Indian Journal of Science and Technology Vol. 3 No. 4 (Apr. 2010) ISSN: 0974- 6846

# A multilayered architecture for hiding executable files in 3D images

P. Mohan Kumar[1] and K. L. Shunmuganathan[2]
[1]CSE Dept., Jeppiaar Engg. College, Chennai-600119; [2]R.M.K. Engineering College, Chennai- 601206, India
mohankumarmohan@gmail.com; kls_nathan@yahoo.com

## Abstract

Steganography is a technique to hide secret messages in a host media called cover media. The advantage of steganography over cryptography is that messages do not attract attention to attackers and even receivers. Steganography and cryptography are often used together to ensure security of the secret messages. This paper introduces steganography work on 3D models. In this paper, we will exploit the geometric characteristics of 3D models to provide high-capacity data hiding. Capacity and invisibility are more important than robustness in the steganography system. Therefore, we aim at maximizing data hiding capacity while limiting distortion of cover models in a lower bounded value. A novel multilayered embedding scheme is proposed for enlarging the hiding capacity. In the extraction procedure, the embedding order can also be obtained using the secret key in the spatial analysis step. The payload can then be correctly extracted in this embedding order.

**Keywords:** Steganography; spatial analysis; polygon models; stego file.

## Introduction

Steganography is the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message, a form of security through obscurity. The word “steganography” is of Greek origin and means "concealed writing" from the Greek words *steganos* (στεγανός) meaning "covered or protected" and *graphein* (γράφειν) meaning "to write". Generally, messages will appear to be something else: images, articles, shopping lists, or some other *covertext* and classically, the hidden message may be in invisible ink between the visible lines of a private letter. The advantage of steganography, over cryptography alone, is that messages do not attract attention to themselves (Katzenbeisser & Petitcolas, 2000). Plainly visible encrypted messages–no matter how unbreakable–will arouse suspicion, and may in themselves be incriminating in countries where encryption is illegal. Therefore, whereas cryptography protects the contents of a message, steganography can be said to protect both messages and communicating parties. Steganography includes the concealment of information within computer files. In digital steganography, electronic communications may include steganographic coding inside of a transport layer, such as a document file, image file, program or protocol. Media files are ideal for steganographic transmission because of their large size (Lin *et al.,* 2005).

There are a number of uses for steganography besides the mere novelty. One of the most widely used applications is for so-called digital watermarking. A watermark, historically, is the replication of an image, logo, or text on paper stock so that the source of the document can be at least partially authenticated (Praun *et al.,* 1999). A digital watermark can accomplish the same function; a graphic artist, for example, might post sample images on her website complete with an embedded signature so that she can later prove her ownership in case others attempt to portray her work as their own. Stego can also be used to allow communication within an underground community.

There are a large number of steganographic methods that most of us are familiar with (especially if you watch a lot of spy movies), ranging from invisible ink and microdots to secreting a hidden message in the second letter of each word of a large body of text and spread spectrum radio communication. With computers and networks, there are many other ways of hiding information, such as:

- Covert channels (e.g., Loki and some distributed denial-of-service tools use the internet control message protocol (ICMP) as the communications channel between the "bad guy" and a compromised system).
- Hidden text within webpages.
- Hiding files in "plain sight".
- Null ciphers (e.g., using the first letter of each word to form a hidden message in an otherwise innocuous text).

The following formula provides a very generic description of the pieces of the steganographic process:

cover_medium+hidden_data + stego_key=stego_medium

In this context, the cover_medium is the file in which we will hide the hidden_data, which may also be encrypted using the stego_key. The resultant file is the stego_medium (which will, of course be the same type of file as the cover_medium). The cover_medium (and thus, the stego_medium) are typically image or audio files. In this article, we will focus on 3D image files and will, therefore, refer to the cover_image and stego_image.

### *Image processing*

Image processing is any form of signal processing for which the input is an image, such as photographs or frames of video; the output of image processing can be either an image or a set of characteristics or parameters related to the image (Fridrich, 1999). Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it. Image processing usually refers to digital image processing, but optical and analog image processing are also possible.

## Information hiding

### *Steganography & digital watermarking*

Steganography applications conceal information in other, seemingly innocent media. Steganographic results may masquerade as other file for data types, be concealed within various media, or even hidden in network traffic or disk space. For many years Information Hiding has captured the imagination of researchers. Digital watermarking and steganography techniques are used to address digital rights management, protect information, and conceal secrets (Cox *et al.,* 2000; Venkat Narayana Rao & Govardhan, 2009). Information hiding techniques provide an interesting challenge for digital forensic investigations. Information can easily traverse through firewalls undetected. Research into steg analysis techniques aids in the discovery of such hidden information as well as leads research toward improved methods for hiding information.

### *Video steganography*

In the year 2007, Irshad developed the concept of Video steganography is a technique to hide any kind of files in any extension into a carrying Video file. This paper is the application developed to embed any kind of data (File) in another file, which is called carrier file. The carrier file must be a video file. It is concerned with embedding information in an innocuous cover media in a secure and robust manner. This system makes the files more secure by using the concepts steganography and cryptography.

### *Audio steganography*

In the year 2006, Shani's project entitled Audio steganography is the application developed to embed an audio file in another audio signal. It is concerned with embedding information in an innocuous cover Speech in a secure and robust manner. This system makes the files more secure by using the concepts steganography and cryptography (Garcia & Dugelay, 2003).

## Overview

Stegano in 3D models consists of two separate procedures: the embedding procedure and the extraction procedure. Both procedures have two main steps: spatial analysis and multilayered embedding/extraction. In the spatial analysis step, we analyze the cover model for obtaining a vertex embedding order and three end vertices for the payload embedding. The vertex traverse approach is adopted to generate the vertex embedding order. The basic idea is to represent a triangle as a pivot edge and two possible exit edges. The traverse order starts with an initial vertex and the next visited vertex is determined by the bit value in a selected secret key to take security into account. Once the initial vertex is determined, the next visited vertex is selected depending on the next bit value in the secret key. If the next bit value is "1," the traversal direction is the right exit edge. If the next bit value is "0," the traversal direction is the left exit edge. Once the vertex embedding order is determined, the next step is to embed the payload on the vertices in that order. Due to encryption, these messages do not attract attention to attackers and even receivers. In the extraction procedure, the embedding order can also be obtained using the secret key in the spatial analysis step (Wu, 2005). The payload can then be correctly extracted in the reverse embedding order. In the past, most of the effort on steganography has been concentrated on various media data types such as an image, an audio file, or even a video file. However, there is relatively little steganography work on 3D models. With the development of 3D hardware, (Ohbuchi, 1998) 3D computing or visualization has become much more efficient than ever. This leads to the widespread use of 3D models in various applications such as digital archives, entertainment, Web3D, MPEG4, and game industry. The main objective of steganographic approach on 3D polygonal meshes is to hide secret messages in cover media through Spatial Analysis, Embedding and Extraction (Chao, 2009).

Our paper introduces steganography work on 3D models. The Input Cover Model which we are going to use in is an OBJ File. From this file we will be able to see the triangle meshes easily. Triangle meshes are a general representation of 3D visual object very well suited to communicate (Benedens, 1999). The 3D object appears as a list of elementary connected objects.

Therefore, 3D models are good candidates and rich resources to serve as the innocuous looking hosts for hiding other types of digital content. There have been many good 3D watermarking methods proposed. Most of them may be easily modified for the purpose of steganography. However, the data-hiding capacity can be very low, since they are not originally designed for this purpose. In this paper, we are going to apply the spatial analysis for vertex ordering. In the spatial analysis step, we analyze the cover model for obtaining a vertex embedding order and the end vertices for the payload embedding. In this model we are going to hide the data in the vertex of the triangle through Embedding Scheme. A novel multilayered embedding scheme is proposed for enlarging the hiding capacity. In the extraction procedure, the embedding order can also be obtained using the secret key in the spatial analysis step. In Extraction the embedded stegano model which contains data is directly processed to spatial analysis after we go to the extraction process to extract the whole data from the embedded cover model. The payload can then be correctly extracted in the embedding order.

## Existing systems

*Lossless data hiding method*

In the year 2008, P. Amat, W. Puech, S. Druon and J.P. Pede boy proposed a new approach of data hiding in 3D objects based on the minimum spanning tree (MST). This method is lossless in the sense that the positions of the vertices are unchanged. The method is blind and does not depend on the ordering of the data in the files. Moreover the embedding capacity is significant and can be as high as 2.5% of the size of the 3D object.

*Data hiding on 3-D triangle meshes*

In the year 2003, Francois Cayre and Benoit Macq*, *Senior Member*, IEEE* presents a new scheme for digital steganography of 3D triangle meshes. This scheme is robust against translation, rotation, and scaling operations. It is based on a substitutive procedure in the spatial domain. The key idea is to consider a triangle as a two-state geometrical object. We discuss its performance in terms of capacity, complexity, visibility, and security. We validate the use of a principal component analysis (PCA) to make our scheme signal-dependent in the line of second generation watermarking scheme. We also define a simple specific metric for distortion evaluation that has been validated by many tests. We conclude by giving some other solutions, including open steganographic schemes that could be derived from the basic ideas presented here.

*Principal component analysis-based mesh decomposition*

Jung-Shiong Chang *et al*. (2001) proposed an automatic mesh decomposition technique based on principal component analysis (PCA) and Boolean operations. First, we calculate the normalized protrusion degree of each dual vertex on a smoothed 3D mesh. The protrusion degree of a vertex and the vertex's 3D coordinates form a 4D feature vector, which we use to represent the polygon mesh. Then, we apply PCA to the set of 4D feature vectors. The projected data along the first principal axis reveals the salient structures of the 3D object. Therefore, by using the first component axis as the search basis, we can identify all the salient parts of an arbitrary 3D object.

## A reversible data hiding approach to mesh authentication

Wu and Cheung (2005), proposed a reversible data hiding method to authenticate 3D meshes by modulating the distances from the mesh faces to the mesh centroid to embed a fragile watermark. It keeps the modulation information in the watermarked mesh so that the reversibility of the embedding process is achieved. Since the embedded watermark is sensitive to geometrical and topological processing, unauthorized modifications on the watermarked mesh can be therefore detected by retrieving and comparing the embedded watermark with the original one. Furthermore, as long as the watermarked mesh is intact, the original mesh can be recovered using some prior knowledge. In the past, most of the effort on steganography has been concentrated on various media data types such as an image, an audio file, or even a video file. There have been many good 3D watermarking methods proposed. Most of them may be easily modified for the purpose of steganography. The existing system is used to hide the information or a secret key inside of a 3D image. In that they are using the spatial analysis to select a location in the image. At the same time it will be very difficult to hide information inside a small polygon. Each polygon contains three coordinates. They are x y and z respectively.

- In existing system, the public key concept is used to hide the information.
- The secret message is hidden in a 3D image.
- In the past, most of the effort on Steganography has been concentrated on various media data types such as an image, an audio file or even a video file.
- In 2004 they hide data's by 2 bits/vertex.
- In 2005 they hide 9 bits/vertex and they done Sliding, extending, and rotating.

In the existing system, data-hiding capacity is very low and less secure. The existing system cannot handles 3D images of small size. When we are hiding information inside an image, the image size will be increased. It will automatically known to the hackers. When we are passing any information the data loss may also occur. To overcome all these problems, the concept of 3D stegano is used.

*Proposed system*

The proposed work introduces steganography work on 3D models with data hiding capacity. We present a very high-capacity and low-distortion 3D steganography scheme (Fig.1). Our steganography approach is based on a novel embedding scheme to hide secret messages in the vertices of 3D polygon models. This novel approach can provide much higher hiding capacity than other state-of-the-art approaches, while obeying the low distortion and security basic requirements for steganography on 3D models. This approach can hide much higher bit rates/vertex than other previous state-of-the-art methods in steganography for 3D polygon models.

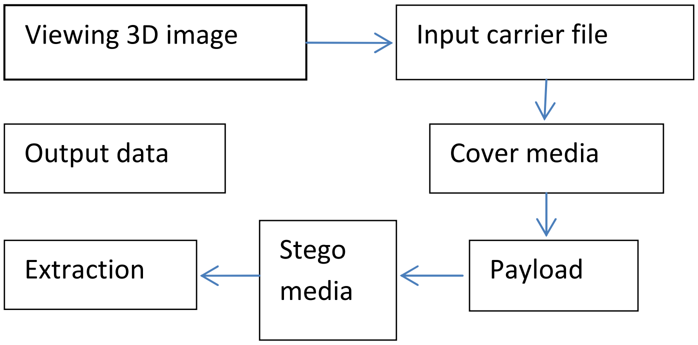
The functional attributes are:

- The novel approach can hide much higher bit rates/vertex (i.e., 35 to 48 bits) than other previous state-of-the-art methods.
- In Steganography for 3D polygon models, they can graphically simplify complicated concepts and convey complex inter-relationships, which are difficult to visualize.
- The 3D image size will be the same after embedding the secret message.
- The proposed system can also use a file (*.EXE) to transfer the information.

- The loss of information is very less in this proposed system.
- The intruders don't know any idea about the message because the image looks same after hiding the data.

The capacity and distortion of our approach depends mainly on the number of vertices. It is because capacity and invisibility are more important than robustness in the steganography system. Here we exploit the geometric characteristics of 3D models to provide high-capacity data hiding. We aim at maximizing data hiding capacity while limiting distortion of cover models in a lower bounded value.

*Fig. 1. Architecture diagram of 3D Stegano system*



*Viewing 3D steganography*

In this Module we are going to view the 3D objects. The 3D file which we are going to use is OBJ file. This file will show the triangle and the vertices. This module is for only viewing the three dimensional images.

*Implementing 3D steganography*

The implementation part contains two main modules, which are used to hide the secret information and also to extract the information. The two main modules in implementing 3D steganography are hiding & extraction.

*Hiding*

In this module we are going to give the carrier File i.e.., OBJ file and also select the output file which is also an OBJ file. After uploading these two files we give the data which we are going to hide in the three dimensional file. The data is encrypted and the number of bytes is determined. If the byte value is within the range then the data is hidden.

*Extraction*

In this module, the system is going to retrieve the hided data from the 3D model. This process will be done by given the saved 3D file in the hiding module. After checking the whether it contains embedded message we will retrieve the content from the given file. Every 3D image contains its own header and footer. Using the key value the system can easily find the ordering of the embedded message.

*Cover model*

The image in which we are going to hide the data is called cover model. Here our cover model is a 3D image. There are many formats for 3D image file. Here our 3D image is of OBJ file format.

*Stego model*

The Image in which data is hidden is called stego model. Stego model is also a 3D image. There are many formats for 3D image file. Here our 3D image is of OBJ file format. The Stego model contains the encrypted hidden data.

*Secret data*

Secret data is the key we are going to use for security. We use the secret data in two place which further adds to the security. It is used initially in spatial analysis and further in embedding and extraction.

*In spatial analysis*

In the spatial analysis secret key is used in the TSPS algorithm. It is used for finding the order of vertex in which we are going to embed the data.

*In multi layered embedding and extraction*

In embedding and extraction the secret key is used to encrypt and decrypt the data. It makes the data illegible to the viewers. Embedding procedure is same as that of extraction. In both the procedures we use the same secret key. Since, multilayered embedding is used the cover model distortion is very less and also the hiding capacity is much more.

*Viewing 3D files*

For viewing 3D image in the 2D plane java 3D is used. Using object loader the 3D file is loaded on to the screen. It also has facilities to retrieve the geometry information from the file.

*Implementing 3D steganography*

The implementation part contains two main modules, which are used to hide the secret information and also to extract the information. Each 3D image contains its own header and footer. If any changes made in the header of footer will cause lose of image or any damage. To protect the actual image we are choosing the vertex of the image to hide the data.

*Viewing 3D image*

In this, we are going to view the 3D objects. The 3D file which we are going to use is OBJ file. This file will show the triangular vertices of the image. This module is for only viewing the 3D images. A 3D model is the mathematical representation of any 3D object (either inanimate or living). A model is not technically a graphic until it is visually displayed. Due to 3D printing, 3D models are not confined to virtual space. A model can be displayed visually as a two-dimensional image through a process called 3D rendering, or used in non-graphical computer simulations and calculations.

## Embedding phase

*Spatial analysis*

Generally spatial analysis or spatial statistics includes any of the formal techniques which study entities using their topological, geometrical or geographic properties. The phase properly refers to a variety of techniques, many still in their early development, using different analytic approaches and applied in fields as diverse as astronomy, with its studies of the placement of galaxies in the cosmos, to chip fabrication engineering, with its use of 'place and route' algorithms to build complex wiring structures. The most fundamental of these is the problem of defining the spatial location of the 3D image being loaded. In this module, we use TSPS algorithm to obtain the vertex embedding order. Using TSPS algorithm the next visited vertex is selected depending on the bit value of the secret key. If the bit value is “1,” the traversal direction is the right exit edge. If the next bit value is “0,” the traversal direction is the left exit edge. This process increases the security of the hidden data.

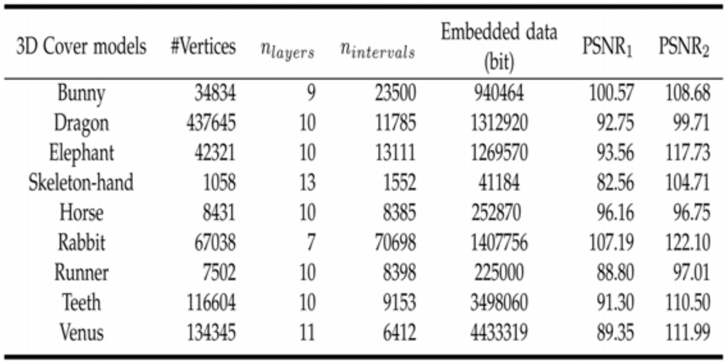
*Encryption*

Encryption is the process of transforming information (referred to as plaintext) using an algorithm (called cipher) to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key. The result of the process is encrypted information (in cryptography, referred to as cipher text). The message to be embedded is encrypted using the secret key. The encryption algorithm used here is DES. The encrypted message is then send to the embedding process.

*DES*

The data encryption standard (DES) is a block cipher (a form of shared secret information) that was selected by the National bureau of standards as an official federal information processing standard (FIPS) in 1976 and which has subsequently enjoyed widespread use internationally. It is based on a symmetric key algorithm that uses a 56-bit key. The algorithm was initially controversial with classified design elements, a relatively short key length, and suspicions about a national security agency (NSA) backdoor. DES consequently came under intense academic scrutiny which motivated the modern understanding of block ciphers and their cryptanalysis. DES is now considered to be insecure for many applications. This is chiefly due to the 56-bit key size being too small; in January, 1999, distributed.net and the electronic frontier foundation collaborated to publicly break a DES key in 22 h and 15 min. There are also some analytical results which demonstrate theoretical weaknesses in the cipher, although they are unfeasible to mount in practice. The algorithm is believed to be practically secure in the form of triple DES, although there are theoretical attacks. In recent years, the cipher has been superseded by the advanced encryption standard (AES). Furthermore, DES has been withdrawn as a standard by the national institute of standards and technology.

*Table 1. Embedding results of various 3D*



| 3D Cover models | #Vertices | $n_{layers}$ | $n_{intervals}$ | Embedded data (bit) | PSNR$_1$ | PSNR$_2$ |
|---|---|---|---|---|---|---|
| Bunny | 34834 | 9 | 23500 | 940464 | 100.57 | 108.68 |
| Dragon | 437645 | 10 | 11785 | 1312920 | 92.75 | 99.71 |
| Elephant | 42321 | 10 | 13111 | 1269570 | 93.56 | 117.73 |
| Skeleton-hand | 1058 | 13 | 1552 | 41184 | 82.56 | 104.71 |
| Horse | 8431 | 10 | 8385 | 252870 | 96.16 | 96.75 |
| Rabbit | 67038 | 7 | 70698 | 1407756 | 107.19 | 122.10 |
| Runner | 7502 | 10 | 8398 | 225000 | 88.80 | 97.01 |
| Teeth | 116604 | 10 | 9153 | 3498060 | 91.30 | 110.50 |
| Venus | 134345 | 11 | 6412 | 4433319 | 89.35 | 111.99 |

*Vertex embedding*

Here we are using embedding of encrypted message in layers of the cover model. Each vertex is embedded with at most 48 bits without cover model distortion. At most embedding is done in two layers, thus increasing the capacity of embedding.

## Extraction phase

*Spatial analysis*

In extraction, the same process as that of embedding is used for spatial analysis. The vertex ordering for retrieval of the message is determined using the TSPS algorithm.

*Extraction*

Data extraction is the act or process of retrieving (binary) data out of (usually unstructured or badly structured) data sources for further data processing or data storage (data migration). The import into the intermediate extracting system is thus usually followed by data transformation and possibly the addition of metadata prior to export to another stage in the data work flow. The message is retrieved in the order of the vertices found by the spatial analysis process. Extraction is repeated for both the layer. After extraction the message retrieved is sent to the decryption process.

*Decryption*

Here, the retrieved message is decrypted using the secret key. The decryption algorithm used is DES. The output of the decryption is the message which is hidden in the cover model. Thus the hiding capacity is increased due to embedding in layers. And also the security is enhanced since the secret key is used in both spatial analysis and embeddind/extraction.

## Experimental results

The novel embedding approach hides information using 3D polygon models. We optimized the number of vertices to obtain a good balance between distortion and capacity. Since the conformal approach has been tested numerically so far mostly in 2D cases with unphysical global structure, it is necessary to implement full 3D codes to run on general enough data. The experimental results showed that the proposed method can provide much higher capacity than previous approaches (Table 1, Fig.2). Currently, the proposed approach has the following limitations that will be solved in the near future. Perfectly smooth (i.e., sphere) or extremely small-size models are not suitable for selection as cover models because the hidden data might be easily observed after even a very small modification by any embedding method. As in the previous work utilizing TSPS to determine the vertex traverse list may potentially hinder the robustness of the proposed method. Although the end vertices and initial vertices obtained from cover and stego models are identical in all cases in our experiments, we cannot guarantee that both are always exactly identical. However, this approach cannot withstand similarity transformations. And we can never recapture the origins of geometry. A better approach for determining vertex traverse list is required in the near future. Another limitation is that our approach cannot withstand certain malicious attacks such as smoothing, additional noise, no uniform scaling, simplification, and vertices resembling. As a result, the proposed approach is not suitable for the applications of digital content protection and authentication.

## Conclusion and future enhancements

In future, the system will be developed to hide more amount of information. The noise of the image will be reduced. The proposed system now deals with three dimensional images only. In future, it will be enhanced to hide the secret messages in any type of media as well as this system will help in the defense also. The system deals with spatial analysis only. In future, the system will be increased to any type of algorithm to secure the information. Due to time and computing limitations, we could not explore all facets of steganography and detection techniques. As you saw, we studied the power in our pictures to test for hidden data. Another method which we were unable to explore was to analyze the noise of the pictures. Adding hidden data adds random noise, so it follows that a properly tuned noise detection algorithm could recognize whether or not a picture had steganographic data or not.

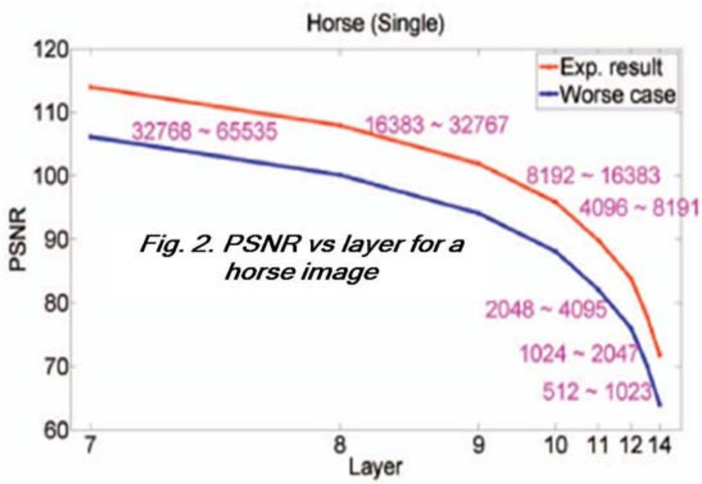


*Fig. 2. PSNR vs layer for a horse image*

## Acknowledgement

We take immense pleasure in thanking our chairman Dr. Jeppiaar M.A, B.L, Ph.D, the Directors of Jeppiaar Engineering College Mr. Marie Wilson, B.Tech, MBA, (Ph.D), Mrs. Regeena Wilson, B.Tech, MBA, (Ph.D) and the principal Dr. Sushil Lal Das M.Sc (Engg.), Ph.D for their continual support and guidance. We would like to extend our thanks to my guide, our friends and family members without whose inspiration and support our efforts would not have come to true. Above all, we would like to thank God for making all our efforts success.

## References

1. Benedens O (1999) Geometry-based watermarking of 3D polygonal models. *Proc. IEEE Computer Graphics & Applications. 19 (1),* 45-46.
2. Cayre F, Devillers O, Schmitt F and Maıtre H (2004) Watermarking 3D triangle meshes for authentication and integrity. *INRIA Res.Report RR-5223.*
3. Chao MW, Lin CH, Yu CW and Lee TY (2009) A high capacity 3D steganography algorithm. *IEEE Trans. on visualization and computer graphics. 15 (2), 274-284.*
4. Cox IJ, Miller ML and Bloom JA (2000) Digital watermarking*.*
5. Fridrich J (1999) Applications of data hiding in digital images. *Tutorial for the ISSPA,* 22-25.
6. Garcia F and Dugelay J (2003) Texture-based watermarking of 3D video objects. *IEEE Trans. Circuits & Systems for Video Technol. 13 (8),* 853-866.
7. Yin K, Pan Z, Jiaoying S and Zhang D (2001) Robust mesh watermarking based on multiresolution processing. Computers & Graphics. 25, 409-420.
8. Katzenbeisser S and Petitcolas FAP (2000) eds., Information hiding techniques for steganography and digital watermarking. *Artech House. pp:156-172.*
9. Lin HY, Liao HYM, Lu CS and Lin JC (2005) Fragile watermarking for authenticating 3D polygonal meshes. *IEEE Trans. Multimedia. 7 (6),* 997-1006.
10. Ohbuchi R, Masuda H and Aono M (1998) Watermarking three-dimensional polygonal models through geometric and topological modifications. *IEEE J. Selected Areas in Comm. 1,* 551-560.
11. Praun E (1999) Robust mesh watermarking. *Proc. ACM SIGGRAPH '99. IJTIC.* pp:49-56,
12. Venkat Narayana Rao T and Govardhan A (2009) Reversible watermarking mechanisms - a new paradigm in mage security. *Indian J.Sci.Technol*. 2 (5), 23-28. Domain site: http://www.indjst.org.
13. Wu HT and Cheung YM (2005) A reversible data hiding approach to mesh authentication. *Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence (WI '05), 1,* 774-777.