

HASIL CEK_60020397_Point- C71-IRD-850GB-Comparative Analysis of Image Steganography using SLT, DCT and SLT-DCT Algorithm

by Imam Riadi 60020397

Submission date: 11-Dec-2020 11:05AM (UTC+0700)

Submission ID: 1471717230

File name: of_Image_Steganography_using_SLT,_DCT_and_SLT-DCT_Algorithm.pdf (814.78K)

Word count: 5165

Character count: 26887

Comparative Analysis of Image Steganography using SLT, DCT and SLT-DCT Algorithm

Lilik Widyawati¹, Imam Riadi², Yudi Prayudi³

¹Universitas Bumigora, Indonesia

²Universitas Ahmad Dahlan, Indonesia

³Universitas Islam Indonesia, Indonesia

Article Info

Article history:

Received, 16 April 2020

Revised, 7 September 2020

Accepted, 20 September 2020

Keywords:

Slantlet Transform

Discrete Cosine Transform

PSNR

MSE

SLT-DCT

ABSTRACT

Steganography is an interesting science to be studied and researched at this time, because steganography is the science of hiding messages on other digital media so that other parties are not aware of the existence of information in the digital media. Steganography is very effective in maintaining information security, because the existence of this information is obscured so that it is difficult to know where it is. This paper discusses hiding text into images using the Slantlet Transform (SLT) method, Descarte Cosine Transform (DCT) and Hybrid of SLT and DCT. The three methods are implemented in the frequency domain where steganographic imagery is transformed from the spatial domain to the frequency domain and the message bit is inserted into the cover image frequency component. The comparison parameters of these three techniques are based on MSE, PSNR, Capacity & Robustness. From the results of the tests that have been done, it is obtained that the highest PSNR value is generated using the SLT-DCT method, the largest storage capacity is the SLT method while the resistance, SLT-DCT method and DCT method are more resistant to attack than the SLT method.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Lilik Widyawati,

Department of Computer Science,

Universitas Bumigora.

Email: lilikwidya@universitasbumigora.ac.id

1. INTRODUCTION

Information security at this time is very important because it is easy to access data exchange to make the data and information must be protected so that its existence is not known by parties who are not interested. Techniques for maintaining the security of digital information can be done in several ways, one of which is by hiding the information into other digital media, this technique is called steganography. One type of files that are widely used and generally contain important information is digital image. Currently image has been used in almost all areas such as security plans, medical sciences, engineering sciences machinery, architectural buildings, works of art, advertising, education and so forth [1]

There are two general stages in digital steganography, namely the process of embedding or encoding and the process of extracting or decoding (expressing). The results obtained after the embedding or coding process are called Stego Objects (if the media container is only in the form of image data, it is called Stego Image) [2].

The image of the steganographic technique is divided into two, namely the spatial domain and transformation domain [3]. Spatial domain techniques insert secret bits directly in the cover file, the spatial domain technique commonly used is Least Significant Bit Insertion (LSB). In LSB, the secret bits are inserted into the least significant image closing bits. The domain transform hides secret bits in important parts of the

cover file. Whereas the Transform Domain technique tries to encode the message bits in changing the domain coefficients of the image. This technique can realize a large embedding capacity for steganography [4].

The following is the design of the data insertion process in general.

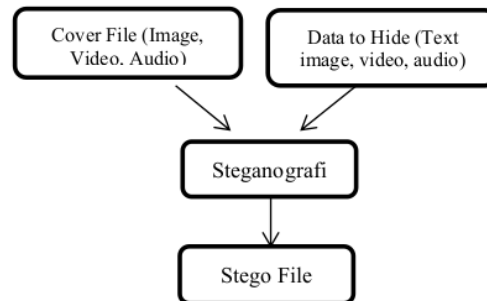


Figure.1 The process of hiding data

The picture above is a data insertion process, the cover file is inserted data to be hidden which is called steganography then produces Stego file.

An image steganography technique is said to be good if it meets 3 criteria, namely the image quality of steganography is still quite good (imperceptibility), resistant to basic image operations such as adding contrast and changing robustness, and inserted messages can be recovered [5].

The test parameters for image quality are PSNR and MSE. PSNR- Peak signal to noise ratio is calculated usually in logarithmic (dB) scale is a metric use to measure the quality of any image reconstructed, restored or corrupted image with respect to its reference or ground truth image, it is a full reference image quality measure defined as the maximum value of maximum signal power with respect to MSE (Mean square error) assumed as noise power, similarly MSE can be calculated as the square difference between reference image and reconstructed image [6].

Chandran & Bhattacharyya discussed the comparison of DCT, SLB and DWT algorithms and revealed higher PSNR from DCT compared to the other two algorithms. Thus, the experimental results show that the DCT algorithm is more suitable for steganography applications compared to LSB and DWT based algorithms [7].

Vaishali & Batt are concerned about the comparative study of DCT and the DWT Method. Both are under the transformation of domain analysis, both methods have good imperceptibility and Robustness against statistical attacks. But as we know the main purpose of Steganography is to increase resistance to attack and also to increase cargo capacity. The method we propose increases the PSNR and also the less capacity in the DCT Method [8].

Kundra & Madaan discussed different image Steganography techniques and their comparison were discussed so that one can choose best method for hiding the secret information. Different proposed techniques in the literature have been discussed and some of techniques results in high quality of image while some are more secure another. By analyzing all the techniques Kundra & Madaan found that performance of the Hash-LSB would be more secure than other techniques as earlier discussed, Hash-LSB is combination of two technologies one from cryptography and another from Steganography. RSA algorithm itself is very secure that no one can break it easily [9].

From the literature study above, the writer tries to do research related to some existing algorithms in the transform domain technique, namely using the Slantlet Transform (SLT), Descrate Cosine Transform (DCT) and Hybrid of SLT and DCT methods, because there has never been a comparison between when the method and parameter comparison of the three methods based on MSE, PSNR, Capacity & Robustness.

2. RESEARCH METHOD

2.1. Sample Data

The data used in this study are digital image file images taken from the site <http://sipi.usc.edu/database/database.php>, SIPI laboratory database (Signal and Image Processing Institute) USC (University of South California). USC-SIPI is a collection of digital images to support image processing and image analysis research. In this study the type of image used is color image, with the tiff format (*. Tiff) and square image size, namely the length and width of the same image. While the secret message file to be used is a file with text (*. txt) format.

The image used in this study is a type of image, namely Color Image (RGB). The database is divided into several volumes based on the basic character of the image. Images in each volume have sizes such as 256x256 pixels, 512x512 pixels, or 1024x1024 pixels. So that the Sample Cover Image is obtained with a tree with an image resolution of 256x256 pixels, and a file size of 193 kb and Airplane (F-16) with an image resolution of 512x512 pixels, and a file size of 769 kb and San Diego that has a resolution of 1024x1024 pixels file size of 3 Mb.

2.2. Steganography of SLT

The SLT algorithm is a method development from DWT where SLT has a better localization time than DWT because it supports shorter filter components. DWT is usually implemented in the form of an iterated bank with a tree structure, but SLT is inspired by the form of a parallel structure with parallel branches [10].

2.2.1. Encode Process

In the insertion process using the SLT algorithm, the cover image will be broken down into 4 sub-bands namely LL, HL, LH, HH, and then the LL subband will be selected as the insertion location, $l(x, y)$ is the image data coefficient to be modified, α is embedding strength factor, in the bit insertion process using an algorithm:

1. If the bit is equal to "1" then the coefficient $l(x, y)$ is equal to the coefficient $l(x, y)$ plus α .
2. If the bit is equal to "0" then the coefficient $l(x, y)$ is equal to the coefficient $l(x, y)$ minus α .

After the algorithm is run, then the coefficient is returned to the original SLT inverse.

2.2.2. Decode Process

The extraction process is done by seeing if the coefficient of $l(x, y)$ is more than "0" then the message bit is equal to "1", if the coefficient $l(x, y)$ is less than "0" then the message bit is equal to "0". Then the resulting message bits are combined and then converted into the original message form.

2.3. Steganography of DCT

Discrete Cosine Transform is a technique for converting a signal into its basic frequency components. Discrete Cosine Transform represents an image of the sum of the sinusoids of varying magnitude and frequency [11].

2.3.1. Encode Process

The steganography algorithm used is the DCT algorithm, the inserted message is a message that has been converted into binary form, the insertion process uses different pixel values ($c1$) and ($c2$) in each image block, the program will first read one block and convert it to DCT, then based on the message bit the following algorithm is used:

1. For the message bit "0", if the value at the pixel location ($c1$) > ($c2$) is greater then the two pixels are exchanged.
2. For the message bit "1", if the value at the pixel location ($c1$) < ($c2$) is smaller then the two pixels are exchanged.

After the algorithm is executed, then the block is returned to the original inverse DCT

2.3.2. Decode Process

The extraction process of the message is almost similar to the message insertion process, namely each block is diversified with DCT, but in the extraction process only checks the greater between the values of pikes ($c1$) and ($c2$), this is to determine the 0 and 1 bit values, as in the algorithm below this:

1. If the pixel value (c_1) $<$ (c_2), the message bit represented is "0".
2. If the value of pikes (c_1) $>$ (c_2), the message bit represented is "1".

Then the resulting message bits are combined and then converted into the original message form.

2.4. Steganography of SLT-DCT

The next method is the combined method of SLT and DCT algorithm plays a role in decomposing the cover image into LL, Lh, HL, and HH Subbands, then the embedding process used the DCT algorithm on the highest subband namely HH [12].

2.4.1. Encode Process

In the SLT-DCT steganography insertion process, the first thing to do is apply the SLT algorithm to the cover image to decompose it into four sub-bands: LL, HL, LH and HH. After the process of decomposing the cover image into four sub-bands, then apply DCT to the sub-band (HH) to obtain the DCT coefficient.

Then insert it with an algorithm:

1. For the message bit "0", if the value at the pixel location (c_1) $<$ (c_2) is greater then both pixels are exchanged.
2. For the message bit "1", if the value at the pixel location (c_1) $>$ (c_2) is greater then the two pixels are exchanged.

After the algorithm is run, then the block is returned to the original inverse DCT, then the last one is inverse SLT (ISLT) to produce stego image.

2.4.2. Decode Process

In the extraction process apply SLT to decompose the stego image into four sub-bands: LL, HL, LH and HH. Then apply DCT to each block in the selected sub-band (HH), then extract the DCT block coefficient. Then the resulting message bits are combined and then converted into the original message form. So that the same message is obtained as previously inserted.

2.5. System Testing Design

The following is a general design of system testing to be carried out. There are three things to consider when designing a steganography system: (a) Invisibility: The human eye cannot distinguish the difference between the original and stego images. (b) Capacity: The more data the image can hide better. However, large confidential information can significantly reduce image quality [9]. (c) Robustness: stego imagery is able to withstand attacks or manipulation of stego image data so that the message is maintained [10].

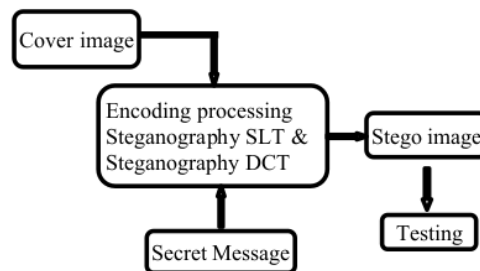


Figure. 2. General system design

In the picture above, the test design process is explained in general, and Stego Image results from the SLT and DCT algorithms will be tested. In the message insertion process, the first process, namely the cover image, will be inserted into the message using the SLT and DCT steganography methods, resulting in a stego image, then each stego image will be tested for image quality, message capacity, and security, so the comparison results are obtained. SLT, DCT and SLT-DCT steganography.

2.6. Image Quality Testing

Image quality assessment is done by comparing the PSNR value from the stego image of each algorithm.

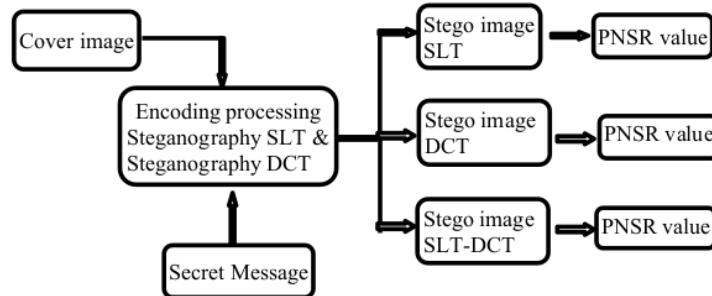


Figure.3. Image Quality Testing

From the picture above, the flow of the message capacity testing process is explained, the cover image will be inserted into a message using SLT and DCT steganography, then it will produce a stego image which will then calculate the value of PNSR from each of the stego images.

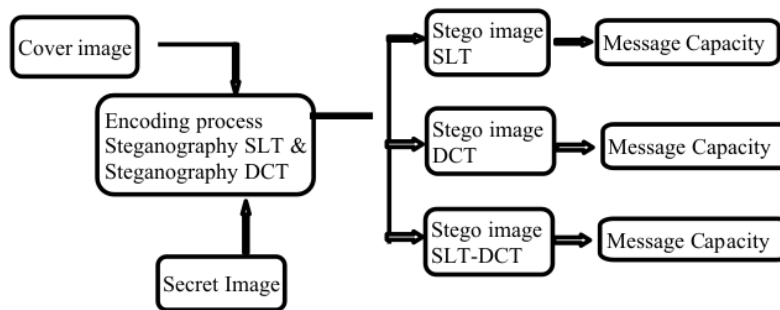


Figure.4. Message Capacity Testing

From the picture above, the flow of the message capacity testing process is explained, the cover image will be inserted into a message using SLT, DCT and SLT-DCT steganography, then it will produce a stego image which will then calculate the maximum message capacity of each stego image.

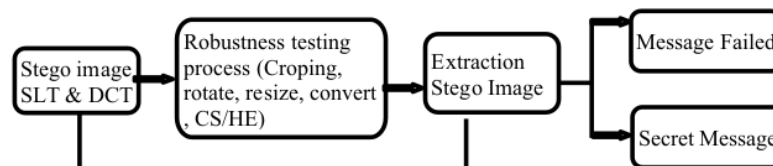


Figure.5. Robustness testing of image

From the picture above, the flow of the image resilience testing process is explained, where the stego image from SLT and DCT steganography will be manipulated image data such as [13]:

1. Cropping is the process of removing some of the existing images so that the parameters of the existing LSB are changed or modified causing the digital content to be damaged in the structure.
2. Rotation is a process to change the position of an image according to the degree of slope to be determined. This process does not cause damage to digital content inside.
3. Resize is the process of changing the area of an image to be larger or smaller than its original size, in this case changing the size can result in a shift in the existing color values and LSB so that with changes in the value of the parameter it also changes the digital content inside.

4. Convert (conversion) is the process of changing data from a bitmap or BMP format to JPG or vice versa. In the process if done, it can result in changes in the color value parameters contained in the image due to the compression process.
5. Contrast stretching is one of them by scanning the histogram from the smallest gray value to the largest gray value (0 to 255) to find the lowest gray value limit and the highest gray value limit from the pixel group (image).

Histogram equalization aims to distribute histograms evenly, so that each gray value has a relatively equal number of pixels.

3. RESULTS AND ANALYSIS

3.1. Output Analysis

Important information about the contents of a digital image can be revealed by creating an image histogram. Histograms can also show a lot of information about the brightness and contrast of an image. Histogram may change RGB image into grayscale one. Therefore, histogram is a valuable tool in image processing work both qualitatively and quantitatively [14].

1. Steganography of SLT and SLT-DCT

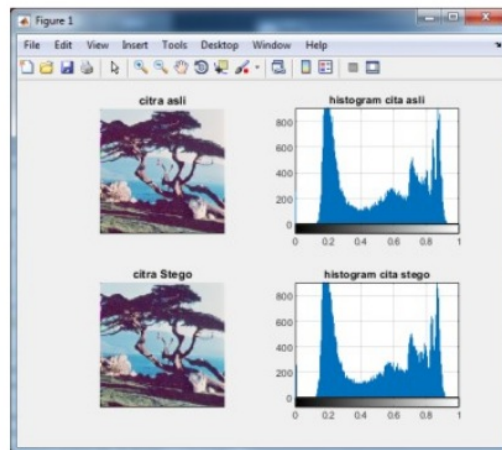


Figure.6. Comparison of the results of message insertion on the Tree using SLT Algorithm

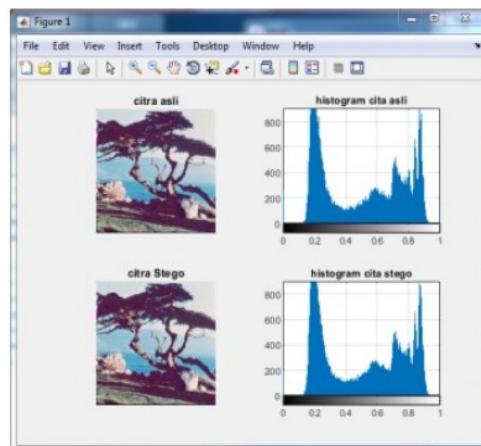


Figure. 7 Comparison of the results of message insertion on the Tree image using SLT-DCT Algorithm

The picture above shows a comparison of the results of the original image ² histogram and SLT steganographic image and SLT-DCT steganography, it can be seen histogram differences between the original

image and steganographic image, in plain view there is no clear difference between the original image and steganographic image but on the histogram there is a difference between the histogram original image and steganographic image histogram.

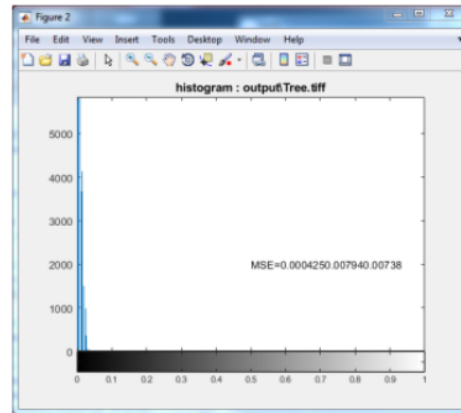


Figure.8. Histogram of image differences in the Tree using SLT Algorithm

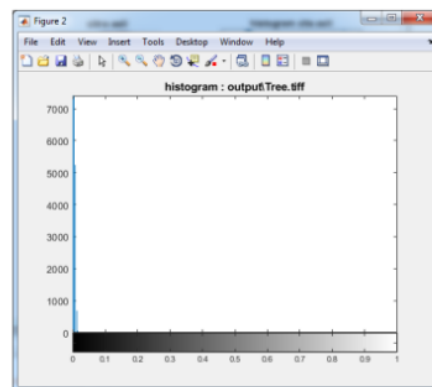


Figure. 9. Histogram of image differences in the Tree using SLT-DCT Algorithm

The picture above shows the results of differences in the image tree histogram using the SLT and SLT-DCT algorithms. From the picture above it can be seen clearly the difference in histogram between Steganography using SLT algorithm and steganography using the SLT-DCT algorithm, the histogram error produced by the SLT-DCT algorithm is less than the SLT algorithm which means the difference between the original image and fewer steganographic images.

2. Steganography of DCT and SLT-DCT

The output of the histogram of DCT steganography and SLT-DCT steganography with the same message size is 30 characters in the Tree image, as follows:

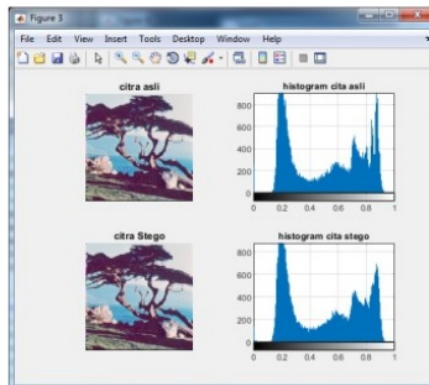


Figure. 10. Comparison of the results of message insertion on the Tree image using DCT Algorithm

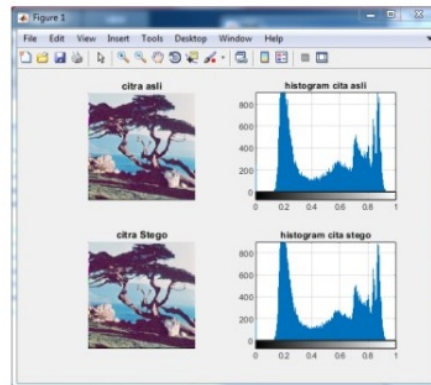


Figure. 11 Comparison of the results of message insertion on the Tree image using SLT-DCT Algorithm

The picture above shows a comparison of the results of the original image histogram and DCT steganographic image and SLT-DCT steganography, it can be seen histogram differences between the original image and steganographic image, in plain view there is no clear difference between the original image and steganographic image but on the histogram there is a difference between the histogram original image and steganographic image histogram.

Comparison of error histograms in Tree images on steganography using the DCT algorithm and the SLT-DCT algorithm as follows:

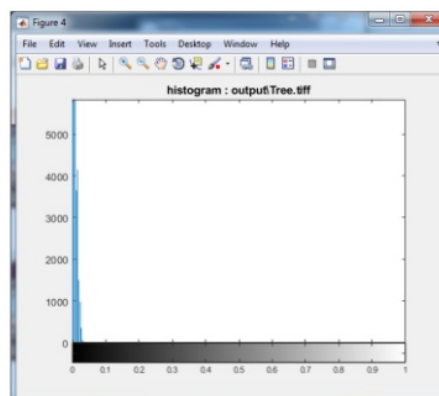


Figure. 12. Histogram of image differences in the Tree using DCT Algorithm

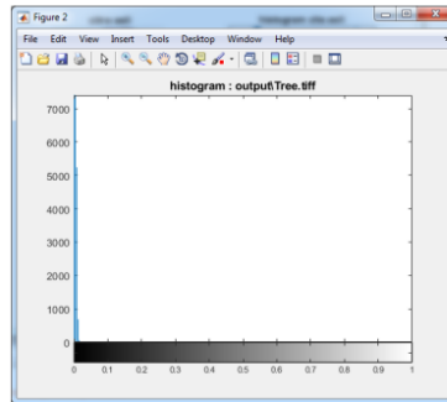


Figure. 13. Histogram of image differences in the Tree using SLT-DCT Algorithm

The picture above shows the results of comparison of image tree histograms using the SLT and SLT-DCT algorithms. From the picture above, it is clear that the histogram difference between Steganography using the DCT algorithm and steganography using the SLT-DCT algorithm, the histogram error produced by the SLT-DCT algorithm is less than the DCT algorithm which means the difference between the original image and fewer steganographic images.

3.2. Comparison of PSNR and MSE Values

In this test the system was tested by inserting a number of different message characters on the cover of 256x256 Tree image, 512x512 Airplane and San Diego 1024x1024 using SLT, DCT and SLT-DCT algorithms to determine the effect of PSNR and MSE values on stego image

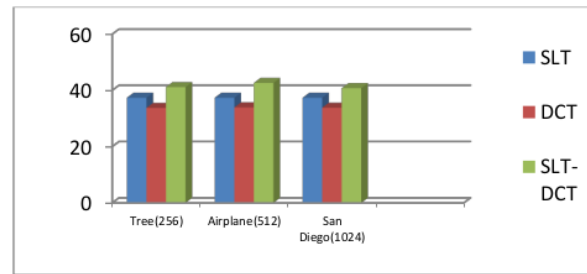
Table 1 The results of the comparison of the quality of images inserted are 30 characters

Size Cover image	Algorithm	PSNR (db)	MSE
Tree(256x256)	SLT	37,3313	0,0001849
	DCT	33,7112	0,0004254
	SLT-DCT	41.13605	0.0007698
Airplane(512x512)	SLT	37,3392	0.00018454
	DCT	33,9062	0.00040689
	SLT-DCT	42.61077	0,0005481
San Diego (1024x1024)	SLT	37,3392	0.00018454
	DCT	33,7955	0,0004173
	SLT-DCT	40.75013	0,0008414

From the table above, it is known that the comparison of PNSR and MSE values with tiff format images are Tree images that have 256x256 pixel resolution, Airplane with 512x512 pixel resolution, and San Diego imagery with 1024x1024 pixel resolution that has 30 characters inserted, the highest PNSR of each image is visible the insertion using the SLT-DCT Algorithm is 41.13605 on the Tree image, 42.61077 on the Airplane image, and 40.75013 in the San Diego image.

3.3. Testing the Effect of the Large Cover Image

From the test results that have been compared the quality of the image inserted message 30 characters cover images with sizes 256x256, 512x512, and 1024x1024 using the SLT, DCT and SLT-DCT algorithms.



Figur.8. Graph Comparison of PSNR values with the same number of messages

From the graph above, it can be seen that the value of PSNR from each algorithm above 30 dB and the highest PSNR value in each image inserted using the SLT-DCT algorithm with a value of PSNR > 40 dB. If the PSNR value falls below 30 dB, it indicates the image quality is relatively low, where the distortion caused by insertion is clearly visible. However, the high quality of the stego image is at a value of 40 dB and above [15].

3.4. Testing the Effect of the Magnitude of the Image Cover on the maximum character message

In this test the system was tested by inserting a number of different message characters on the cover of 256x256 Tree image, 512x512 Airplane and San Diego 1024x1024 using SLT, DCT and SLT-DCT algorithms to determine the effect of PSNR and MSE values on stego image.

Table 2. Comparison of PSNR and MSE values based on the size of the cover image

Algorithm	Cover image size	Character max	PSNR	MSE
SLT	Tree	2048 Characters	37.2584	0.0001880
	Airplane	8192 Characters	37.3390	0.00018454
	San Diego	32.768 Characters	37.3390	0.00018454
DCT	Tree	128 Characters	33.8337	0.00041365
	Airplane	512 Characters	33.9053	0.00040689
	San Diego	2048 Characters	33.7710	0.00041966
SLT-DCT	Tree	32 Characters	41.13472	0.00077007
	Airplane	128 Characters	42.6226	0.00054669
	San Diego	512 Characters	40.74333	0.00084269

From the table above, it can be seen in the SLT algorithm, the insertion of characters with the number 2024 on the cover image tree 256x256 has MSE = 0.0001880 and PSNR = 37.2584, and characters with the number 8192 on the cover image Airplane 512x512 have MSE = 0.00018454 and PSNR = 37.3390, this shows the greater the cover image, the smaller the MSE value and the greater the value of PSNR, in the DCT and SLT-DCT algorithms for the cover image San Diego.

3.5. Testing Message Capacity

In SLT steganography, message capacity testing is done by dividing the image pixels into 2x2 blocks, then the results of the bits obtained are divided by 8, so that the maximum number of characters is obtained from an image.

In DCT steganography, capacity testing is done by calculating the maximum capacity value of an image, this is done by dividing image pixels into blocks 8x8, then the results of the bits obtained are divided by 8, so that the maximum number of characters can be obtained by an image

In SLT-DCT steganography, message capacity testing is done by dividing image pixels in 2x2 blocks, then divided into 8x8 blocks and the results of bits obtained are divided into 8 so that the maximum number of characters can be obtained by an image

Table 3. Table of message capacity for SLT, DCT and SLT-DCT steganography

Image	Size	SLT	DCT	SLT-DCT
Tree	256 x 256	2048 Characters	128 Characters	32 Characters
Airplane	512 x 512	8192 Characters	512 Characters	128 Characters
San Diego	1024 x 1024	32.768 Characters	2048 Characters	512 Characters







From the table above, it is known that the SLT algorithm has the most message storage capacity in each image, such as Tree 256x256 can store 2046 characters, as easy as the SLT-DCT algorithm on 256x256 Tree can only accommodate 32 characters so the lowest traffic is owned by the SLT-DCT algorithm.

3.6. Robustness Testing of Image

Stego image robustness testing is done by manipulating data on images, such as cropping, which removes some image data, Rotate that is rotating the image, Resizing is changing the image size, convert is image format, CS is contrasting stretching the image and finally adding Histogram on the image.

The table below will explain the results of endurance testing on steganographic images using the SLT, DCT and SLT-DCT algorithms

Table 4. Robust image in Steganography of SLT-DCT

Process Testing	Image	Size (byte)	Extraction	Information
Cropping		122,534 bytes	Failed	Eliminating othes parts of the image
Rotate		194,058 bytes	Failed	Rotate the Image toward 180 degrees
Resize		49,742 bytes	Failed	Rezise 256x256 pixels menjadi 128x128 pixels
Convert		136,711 bytes	Succesful	Change the format from tiff to PNG
CS		197,708 bytes	Succesful	Add contrast stratching to the image
HE		197,686 bytes	Succesful	Perform a Histogram equalization process

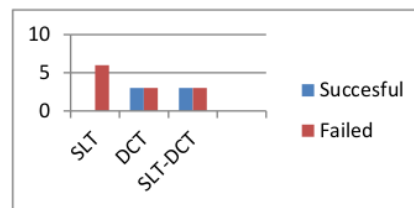


Figure 14. Graph Comparison of Results of analysis of SLT, DCT and SLT-DCT steganographic image resistance

From the table above, it can be seen from the six attempted manipulations, three experiments that did not damage the message which meant the original message could be extracted, the stego image generated from SLT-DCT steganography was resistant to conversion, added contrast stretching and histogram equalization.

From the table and figure above, it can be seen from the six attempts of manipulation carried out, in SLT steganographic images all attacks or manipulations of image data carried out can damage the messages contained in the steganographic image so that the messages in the image cannot be returned to their original form, while the process test in DCT steganography images, from 6 process tests there are 3 successful process tests or steganographic images resistant to 3 attacks including the Convert process, the addition of Contrast Stretching, and the Histogram Equalization. While the process test on SLT-DCT steganography with DCT steganography is able to withstand 3 attacks, namely Convert, Contrast Stretching and Histogram Equalization

4. CONCLUSION

From the results of experiments that have been done on the SLT-DCT steganographic application that has been proven to prove that SLT-DCT steganography works well, messages that have been inserted in the image can be returned or extracted in its original form and SLT-DCT steganography applications can increase the PSNR value of the image steganography when compared with SLT and steganografi DCT Steganography with the increase of PSNR value, steganographic image is increasingly difficult to distinguish from the original image so that the possibility of messages in the steganography image is more difficult to know and the PSNR value of more than 40 db proves image quality The resulting steganography is very good, with a PSNR value of more than 40 db, the quality of steganographic images is in a good category and is more resistant to multiple attacks, so in terms of the resistance of SLT-DCT steganography is able to withstand 3 attacks which means there is an increase in image resilience.

REFERENCE

- [1] P. Irfan, Y. Prayudi, and I. Riadi, "Image Encryption using Combination of Chaotic System and Rivers Shamir Adleman (RSA)," *International Journal of Computer Applications (0975 – 8887)*, vol. 123, no. 6, pp. 11–16, 2015.
- [2] Y. Prayudi and P. Kuncoro, "Implementasi Steganografi Menggunakan Teknik Adaptive Minimum Error Least Significant Bit Replacement (AMELSB)," in *Seminar Nasional Aplikasi Teknologi Informasi 2005*, 2005, pp. 1–6.
- [3] T. Morkel, M. S. Olivier, and S. Africa, "An Overview Of Image Steganography," in *Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA 2005)*, 2005, pp. 1–12.
- [4] D. Bansal and R. Chhikara, "An Improved DCT based Steganography Technique," *International Journal of Computer Applications (0975 – 8887)*, vol. 102, no. 14, pp. 46–49, 2014.
- [5] R. Munir, *Pengolahan Citra digital dengan Pendekatan Algoritmik*. Bandung: Informatika, 2004.
- [6] Z. Alqadi, "Comparative Analysis of Color Image Steganography," *International Journal of Computer Science and Mobile Computing*, vol. 5, no. 11, pp. 37–43, 2016.
- [7] S. Chandran and K. Bhattacharyya, "Performance analysis of LSB, DCT, and DWT for digital watermarking application using steganography," in *2015 International Conference on Electrical, Electronics, Signals, Communication and Optimization*, Jan. 2015, vol. 1, pp. 1–5.
- [8] P. Vaishali and P. Bhat, "Transform Domain Techniques for Image Steganography," *International Journal Of Innovative Research In Electrical, Electronics, Instrumentation And Control Engineering*, vol. 3, no. 1, pp. 65–68, 2015.
- [9] H. Arora, C. Bansal, and S. Dagar, "Comparative study of image steganography techniques," in *2018 International Conference on Advances in Computing, Communication Control and Networking*

- (ICACCCN), Oct. 2018, vol. 3, no. 4, pp. 982–985.
- [10] R. Wissarto, "Implementasi Slantlet Transform (SLT) Dan Huffman Coding Pada Steganografi Citra Grayscale," 2014.
- [11] A. A. Faruqi and I. F. Rozi, "Implementasi Steganography Menggunakan Algoritma Discrete Cosine Transform," *Jurnal Informatika Polinema*, vol. 2, no. 1, p. 35, Mar. 2017.
- [12] L. Widyawati, "Penerapan Metode Steganografi SLT dan DCT Pada Citra dengan Contrast Stretching dan Histogram Equalization untuk Meningkatkan Kapasitas Pesan Lilik Widyawati," 2018.
- [13] E. S. Wijaya and Y. Prayudi, "Integrasi Metode Steganografi DCS Pada Image Dengan Kriptografi Blowfish Sebagai Model Anti Forensik Untuk Keamanan Ganda Konten Digital," in *Seminar Nasional Aplikasi Teknologi Informasi*, 2015, pp. 11–17.
- [14] I. Riadi, A. Fadlil, and T. Sari, "Image Forensic for detecting Splicing Image with Distance Function," *International Journal of Computer Applications (0975 – 8887)*, vol. 169, no. 5, pp. 6–10, 2017.
- [15] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal Processing*, vol. 90, no. 3, pp. 727–752, Mar. 2010.

HASIL CEK_60020397_Point-C71-IRD-850GB-Comparative Analysis of Image Steganography using SLT, DCT and SLT-DCT Algorithm

ORIGINALITY REPORT

5%

SIMILARITY INDEX

4%

INTERNET SOURCES

2%

PUBLICATIONS

0%

STUDENT PAPERS

PRIMARY SOURCES

1

journal.universitasbumigora.ac.id

Internet Source

3%

2

"Security in Computing and Communications",
Springer Science and Business Media LLC,
2020

Publication

1%

3

Donny Kurniawan, Anthony Anggrawan, Hairani
Hairani. "Graduation Prediction System On
Students Using C4.5 Algorithm", MATRIK :
Jurnal Manajemen, Teknik Informatika dan
Rekayasa Komputer, 2020

Publication

1%

Exclude quotes On

Exclude bibliography On

Exclude matches < 1%