Steganography & Steganalysis of Images

Mr C Rafferty

Msc Comms Sys Theory 2005

Definitions

- **Steganography** is **hiding a message** in an **image** so the manner that the very existence of the message is unknown.
- The goal of Steganography is to avoid drawing suspicion to the transmission of a hidden message. If suspicion is raised, then this goal is defeated.
- **Steganalysis** is the art of **discovering** and rendering useless such covert messages.

Least significant Bit (LSB)

- LSB insertion is a common simple approach to embedding information in a image
- But it is vulnerable to even slight image manipulation.
- Converting image from a format like GIF or BMP to JPEG and back could destroy the information hidden in LSBs.
- Here’s how it works…

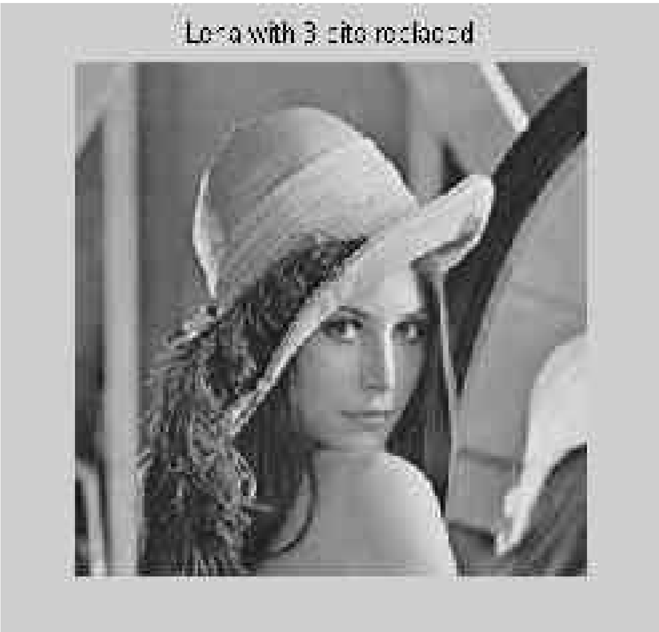
LSB

- Taking advantage of the way the human eye perceives images, this technique involves replacing the N least-significant bits of each pixel of a container image with the data of a hidden message.
- Example: Consider 1 pixel of an 8 bit grayscale Image
- Eg. graylevel intensity
- 156 = 1001 1100
- Hide data at say N=2 1001 1111
- New value 1001 1111 = 159

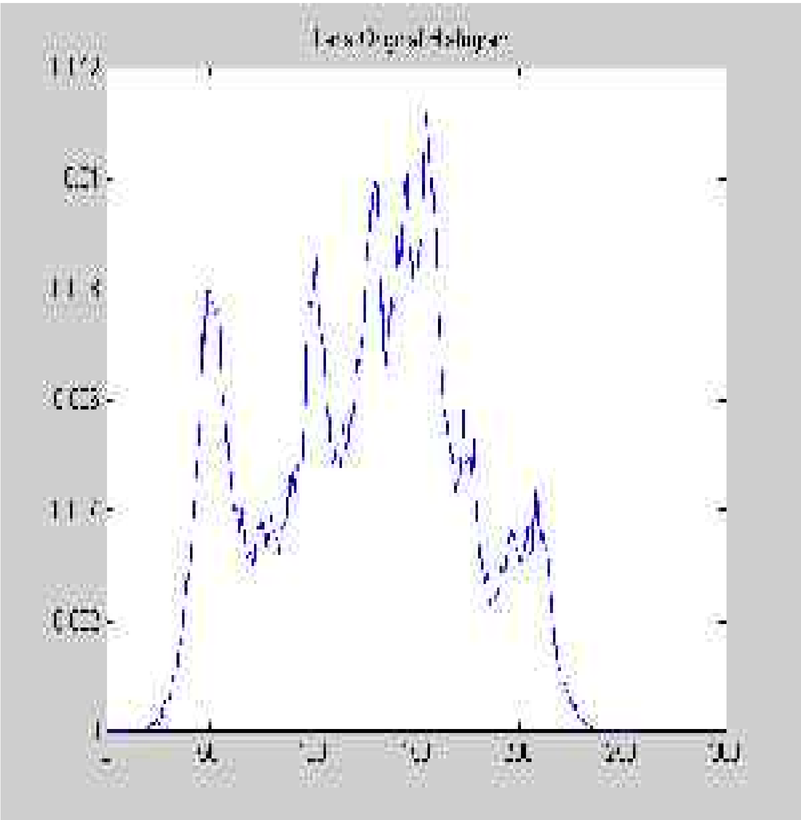
LSB

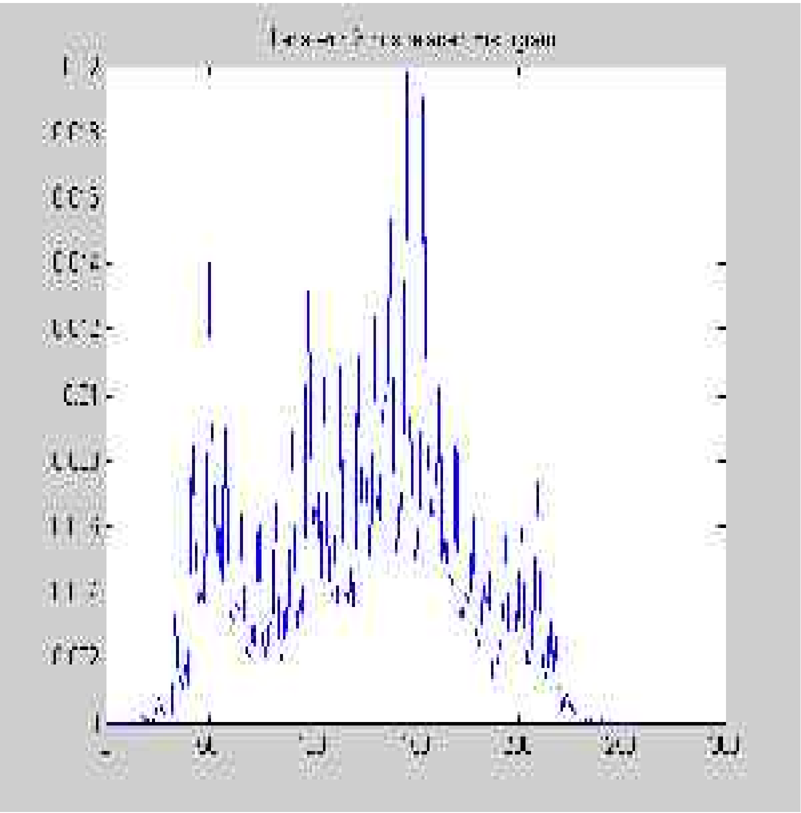
- ✂←The famous 'Lena' image.(8 bit grayscale)

- ***"We see little difference"***

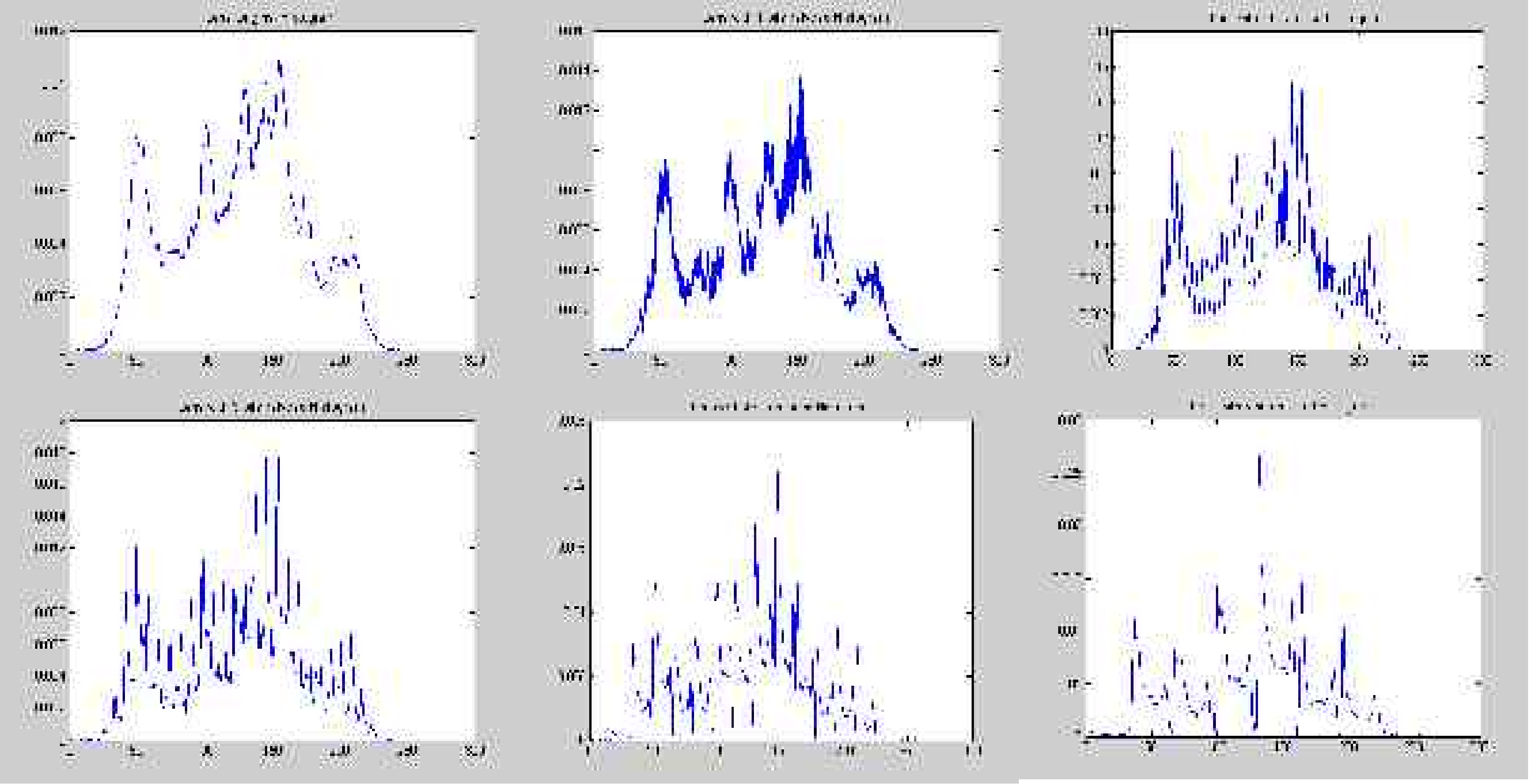


- ✂←'Lena' with 3 bits replaced using LSB





- Best viewed Plotting Histogram of Pixel Values
- We can see Original Image have ‘smooth’ curve
- Most Images have a ‘smooth’ distribution
- The Image with Data (Stego-Image) produces this ‘zig-zag’ curve
- This ‘zig-zag’ curve is common in LSB hiding.



- Note as **N Increases**, **Hiding Capacity Increases** but Image degrades.
- Notice this ‘zig-zag’ feature appearing after N bits replaced with image
- Most Images used have ‘Smooth’ Histogram
- Easy for Steganalysist to Suspect Image may have Hidden data

LSB Conclusions

- So we see that LSB technique is an easy simple method for hiding data
- But… stego-images can draw suspicion or be easily detected from statistical analysis. (Histogram analysis, as shown)
- Stego-Image can become distorted as N Increases

- ***So, Must defeat this Statistical Analysis Attack!!***

Project Outline

- LSB hides data in the Spatial domain
- Lets look at hiding in the **Frequency Domains**…
- Hiding Data using…
  - **Discrete Cosine Transform – DCT**
  - **Fast Fourier Transform - FFT**
  - **Wavelet Tranform – WT**
  - **Gabor Transform**

Project Outline

- Today look at Hiding Data using

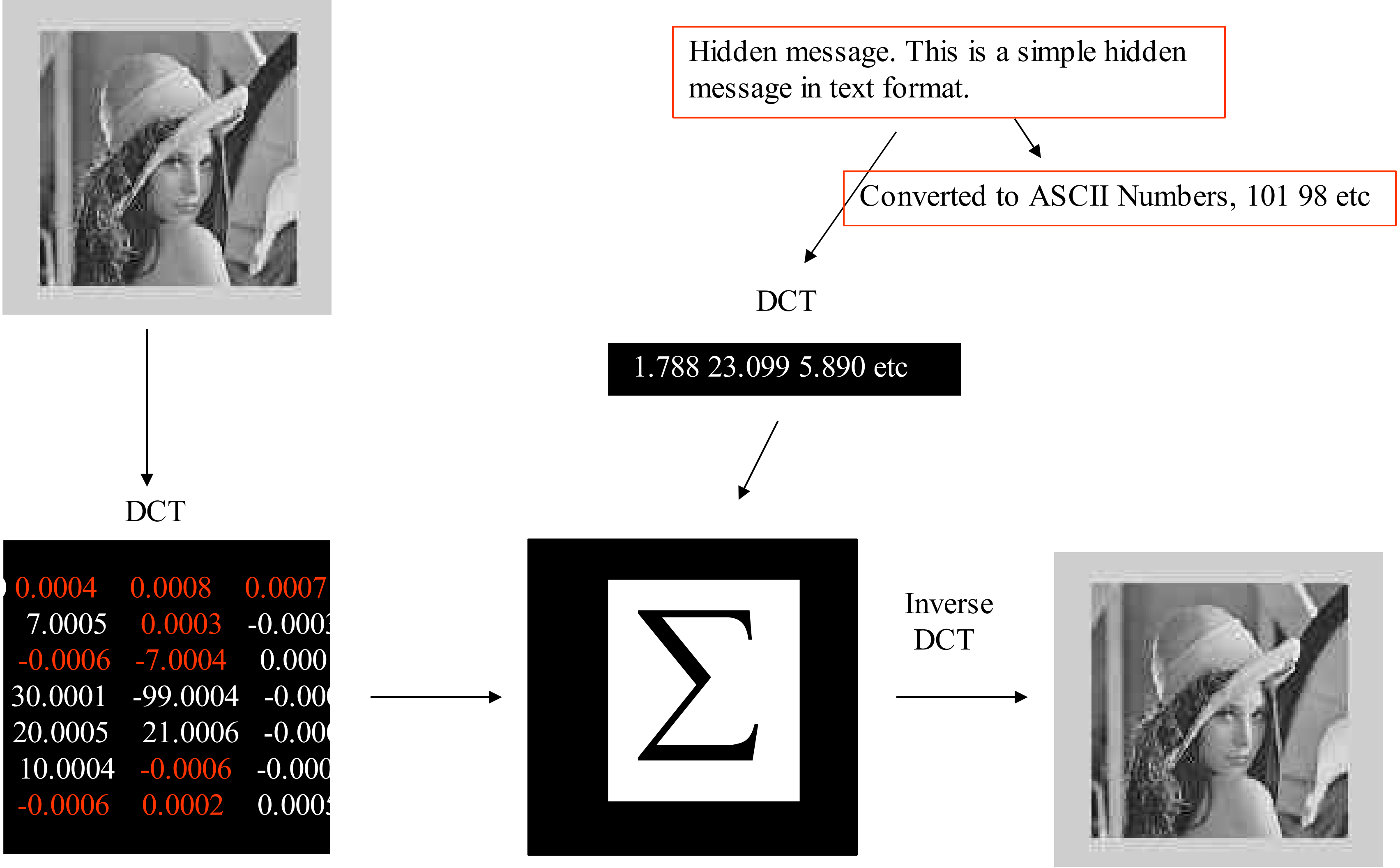
3. Discrete Cosine Transform – DCT
4. Wavelets Transform - WT

DCT

- The DCT transforms a signal or image from the spatial domain to the frequency domain.
- It separates the image into parts (or spectral sub-bands) of **differing importance** (with respect to the image's visual quality).
- It can separate the Image into **High**, **Middle** and **Low** Frequency components.

DCT Algorithm

- Take DCT of Image
- Take DCT of hidden Message
- Set **T**, the ***Significant Threshold*** value, below which transform coefficients will de deemed Insignificant
- Find these **T** Values and Replace these by a Function of the hidden message
- Take the Inverse DCT of this New Image
- Output → Stego Image



Hidden message. This is a simple hidden message in text format.

Converted to ASCII Numbers, 101 98 etc

DCT

1.788 23.099 5.890 etc

DCT

0.0004	0.0008	0.0007
7.0005	0.0003	-0.0003
-0.0006	-7.0004	0.000
30.0001	-99.0004	-0.00
20.0005	21.0006	-0.00
10.0004	-0.0006	-0.000
-0.0006	0.0002	0.0005

Σ

Inverse DCT

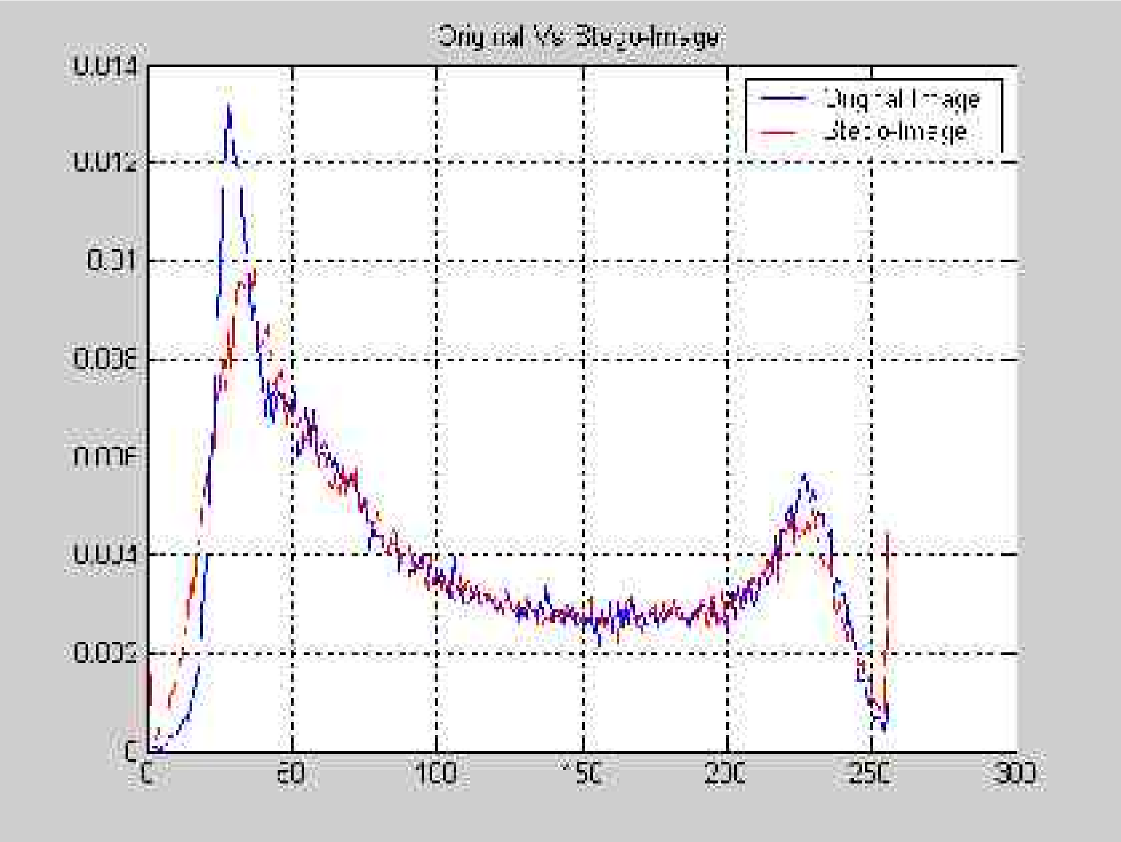
Select values below T, **Significant Threshold**

Replace (using an operator) each discaded insignificant Transform coefficient with coefficient from message

Stego – Image Returned

DCT Results to date

- The **human eye** is able to **catch modifications** to the **lower frequencies** since most of the image's frequency content is located in this area.
- **Modifying high frequencies** can cause a multitude of local **distortions** along the image's sharp edges
- To minimize distortion, then, the data should be inserted in the **middle frequencies**.
- **Middle Frequencies are deemed insignificant transform coefficients as thus set Theshold Value in this space**

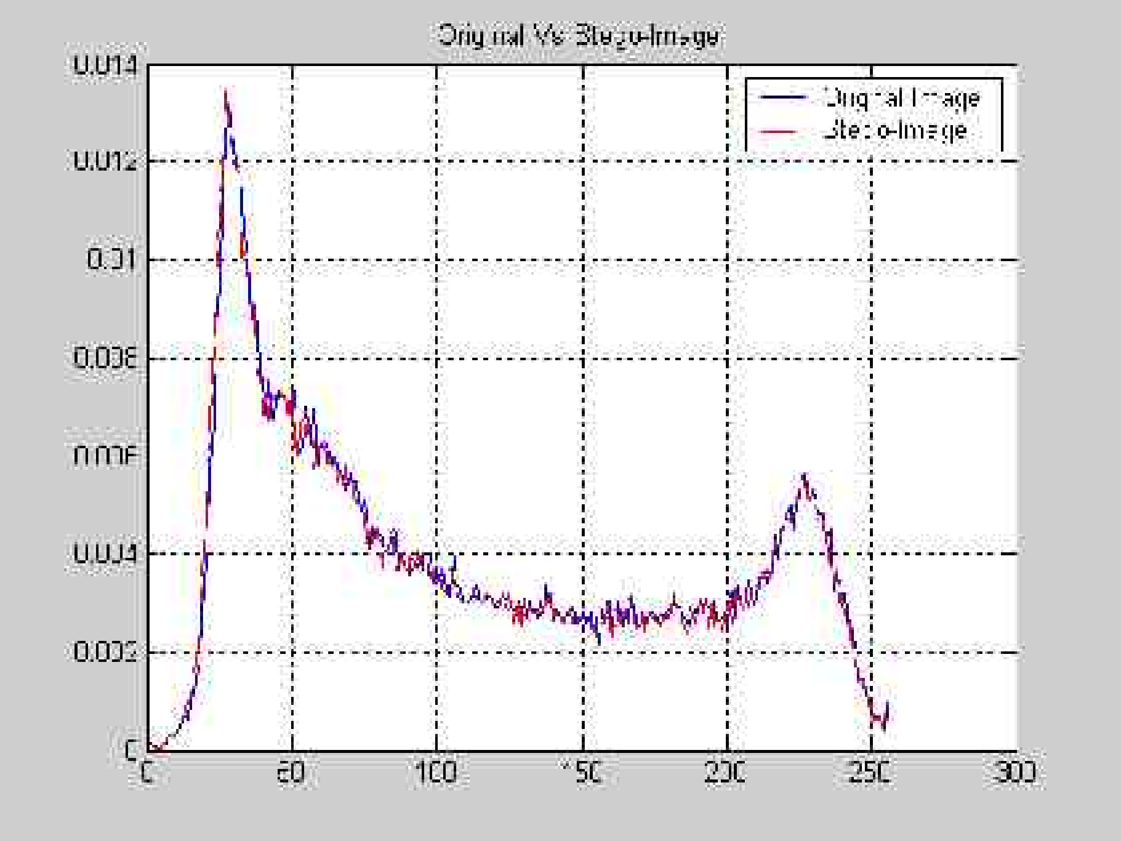


DCT Algorithm Results

- Histograms taken setting T below **low** frequencies

•**But** sharp ‘spike’ at 255 pixel value (not good Result)

•Steganalysist “Alarm Bells”



DCT Algorithm Results

- Histograms taken setting T below **Middle** frequencies
- We can see it closely matches the original Image
- We can see already we have elimated the ‘zig-zag’ effect from LSB Insertion
- **Thus DCT Algorithm Defeats Histogram Analsis**

Extracting Data

- The reciever will need the Following

2. Stego-image (downloaded from web)
3. Stego Key
   - T, the Threshold Significant Value, to look for the coefficients below these values
   - Original Image
   - Then extract by subtracting the coefficients below T from Original coefficients and Inverse DCT

Extracting Data

- To date, DCT Algorithm can sucessfully extract a small Text Message
- **Future work** will be increasing the **Payload** (amount hidden data)
- Thus seeing how much hiding bandwidth can be used on different Images.
- **Plot : Image Degradation Vs Payload**
- **Plot : Successfully Extraction Vs Payload**

Wavelet Experimentation: The Wavelet Transform

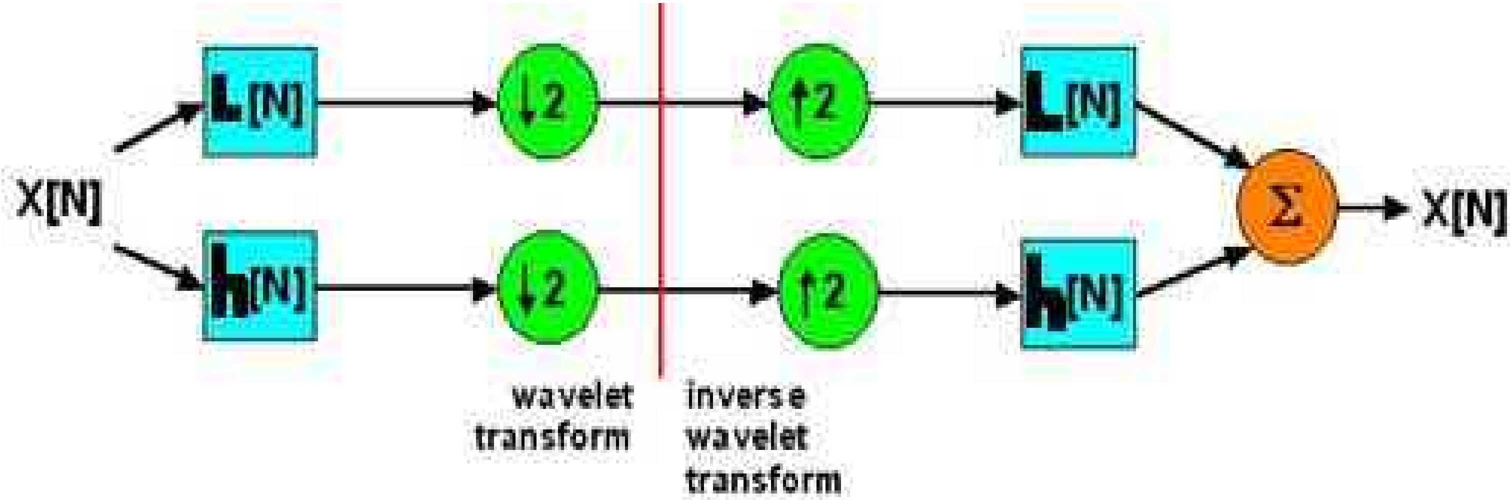
- The Haar wavelet basis was chosen.
- The basic process involves a lowpass filter (l[n]) and a highpass filter (h[n])
- The image is processed in 4 ways (producing 4 separate images as output). Its rows are convolved with l or h, as are its columns.
- The 4 image outputs have l rows and l columns, h rows and l columns, l rows and h columns, and h rows and h columns.
-

Wavelet Experimentation: The Wavelet Transform

- These images are then downsampled by 2, meaning that every other row and every other column is eliminated.
- The 4 images are then combined into one having the same dimensions as the original source image
-

Low Pass Rows Low Pass Cols	Low Pass Rows High Pass Cols
High Pass Rows Low Pass Cols	High Pass Rows High Pass Cols

Wavelet Experimentation: The Wavelet Transform

- The inverse wavelet transform must break down the four-part image and up sample by 2, inserting a row of zeros after every row and a column of zeros after every column. Next, the 4 images have their rows and columns convolved with the same filters (lowpass or highpass) as before. And lastly, the 4 component images are summed to regain the original image.
- 

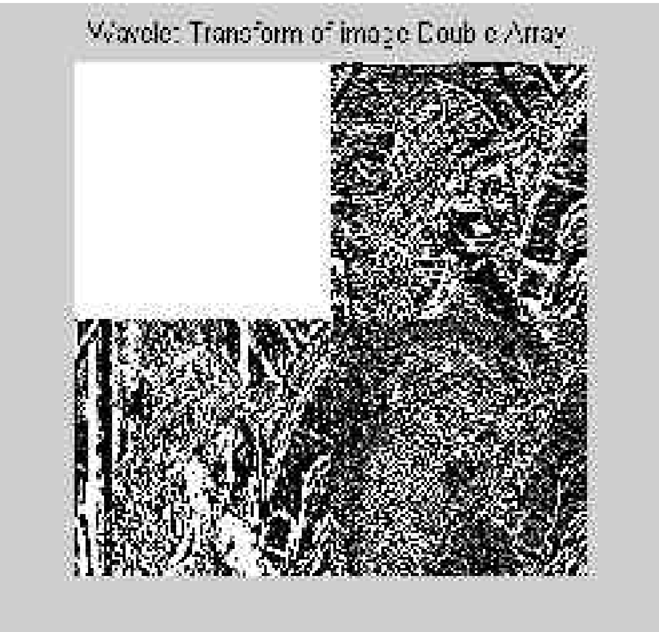
Wavelet Experimentation: The Wavelet Transform

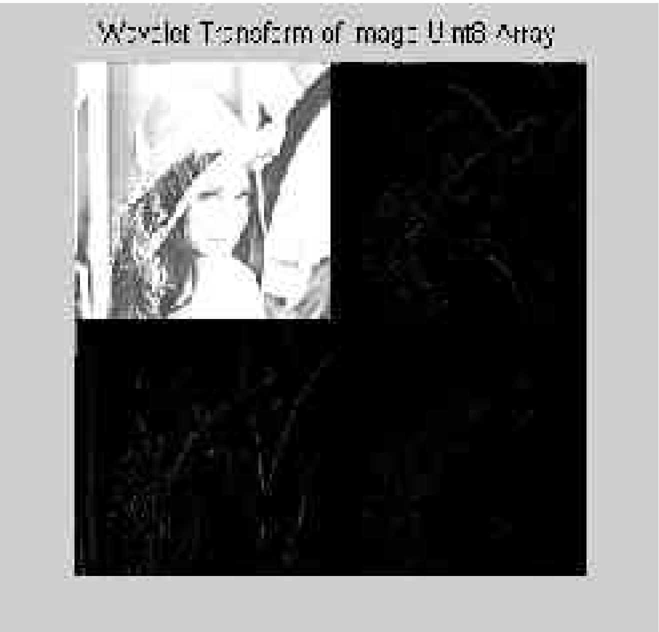
Data Hiding

Like the DCT experiment the hidden data (payload) was a simple text message

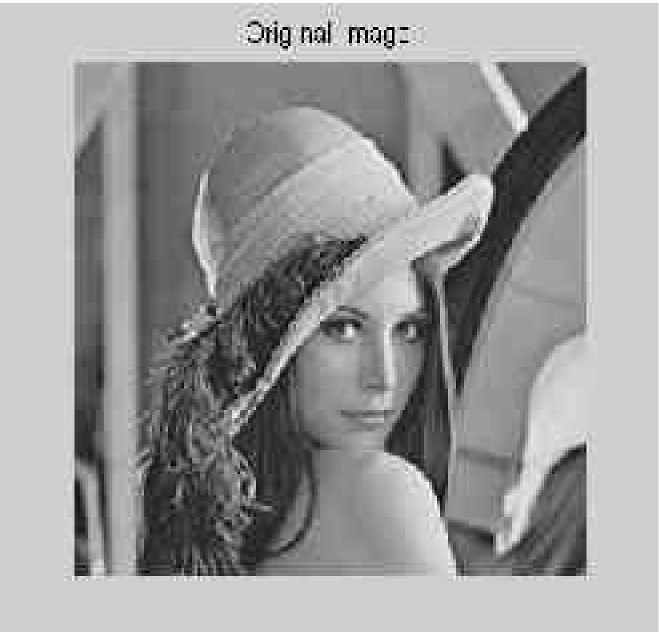
Encoding the information in only one of the four quadrants created by the Wavelet Transform

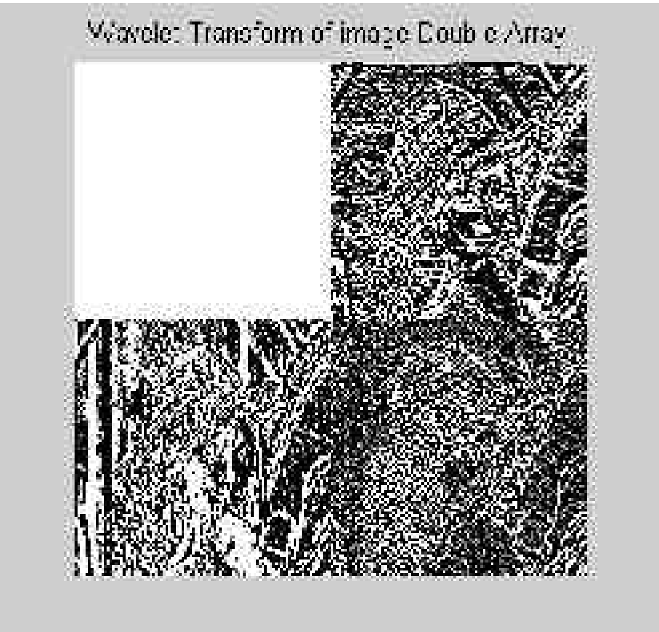
This was done in the lower right quadrant of the WT (high pass across rows and columns) because it was **less visible** than any of the other three quadrants.





Low Pass Rows Low Pass Cols	Low Pass Rows High Pass Cols
High Pass Rows Low Pass Cols	High Pass Rows High Pass Cols





Hidden message. This is a simple hidden message in text format.

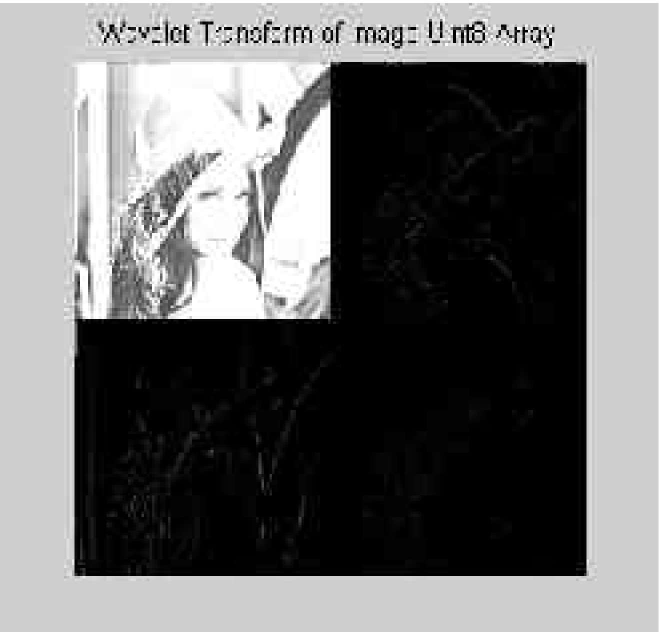
↓

Converted to ASCII Numbers and added to the coefficients along the Rows

↓

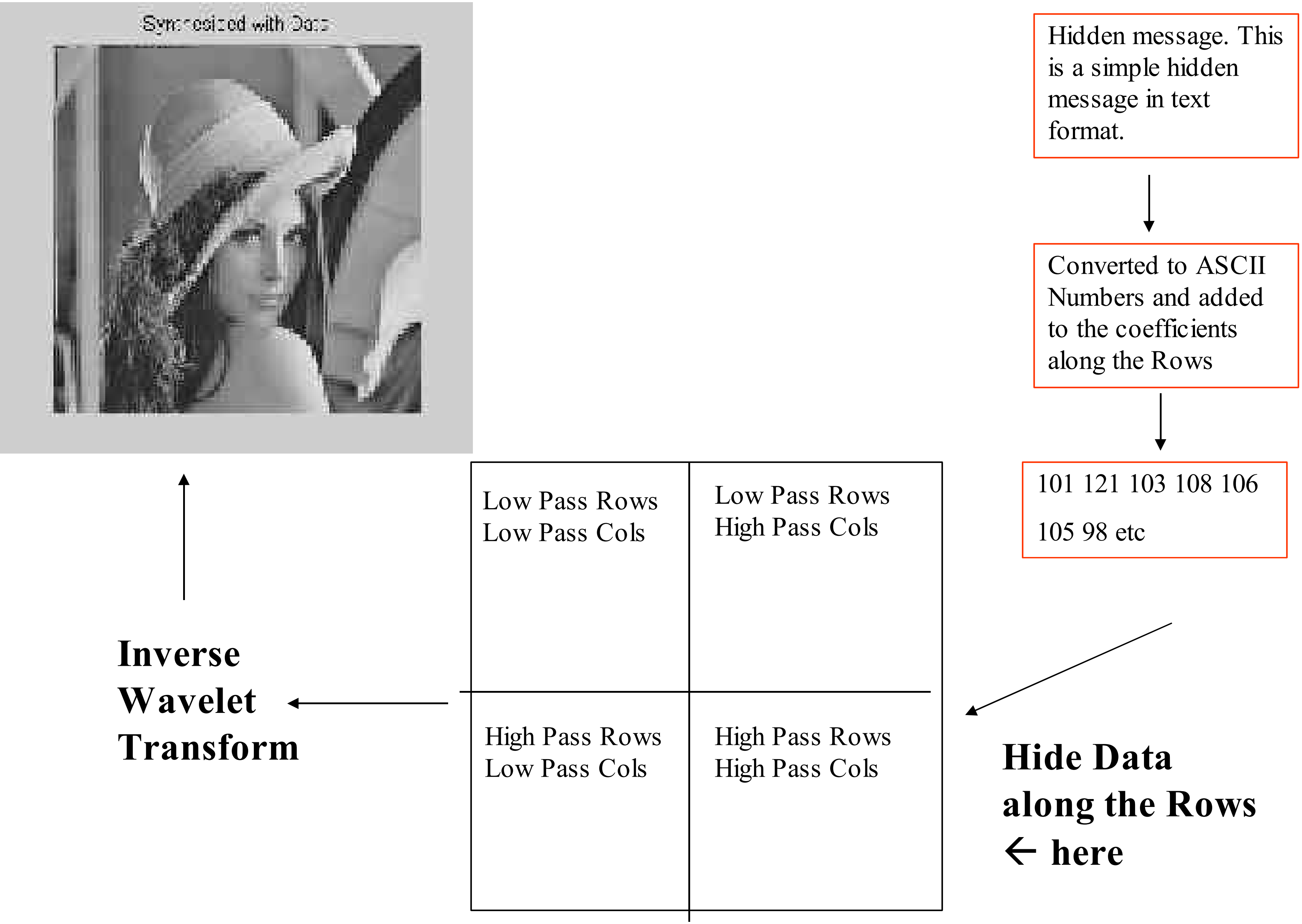
101 121 103 108 106

105 98 etc



Low Pass Rows Low Pass Cols	Low Pass Rows High Pass Cols
High Pass Rows Low Pass Cols	High Pass Rows High Pass Cols

Hide Data along the Rows ← here



Hidden message. This is a simple hidden message in text format.

Converted to ASCII Numbers and added to the coefficients along the Rows

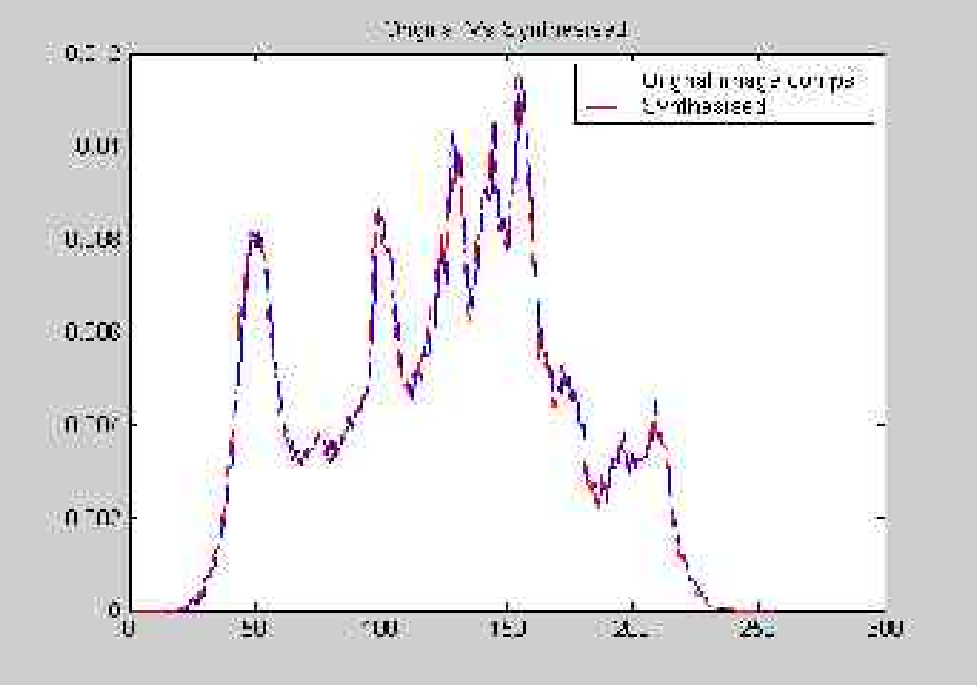
101 121 103 108 106

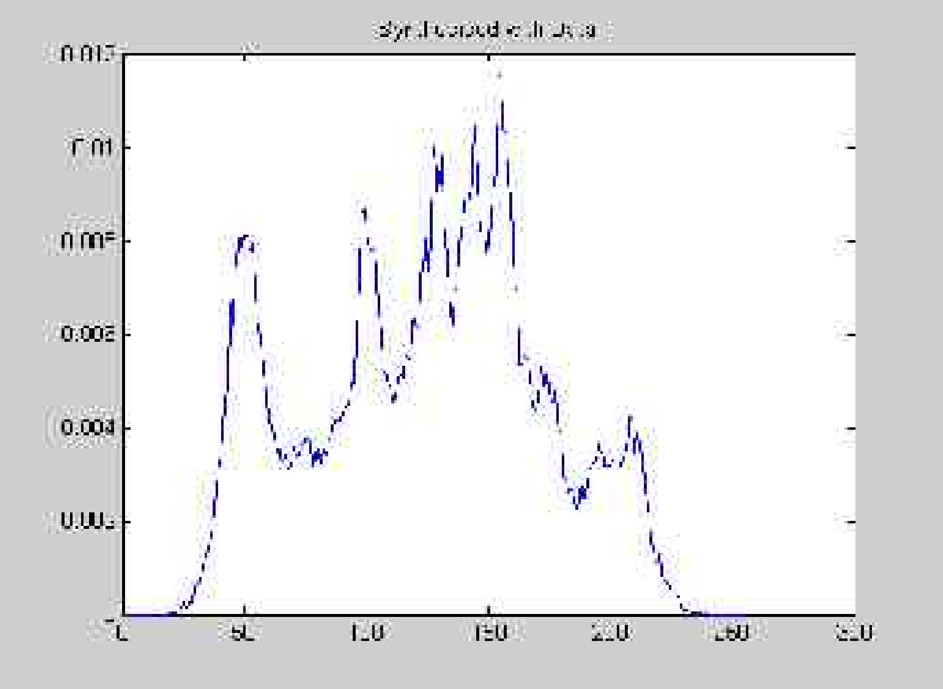
105 98 etc

Hide Data along the Rows ← here

Low Pass Rows Low Pass Cols	Low Pass Rows High Pass Cols
High Pass Rows Low Pass Cols	High Pass Rows High Pass Cols

Inverse Wavelet Transform





Results:

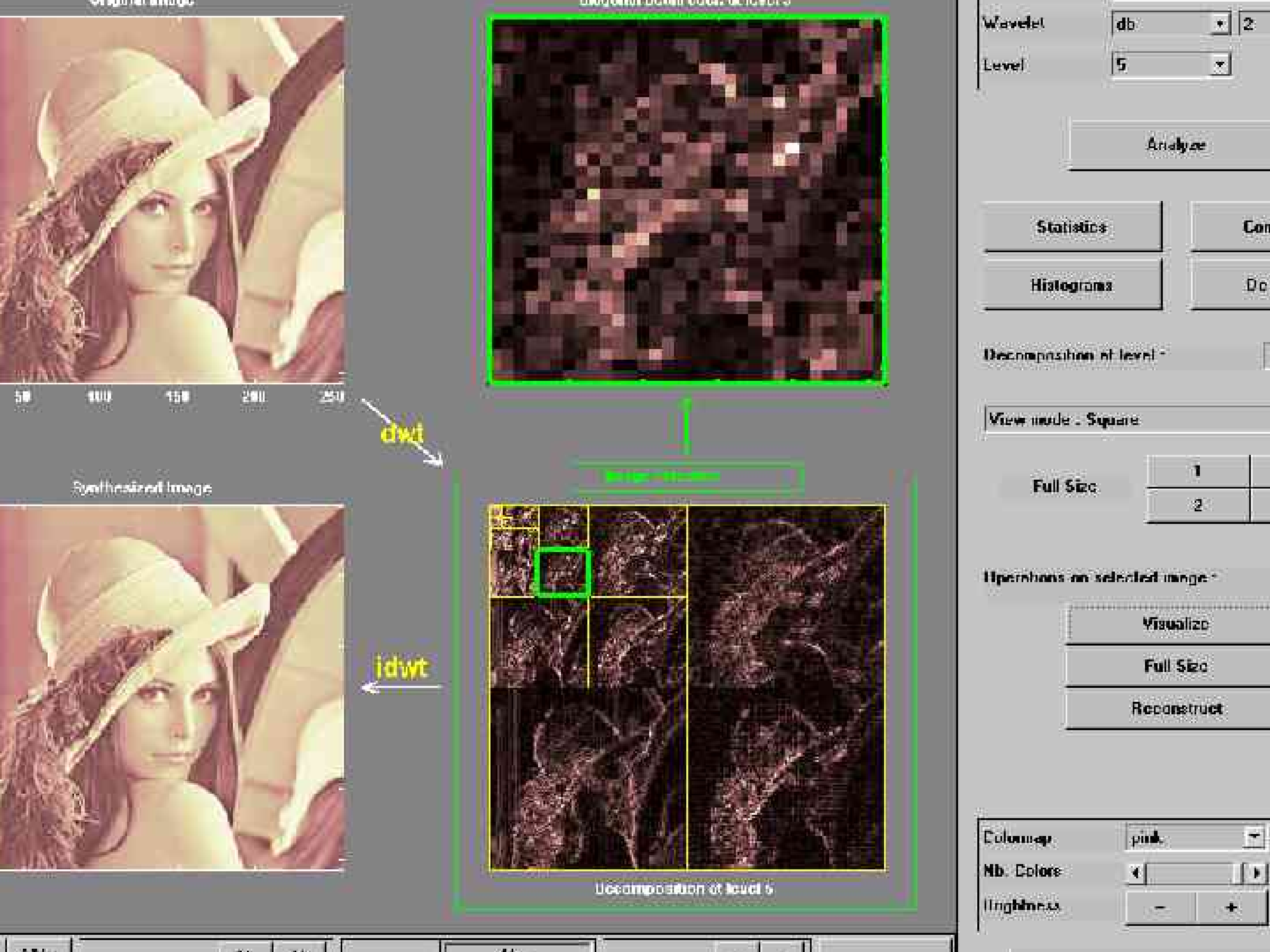
Using this my algorithm we can see the synthesised image differs slightly from original

But wavelets method for Hiding is very good, ‘near perfect’ synthesised with Data

This method cannot be used tough as synthesised image is too distorted, and would draw suspicion.

Future work, to get ‘perfect’ synthesised image.

- **Future Work…**
- Again Increasing Payload Vs Image Degradation
- Successfully Data Extraction Vs Payload
- Inserting payload in a different manor (transform)
- Adding noise to Images (steganalysis) to attack the hidden message
- JPEG compression,
- Translation
- Rotation
- Use of Wavelet Toolbox
    - Haar
    - Db2 (Daubechies)
    - All At Different Levels (up to 5)



Wavelet db 2

Level 5

Decomposition at level :

View mode : Square

Operations on selected image :

Decomposition at level 5

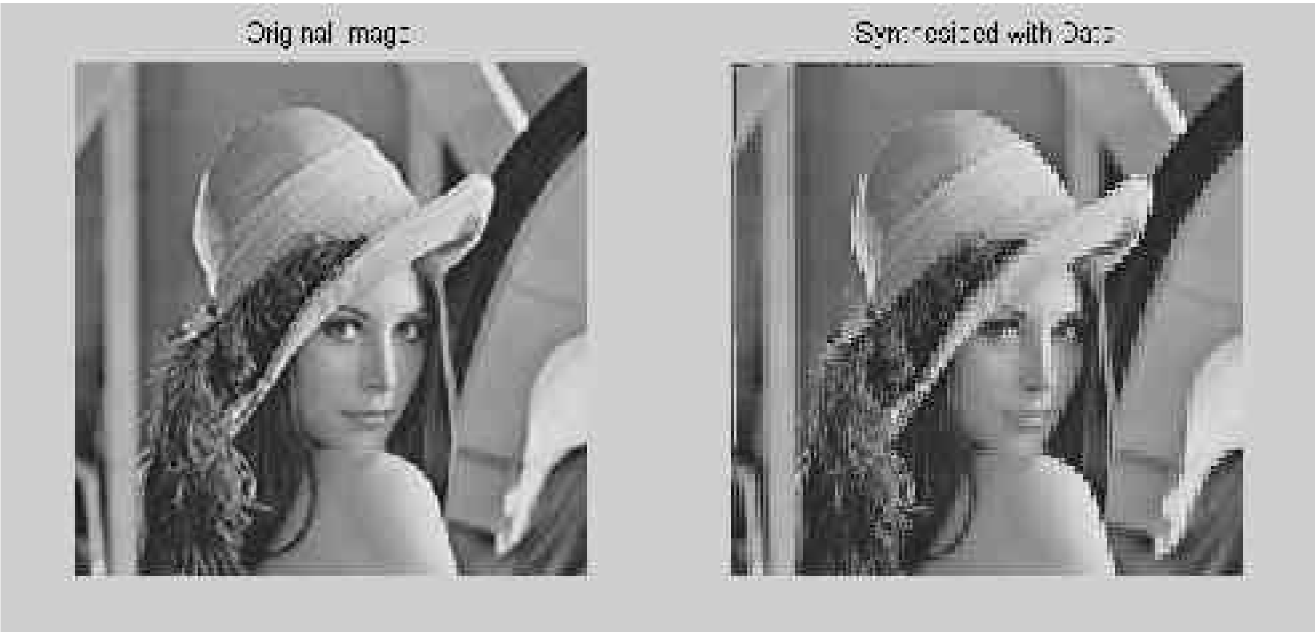
Colormap pink

Thank You!

Questions ?

Image High pass Rows High Pass Columns

Image High pass Rows High Pass Columns with Data



- As shown, there is slight difference between original and Synthesised images.
- My wavelets algorithm needs to be improved