on a steganographic system and a shared secret key that determines how a message is encoded in the cover medium. To send a hidden message, for example, Alice creates a new image with a digital camera. Alice supplies the steganographic system with her shared secret and her message. The steganographic system uses the shared secret to determine how the hidden message should be encoded in the redundant bits. The result is a stego image that Alice sends to Bob. When Bob receives the image, he uses the shared secret and the agreed on steganographic system to retrieve the hidden
message. Figure below shows an overview of the encoding step; as mentioned earlier, statistical analysis can reveal the presence of hidden content.

**Discrete cosine transform**

For each color component, the JPEG image format uses a *discrete cosine transform* (DCT) to transform successive 8 X 8 pixel blocks of the image into 64 DCT coefficients each. The DCT coefficients $F(u, v)$ of an 8 X 8 block of image pixels $f(x, y)$ are given by, where $C(x) = 1/2$ when $x$ equal 0 and $C(x) = 1$ otherwise.

Afterwards, the following operation quantizes the coefficients:

$$F(u, v) = \frac{1}{4} C(u)C(v) \sum_{x=0}^{7} \sum_{y=0}^{7} f(x, y) \cdot \cos \left( \frac{(2x + 1)u\pi}{16} \right) \cdot \cos \left( \frac{(2y + 1)v\pi}{16} \right)$$

$$F^Q(u, v) = \left[ \frac{F(u, v)}{Q(u, v)} \right]$$

where $Q(u,v)$ is a 64-element quantization table.

We can use the least-significant bits of the quantized DCT coefficients as redundant bits in which to embed the hidden message. The modification of a single DCT coefficient affects all 64 image pixels.
In some image formats (such as GIF), an image’s visual structure exists to some degree in all the image’s bit layers. Steganographic systems that modify least-significant bits of these image formats are often susceptible to visual attacks. This is not true for JPEGs. The modifications are in the frequency domain instead of the spatial domain, so there are no visual attacks against the JPEG image format. Figure 2 shows two images with a resolution of 640 × 480 in 24-bit color. The uncompressed original image is almost 1.2 Mbytes (the two JPEG images shown are about 0.3 Mbytes). Figure a is unmodified; Figure b contains the first chapter of Lewis Carroll’s *The Hunting of the Snark*. After compression, the chapter is about 15 Kbytes. The human eye cannot detect which image holds steganographic content.
2. Topics covered:

JSteg algorithm:

Input: message, cover image  
Output: stego image  
while data left to embed do  
    get next DCT coefficient from cover image  
    if DCT ≠ 0 and DCT ≠ 1 then  
        get next LSB from message  
        replace DCT LSB with message LSB  
    end if  
end while  
insert DCT into stego image

chi-square test:

Derek Upham’s JSteg was the first publicly available steganographic system for JPEG images. Its embedding algorithm sequentially replaces the least-significant bit of DCT coefficients with the message’s data (see Figure below). The algorithm does not require a shared secret; as a result, anyone who knows the steganographic system can retrieve the message hidden by JSteg. Andreas Westfeld and Andreas Pfitzmann noticed that steganographic systems that change least-significant bits sequentially cause distortions detectable by steganalysis. They observed that for a given image, the embedding of high-entropy data (often due to encryption) changed the histogram of color frequencies in a predictable way. In the simple case, the embedding step changes the least-significant bit of colors in an image. The colors are addressed by their indices $i$ in the color table; we refer to their respective frequencies before and after embedding as $n_i$ and $n_i^*$. Given uniformly distributed message bits, if $n_i > n_{i+1}$, then pixels with color $2i$ are changed more frequently to color $2i + 1$ than pixels with color $2i + 1$ are changed to color $2i$. As a result, the following relation is likely to hold:

$$|n_{2i} - n_{2i+1}| \geq |n_{2i}^* - n_{2i+1}^*|.$$  

In other words, embedding uniformly distributed message bits reduces the frequency difference between adjacent colors. The same is true in the JPEG data format. Instead of measuring color frequencies, we observe differences in the DCT coefficients’ frequency. Figure below displays the histogram before and after a hidden message is embedded in a JPEG image. We see a reduction in the frequency difference between coefficient –1 and its adjacent DCT coefficient –2. We can see a similar reduction in frequency difference between coefficients 2 and 3.
Westfeld and Pfitzmann used a \( \chi^2 \)-test to determine whether the observed frequency distribution \( y_i \) in an image matches a distribution \( y^* \) that shows distortion from embedding hidden data. Although we do not know the cover image, we know that the sum of adjacent DCT coefficients remains invariant, which lets us compute the expected distribution \( y^* \) from the stego image. Letting \( n \) be the DCT histogram, we compute the arithmetic mean to determine the expected distribution and compare it against the observed distribution \( y_i = n_{2i} \).

\[
y_i^* = \frac{n_{2i} + n_{2i+1}}{2}
\]

The value for the difference between the distributions is given as

\[
\chi^2 = \sum_{i=1}^{v+1} \left( \frac{y_i - y_i^*}{y_i^*} \right)^2,
\]

where \( v \) are the degrees of freedom—that is, one less than the number of different categories in the histogram. It might be necessary to sum adjacent values from the expected distribution and the observed distribution to ensure that each category has enough counts. Combining two adjacent categories reduces the degrees of freedom by one. The probability \( p \) that the two distributions are equal is given by the complement of the cumulative distribution function,

\[
p = 1 - \int_0^{\chi^2} \frac{t^{(v-2)/2} e^{-t/2}}{2^{v/2} \Gamma(v/2)} dt.
\]

where \( \Gamma \) is the Euler Gamma function. The probability of embedding is determined by calculating \( p \) for a sample from the DCT coefficients. The samples start at the beginning of the image; for each measurement the sample size is increased. Figure 5 shows the probability of embedding for a stego image created by JSteg. The high probability at the beginning of the image reveals the presence of a hidden message; the point at which the probability drops indicates the end of the message.

**color clipping:**

Depending on the rounding used in quantization, it is possible that the reconstructed image data may be outside the expected range which is often known as color overflowing. In other words, the clipping
Many JPEG steganographic techniques embed message into the quantized DCT coefficients in the compression process. For example, Jsteg embeds the message by modulating the least significant bit of the non-zero and non-one quantized DCT coefficients. This embedding process as well as the rounding one degrades the image quality. Accordingly, the number of color clipping in the decompression process also increases.

### Proposed Steganalytic Model

When a JPEG image is intercepted on the Internet, the following steps are applied to it:

**Step 1.** Decompress the image and record the number of color overflows. Let $CO_0$ denote the overflow number.

**Step 2.** Embed the binary random data into the LSB of non-zero and non-one quantized DCT coefficients and record the number of color overflows. Repeat Step 2 $n$ times, and let $CO_1, CO_2, \ldots, CO_n$ denote the sequence of overflow numbers, $CO = \{ CO_1, CO_2, \ldots, CO_n \}$.

**Step 3.** Find the minimal value of the $CO$ sequence and let $CO_{\text{min}}$ denote the minimal value of the $CO$ sequence.

**Step 4.** IF ($CO_0 \geq CO_{\text{min}}$)
Classify the image into “Jsteg image”

ELSE

Classify the image into “not Jsteg image”.

The next problem we will address is that how many times should Step 2 be repeated. Fig. 5 shows the change of accumulated overflows and the minimal value under different sample sizes. The amount of different overflows – union of all overflow sets – is 10271, about 1.31% of total color values. When the sample size is 181, all of the possible overflows had taken place at least one time. The sample size with maximal accumulated overflows (MAO) is called the saturation point (SP) of the image. The first minimal value (FMV) appears at the 144th sample in the experiment on Waterlily image. Observing our experimental results, all of the first minimal values appear before the saturation point. Therefore, the repeating process in Step 2 stops at the saturation point is good enough to find the minimal value of the CO sequence.

3. References:

1. A NOVEL QUANTITY BASED ON CLIPPING STATISTICS FOR JSTEG STEGANALYSIS, 8th IASTED Int. Con. on Signal & Image Processing (SIP 2006) August 14-16, 2006, Honolulu, Hawaii, USA
2. JPEG Quantization-Distribution Steganalytic Method Attacking Jsteg
3. Attacks on Steganographic Systems Breaking the Steganographic Utilities EzStego, Jsteg, Steganos, and S-Tools—and Some Lessons Learned, Andreas Westfeld and Andreas Pfitzmann