CORESA 2007, Montpellier

3-D Digital Watermarking

Adrian G. Bors

University of York
Computer Science Department
YORK Y010 5DD, UK

adrian.bors@cs.york.ac.uk

9 November 2007

Outline

- Overview of 3D watermarking
- Spatial methods
- Transform domain methods
- Results
- Conclusions

Watermarking properties

Protection Technique	***Purpose***	***Alteration of the image***	***Size of the key***	***Robustness to alterations***
Cryptography	To encode data	Entirely changed	A changeable codebook- large	Robust
Watermarking	Image copyright protection	No alteration	small	Very robust
Authentication (Integrity)	Certify the authenticity of an image	No alteration	small	It must vanish when image content changes
Steganography (Information hiding)	Hide information into the image	No alteration	Very large	Robust

Techniques for data protection

3-D Watermarking literature

- Watermarking images – research started in early 90’s.

- Watermarking graphics – research started in 1997 but initially with fewer results.

- Actually the idea has been around for longer …

  Herodotus relates that in 480 B.C. a secret message was send by means of tattooing it on a shaved sclave head.

  Towards the end of cold war steganography became interesting as a mean to hide secret information due to the deficiencies of cryptography.

3-D Watermarking literature

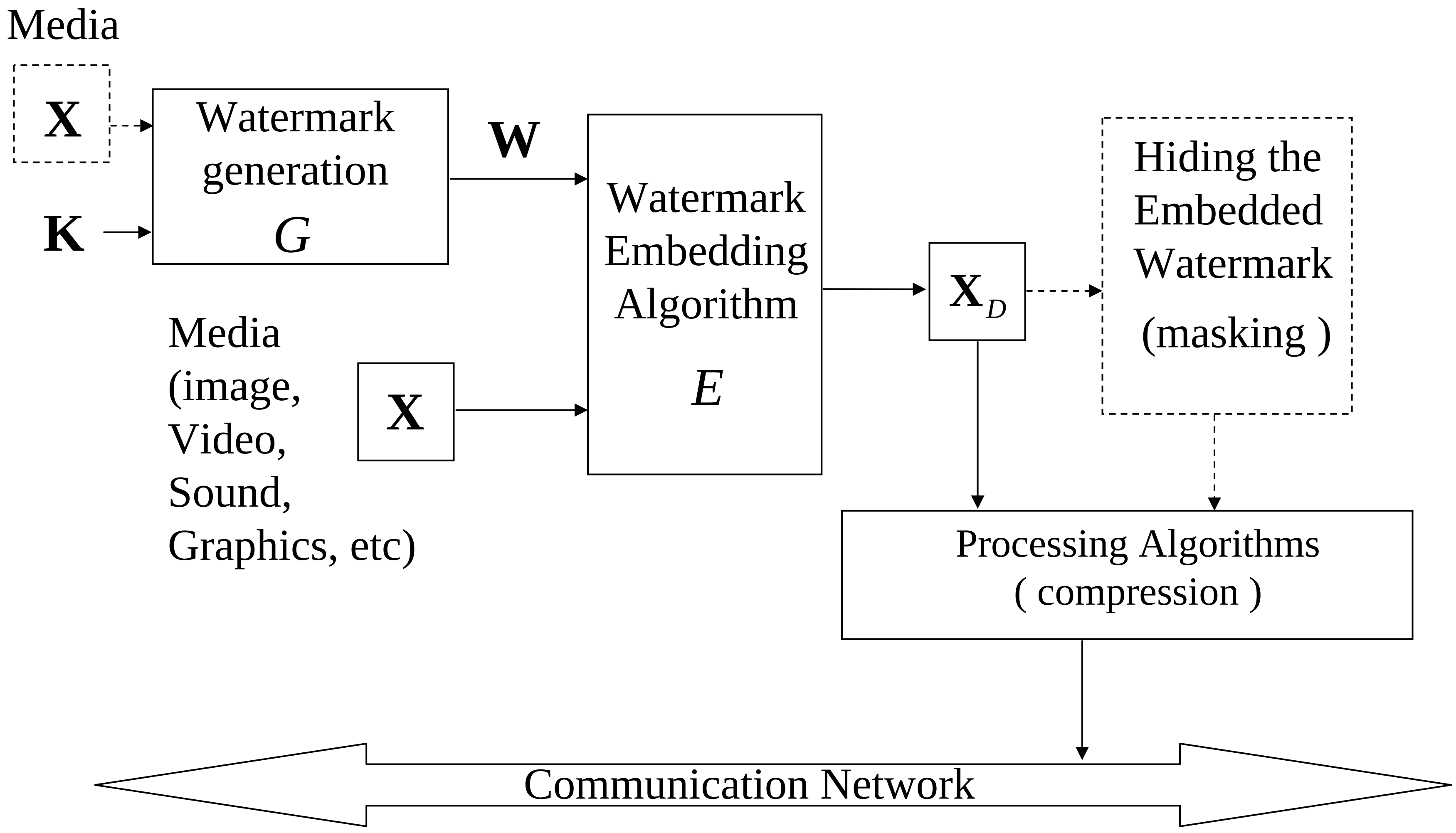
- Watermarking algorithm is characterized by two stages:

  Stage I – Watermark embedding – a signal is generated based on a key and is inserted in the media object such that it cannot be identified (either visually or by electronic means).

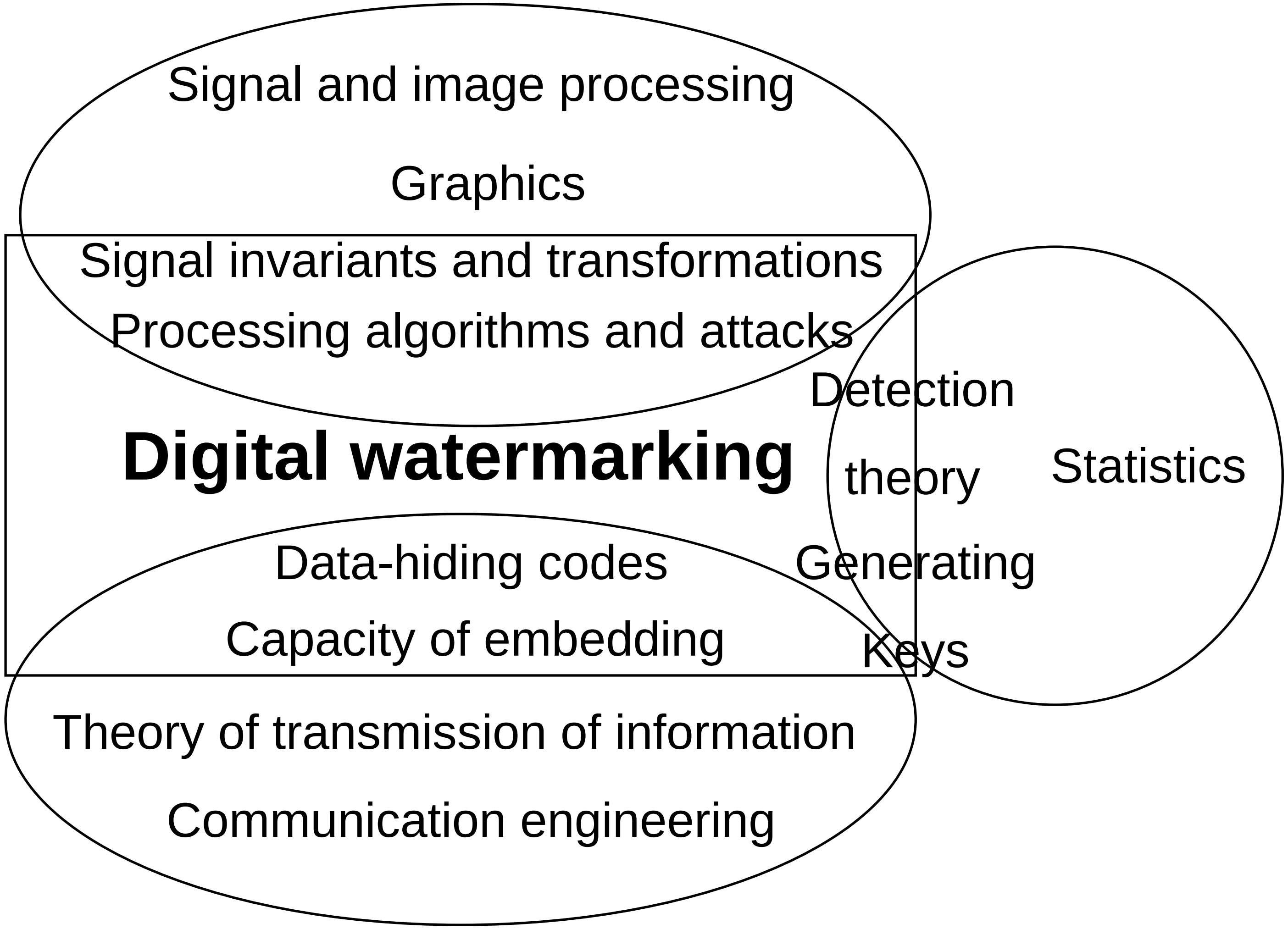
The media object is used in various applications or processed with the intention of removing the signal characterizing the watermark. (data compression, filtering, partial data removal, smoothing, etc.)

  Stage II – Watermark detection – the watermarked media object is received and the signal characterizing the key is extracted or detected in it (for example using correlation).

Watermark Embedding



Connection with other areas



Signal and image processing

Graphics

Signal invariants and transformations

Processing algorithms and attacks

Digital watermarking

Data-hiding codes

Capacity of embedding

Theory of transmission of information

Communication engineering

Detection theory

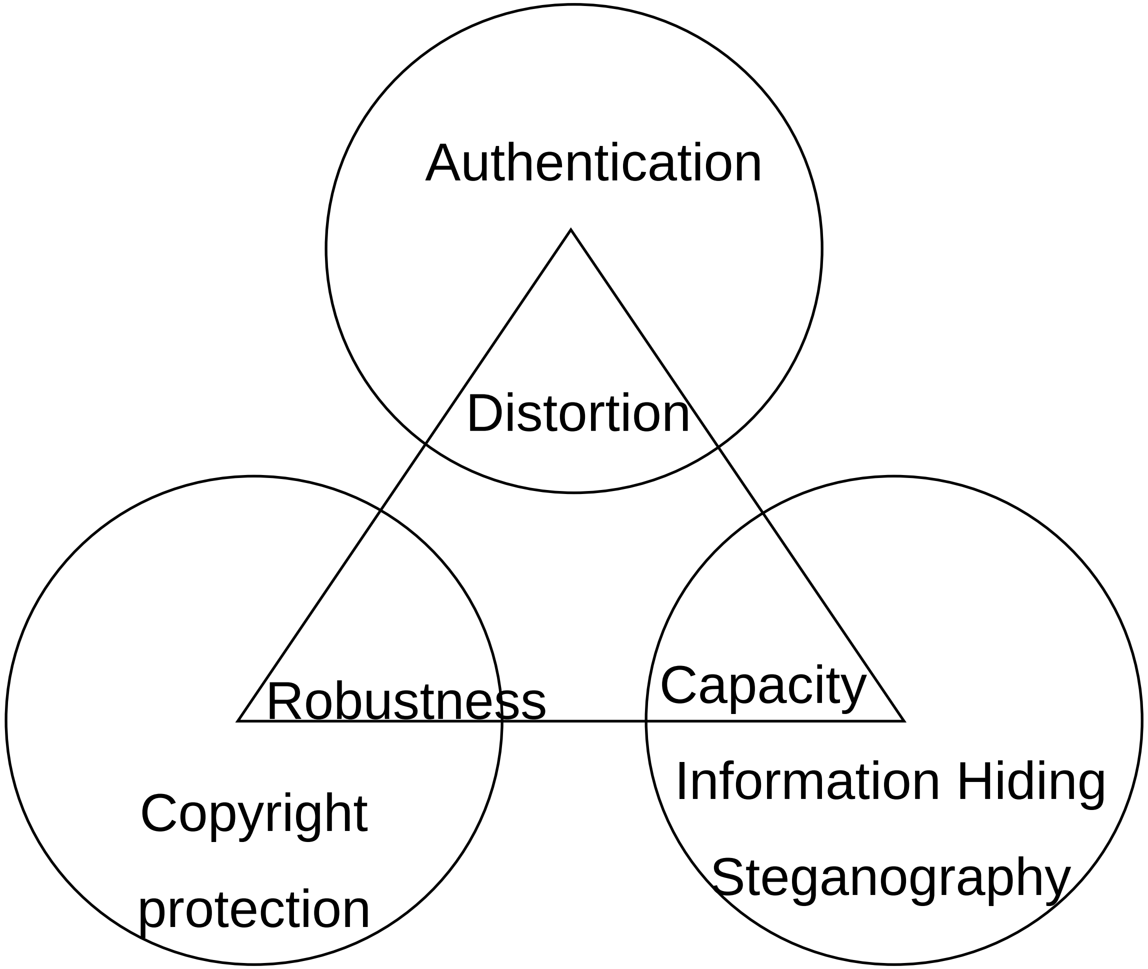
Generating Keys

Statistics

Watermarking trade-offs

Trying to improve any of these characteristics

leads to limiting the other two



3-D Watermarking Review

- 3-D Watermarking approaches have been attempted on various multimedia data:
  - Watermarking 3-D objects from video streams – in MPEG-4 parametric space (Hartung, Eisert and Girod, 1998) , (Yang, Liao and Hsieh, 2002)
  - Watermarking texture of 3-D video objects (Garcia and Dugelay, 2003)
  - Watermarking using phased shift interferometry for watermarking holograms (Kishk and Javidi, 2003)

- A 3-D graphical object can be represented in various ways
  - Voxel models
  - Constructive solid geometry (CSG)
  - Parametric models (splines, NURBS)
  - Polygonal meshes

  Authetication of CSG models (Fornaro and Sanna, 2000)

  Watermarking NURBS (Benedens, 2000), (Ohbuchi, Masuda and Aono, 1999)

3-D Watermarking Review

- Attributes such as colour, texture or shading can be easily removed in graphical objects.

- Unlike images which are represented as regular 2-D lattices of data which can be represented as matrices.

- Let us consider $\{\mathbf{V_1, V_2, \ldots, V_N}\}$ a set of vertices which are joined by edges and polygons

- In this study we consider only manifolds, each edge is contained in only two polygons. The mesh describes a surface in 3D.

3-D Watermarking Review

Classification of 3-D mesh watermarking approaches

- **Non-blind methods**
  - Which require the original object in the detection stage for comparing it with the watermarking object for extracting the watermark

  (Kanai, Date, Kishinami, 1998), (Ohbuchi, Masuda and Aono, 1999) ,

  (Ohbuchi, Mukaiyama and Takahashi, 2002), (Yin et. al, 2001),

  (Yu, Ip, Kwok, 2003)

- **Semi-blind methods**
  - Require in the detection stage: complex registration or alignment in order to extract the watermark

  (Praun, Hoppe, Finkelstein, 1999),(Benedens, Busch, 2000)

  (Zafeiriou, Tefas, Pitas, 2005)

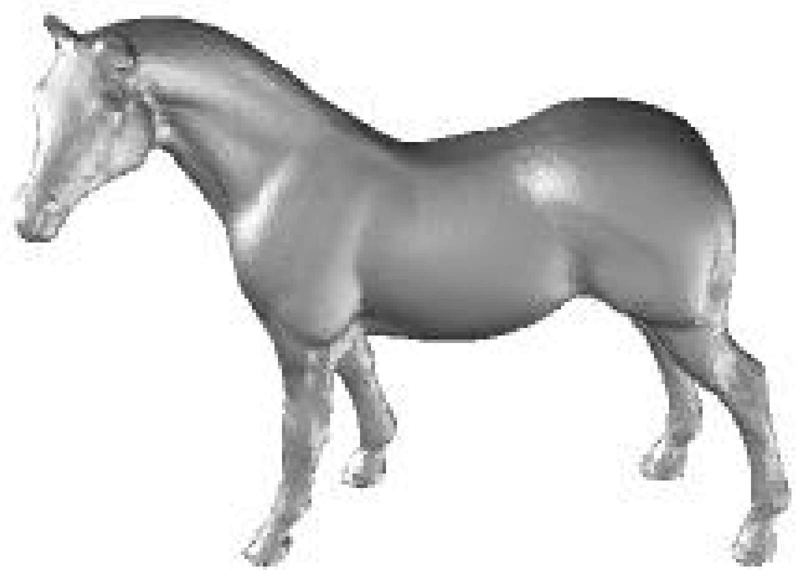
  - Require additional parameters or information

  (Benedens, 2000)

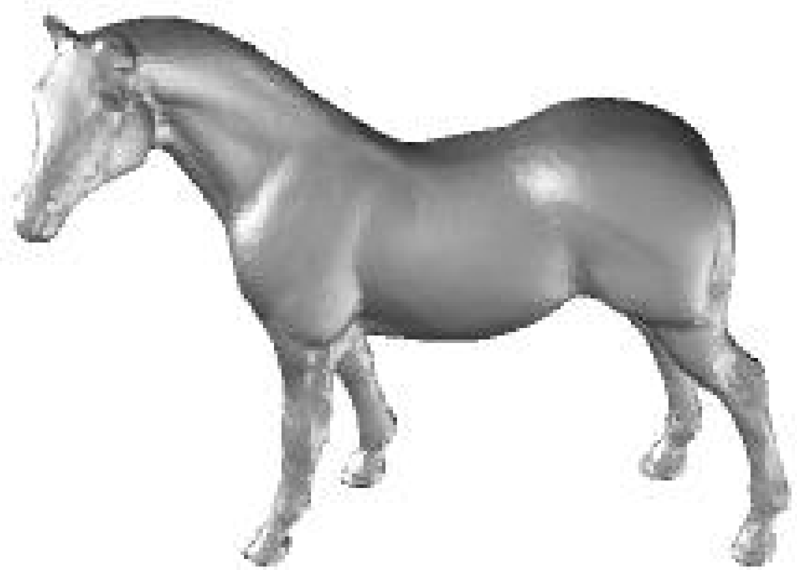
- **Blind methods** – most practical but generally less robust

3-D Watermarking Review

Exemple of non-blind watermarking



Original



Watermarked

(Ohbuchi, Mukaiyama and Takahashi, 2002)

Method applied in the spectral domain

– Such methods tend to be more robust but not very practical because they require the original object in the detection strage

3-D Watermarking Review

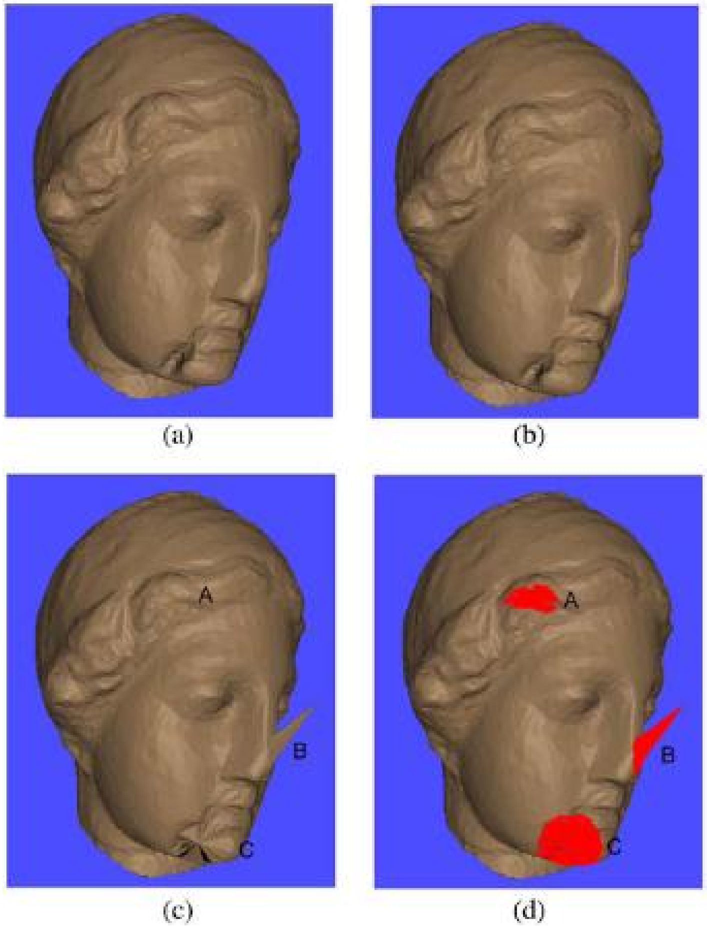
- Authentication
  - Protection of the cover media
  - Robust to certain algorithms (compression)
  - Non-robust to distortions and deformations (designed to fail)
  - Locate the affected area
  - Identify the endured attack
  - Also called fragile watermarking

  (Ohbuchi, Masuda and Aono, 1998)
  (Yeo, Yeung, 1999)
  (Fornaro and Sanna, 2000) , (Lin et. al., 2005) ,
  (Chou, Tseng, 2006)

- Methods that maximize the capacity of embedding

  (Cheng, Wang, 2006), (Tsai et. al., 2006)

Original Watermarked



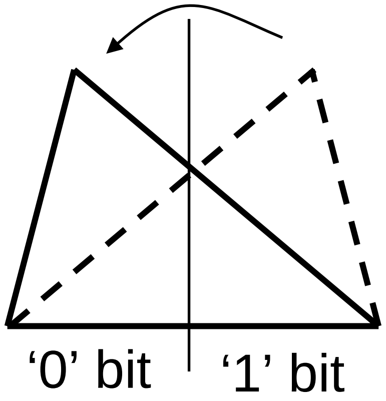
Changes Detection of change

3-D Watermarking in spatial domain

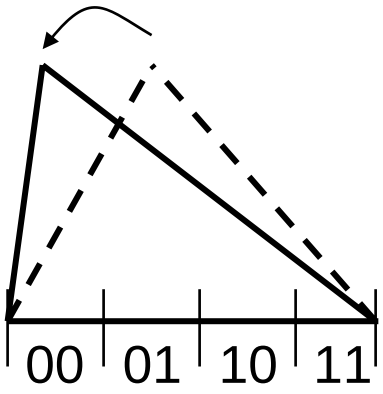
- In the spatial domain (geometric)
- Localised embedding

  Using ratios of 2-D and 3-D geometrical measures

  (Ohbuchi, Masuda and Aono, 1998), (Benedens, 1999), (Wagner, 2000), (Cayre, Macq, 2003)



Embed one bit



Embed two bits

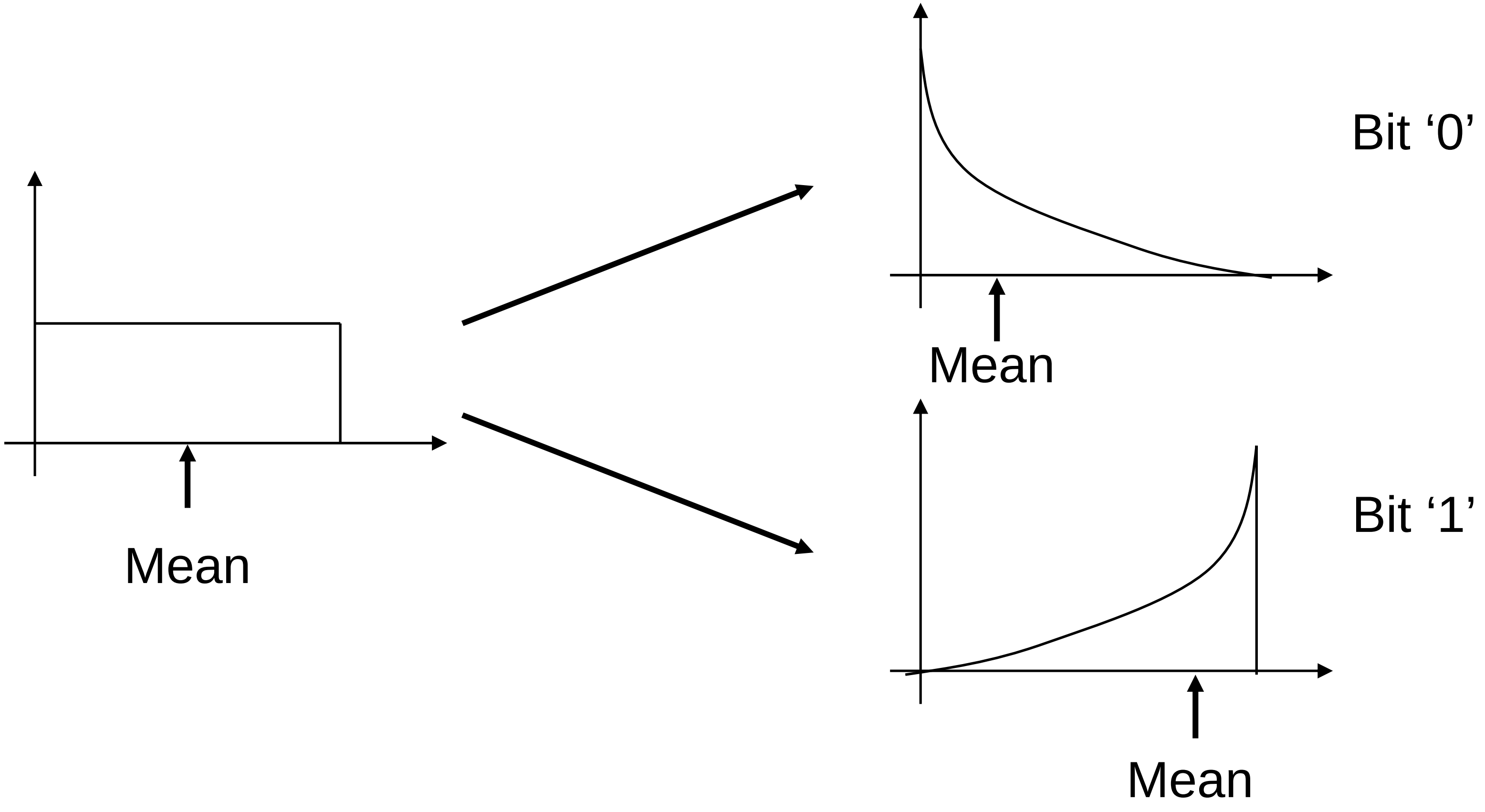
  - Using surface normals (Benedens, 1999), (Lee, Kwon, 2007)
  - Constraints mapping in 3-D (Bors, 2002, 2006)
  - Insert watermarks into 2-D contours of 3-D mesh objects (Bennour,Dugelay,2006)

3-D Watermarking in Spatial domain

- Global embedding
  - Altering the symmetry of the graphical object

(Zafeiriou, Tefas, Pitas, 2005), (Cho, Prost, Jung, 2007)

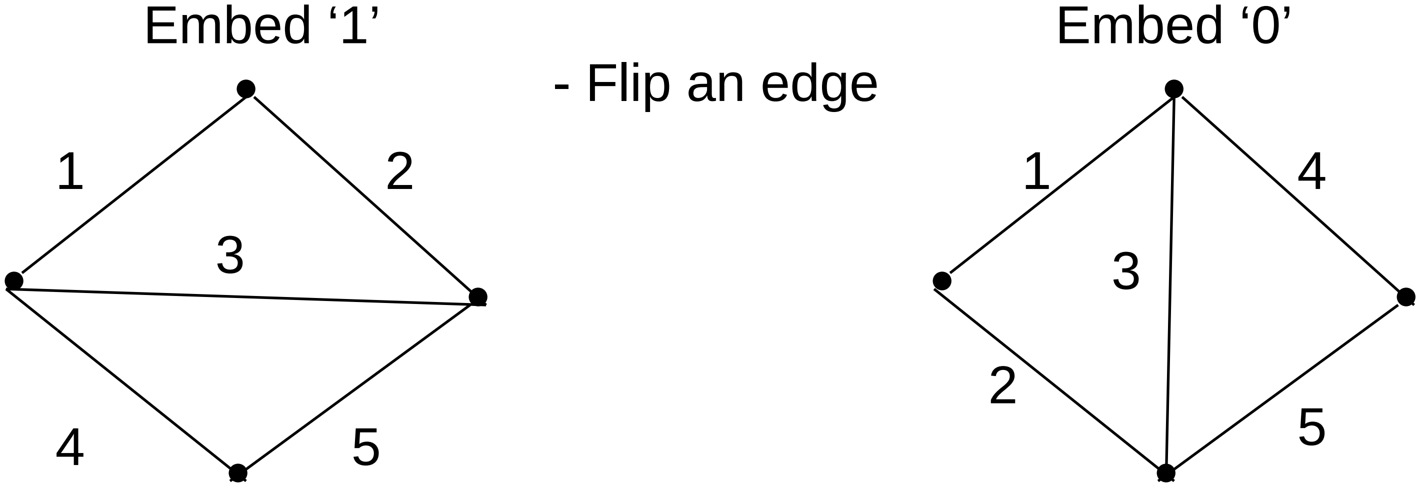
- use statistical detection on a measure – usually the result of correlation



3-D Watermarking in Spatial domain

- Modifying the connectivity – mostly fragile

(Ohbuchi, Masuda, Aono, 1997) , (Amat, Puech, 2007)



Embed ‘1’

- Flip an edge

Embed ‘0’

- Using the order of edges

- Using parametrization

(Li et. al., *Comp. and Graphics*, 2004), (Song, Cho, 2004)

3-D Watermarking in the Transform domain

- Multiresolution filters
  - Basis Functions (Praun, Hoppe, Finkelstein,1999), (Wu, Kobbelt, 2005)
  - Pyramid based (Yin et. al, *Computers & Graphics*, 2003)
  - 3-D Wavelets (Kanai, Date, Kishinami, 1998), (Yang, Liao and Hsieh, 2002), (Uccheddu, Corsini, Barni, 2004)

Wavelet decomposition – at each resolution *j* there are:

- two analysis filters A_j and B_j

- two synthesis filters - the scaling function P_j and the wavelet function Q_j

that decompose the 3-D shape as follows

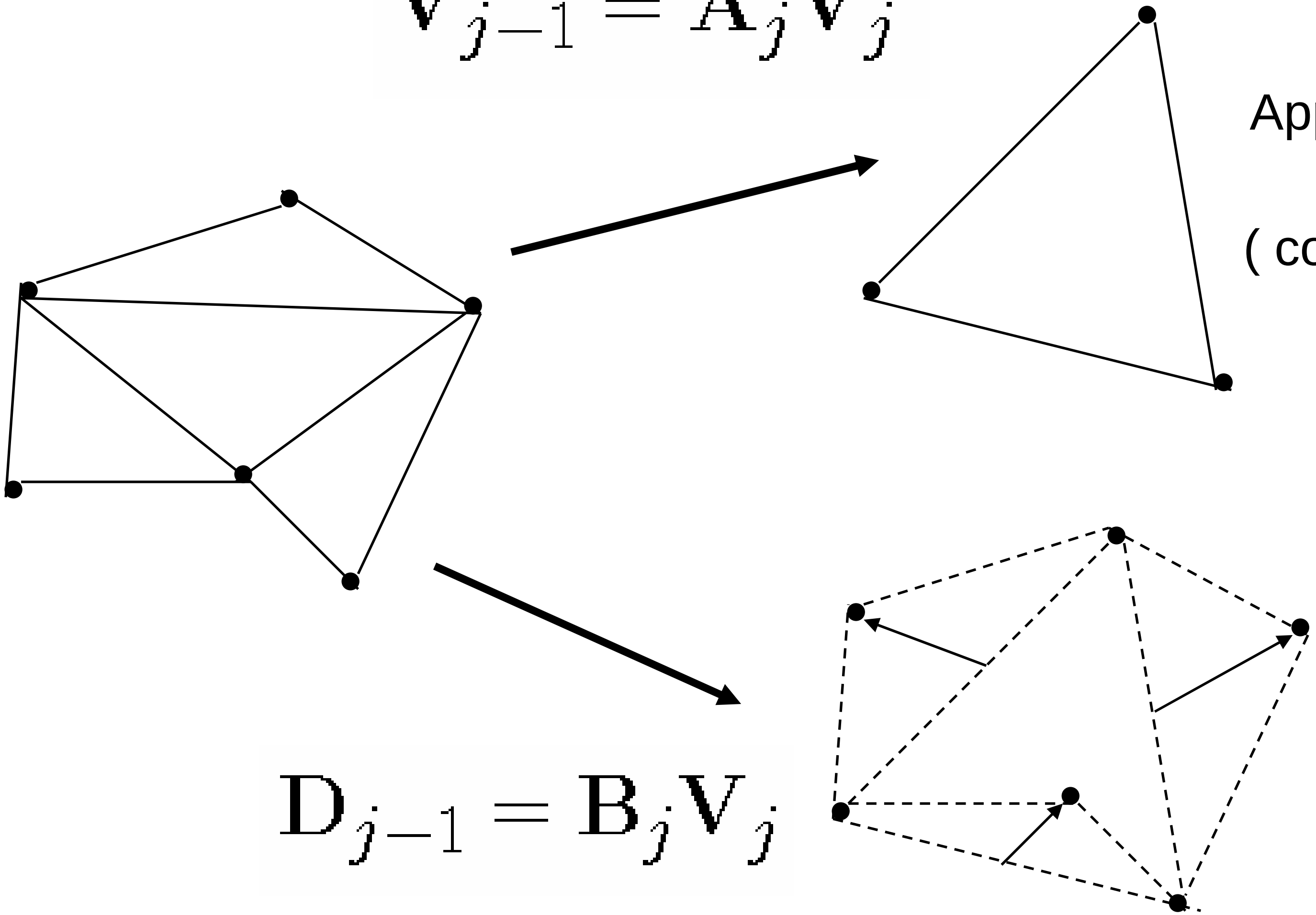
$$V_j = P_j V_{j-1} + Q_j D_{j-1}$$

While D_{j-1} represents the wavelet coefficients and the filters are related as:

$$\begin{bmatrix} \mathbf{A}_j \\ \text{----} \\ \mathbf{B}_j \end{bmatrix} = [\mathbf{P}_j | \mathbf{Q}_j]^{-1}$$

3-D Watermarking in the Wavelet domain

$$\mathbf{V}_{j-1} = \mathbf{A}_j \mathbf{V}_j$$



Approximation

(coarse model)

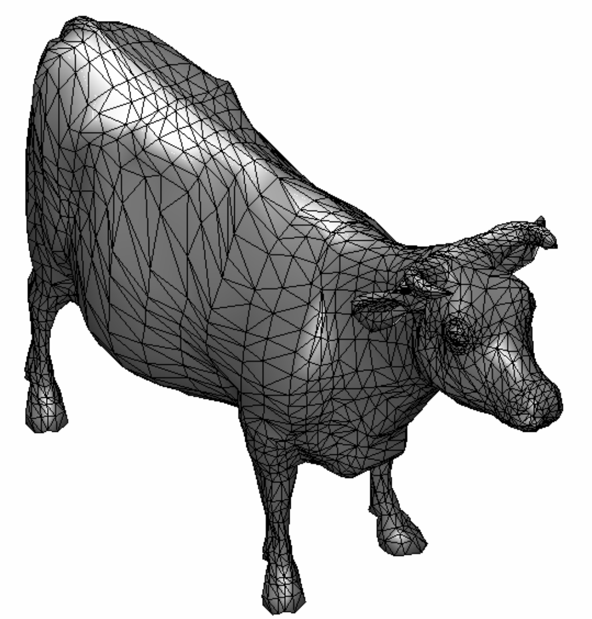
$$\mathbf{D}_{j-1} = \mathbf{B}_j \mathbf{V}_j$$

Wavelet

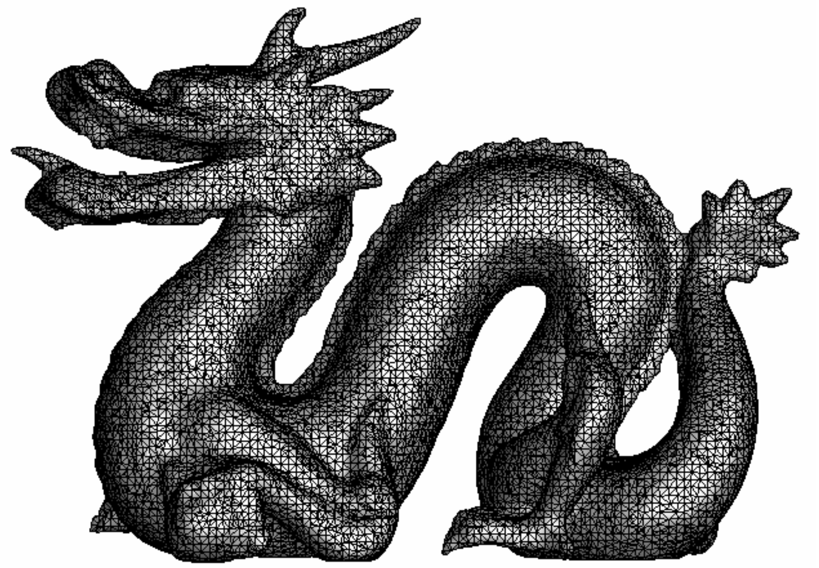
coefficients

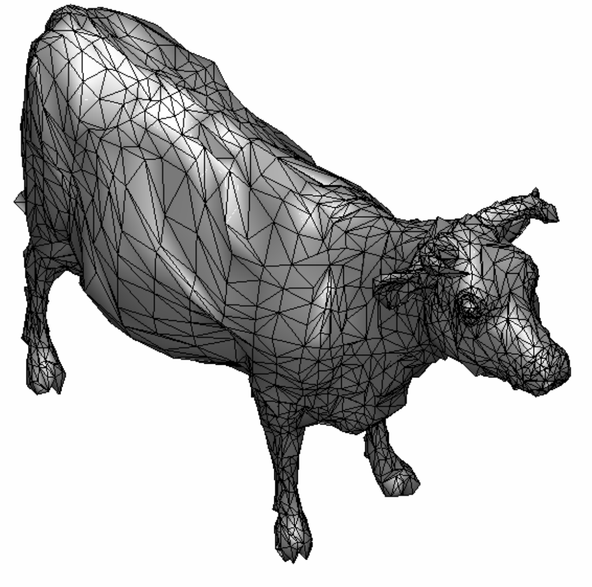
(details of the model)

3-D Watermarking in the Wavelet domain

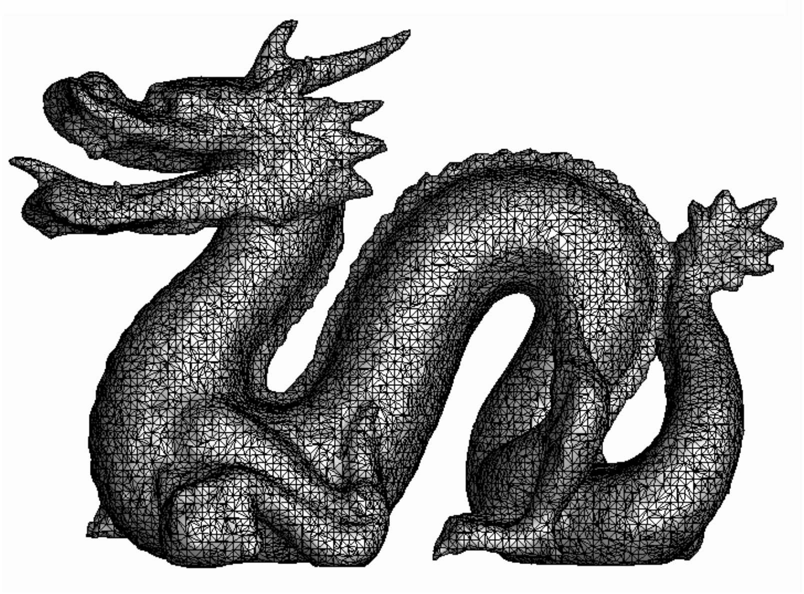


Original





Watermarked



3-D Watermarking in the Spectral domain

- Spectral coefficients - global embedding

  (Ohbuchi et. al., 2001), (Ohbuchi, Mukaiyama and Takahashi, 2002)

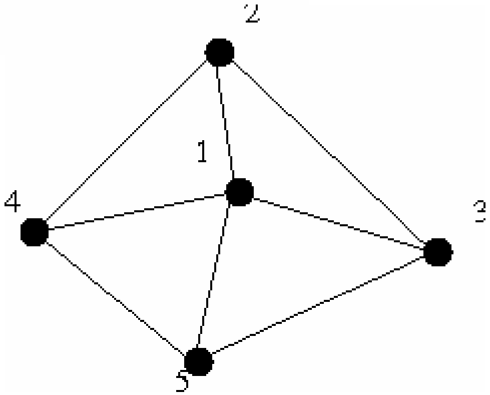
  (Cayre et. al., 2003)

  - Most of these are non-blind but they can be made blind.

  -The Laplacian Matrix is formed for the graphical object and eigendecomposed

$$L_{i,j} = \begin{cases} d(\mathbf{V}_i) & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } \mathbf{V}_i \text{ adjacent } \mathbf{V}_j \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{L} = \mathbf{E}^T \Omega \mathbf{E}$$



$$\longrightarrow \quad L = \begin{bmatrix} 4 & -1 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 & 0 \\ -1 & -1 & 3 & 0 & -1 \\ -1 & -1 & 0 & 3 & -1 \\ -1 & 0 & -1 & -1 & 3 \end{bmatrix}$$

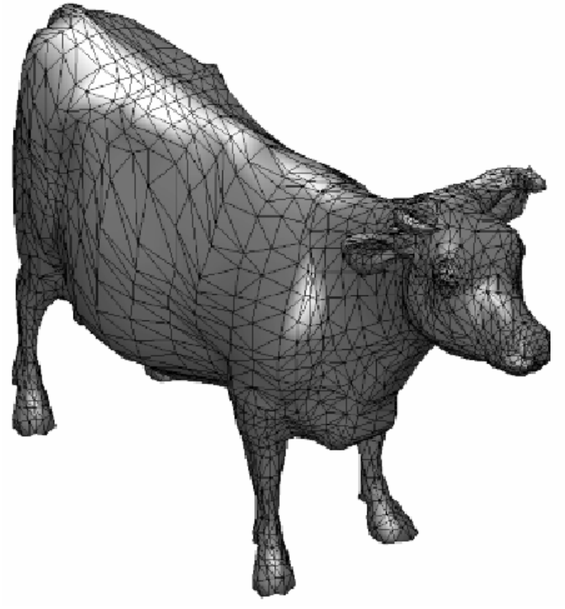
Watermarking is performed by changing the spectral coefficients

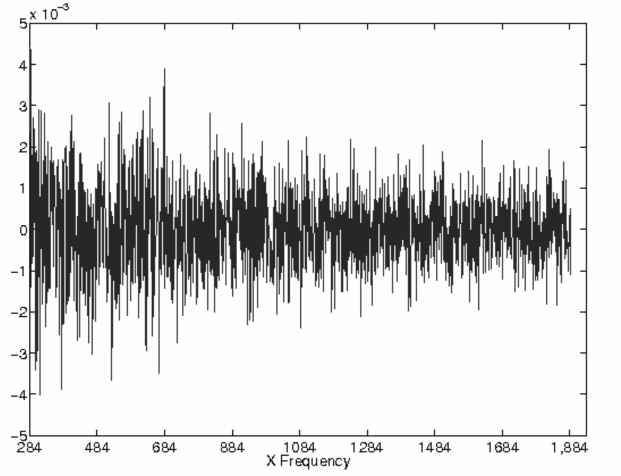
$$\mathbf{C} = \mathbf{E}\mathbf{V} \quad \longrightarrow \quad \mathbf{V} = \mathbf{E}^T \hat{\mathbf{C}}$$

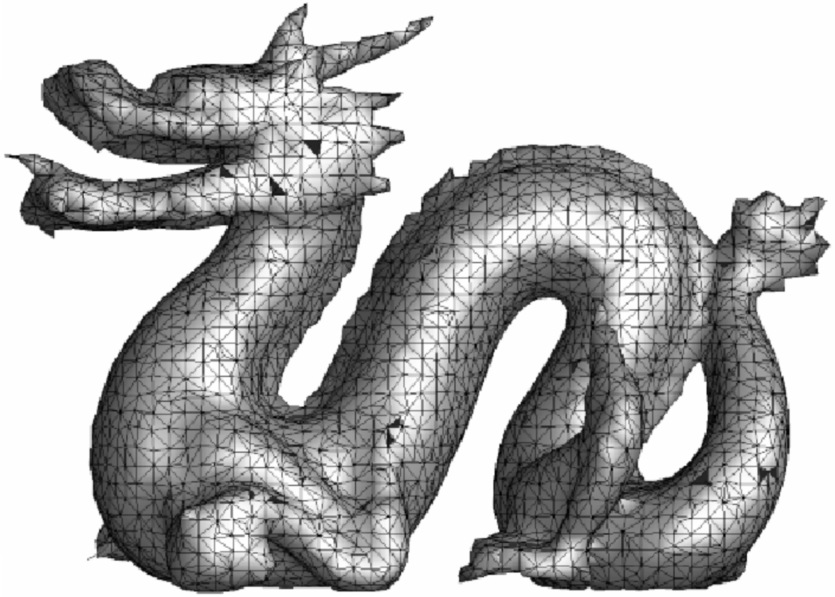
Computational intensive for real 3-D graphical objects

3-D Watermarking in the Spectral domain

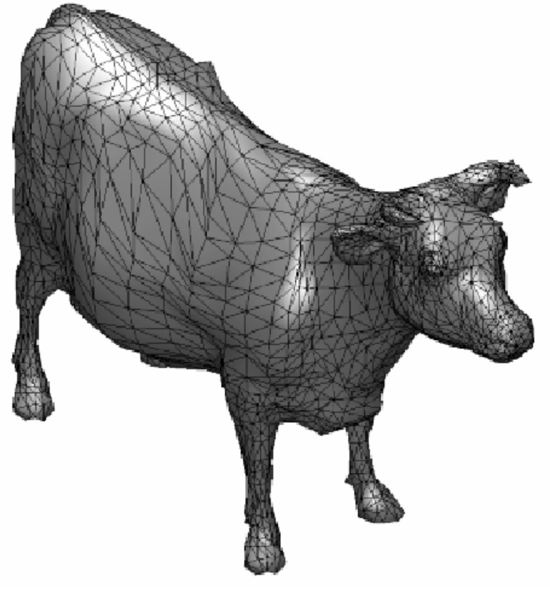
Original

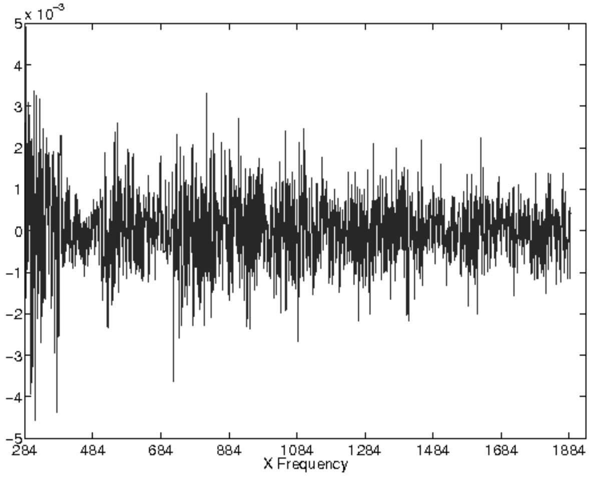


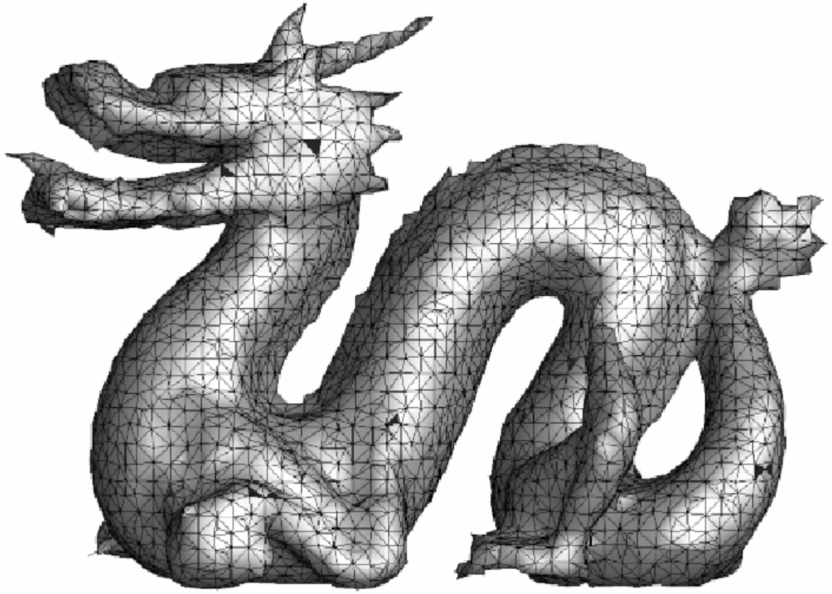




Spectral coefficients







Watermarked

3D watermarking using local moments

- 3D watermarking method presented in:
- A.G. Bors, “Watermarking mesh based representations of 3-D objects using local moments ”, *IEEE Trans. on Image Processing*, vol. 13, no. 3, pp. 687-701, March 2006.
  - Method in spatial domain
  - Blind method – does not need the original shape
  - Localised embedding
- Main steps:

- Embedding

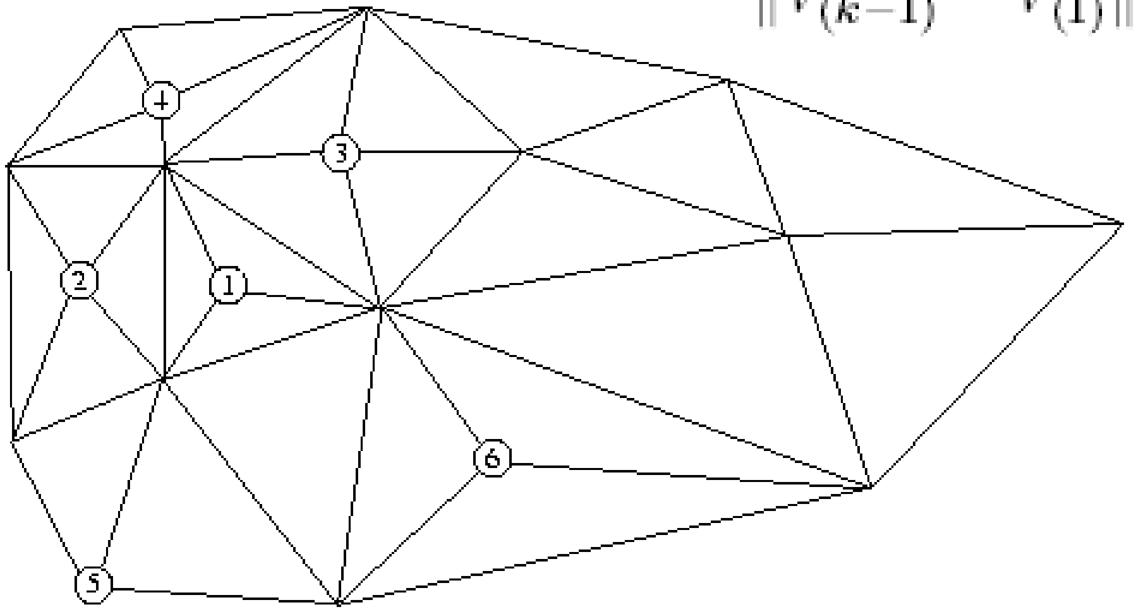
  - Ordering selected neighbourhoods for watermarking
  - Embedding changes in local neighbourhoods
  - Verifying changes

- Detection – each bit detected separately

Vertex Ordering

- The first vertex fulfils: $V_{(1)} = \min_{V_i \in \mathcal{B}} D(\hat{V}_i)$
- The vertices are ordered according to their increasing distance to the first chosen vertex

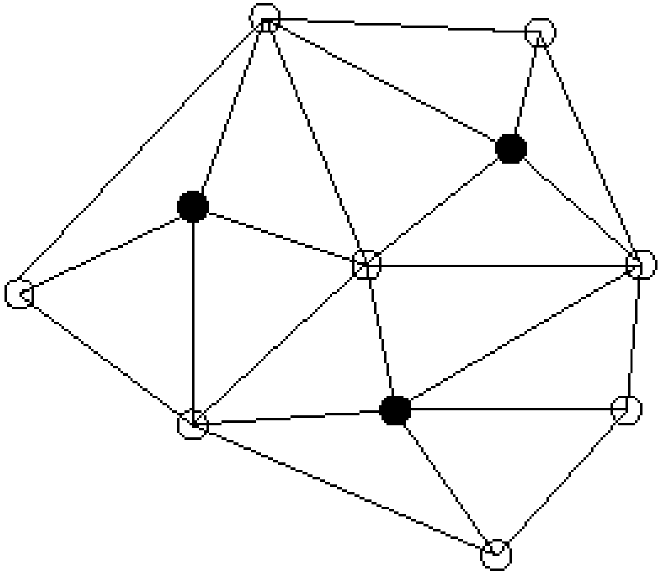
$$\|V_{(k-1)} - V_{(1)}\|^2 < \|V_{(k)} - V_{(1)}\|^2$$



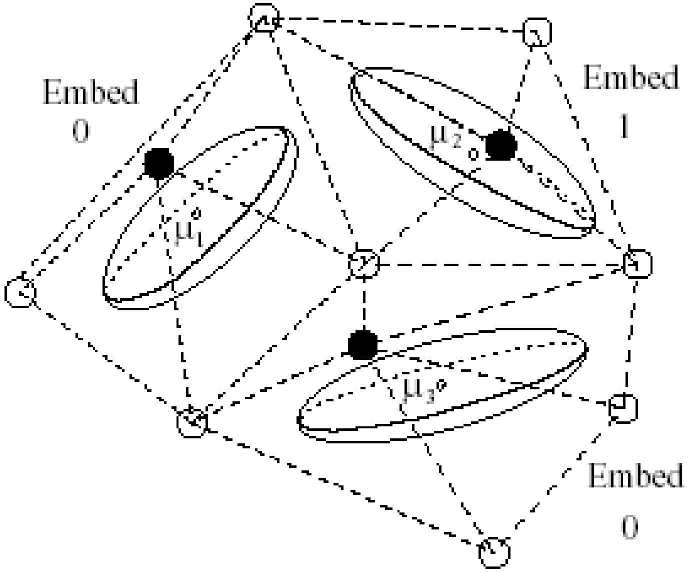
Example of vertex ordering

Embedding and retrieving the watermark

Exemplification of embedding information using the bounding ellipsoids



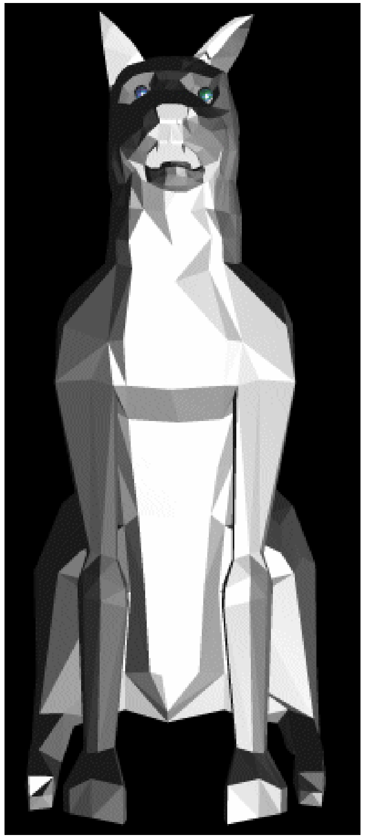
Original Mesh Structure



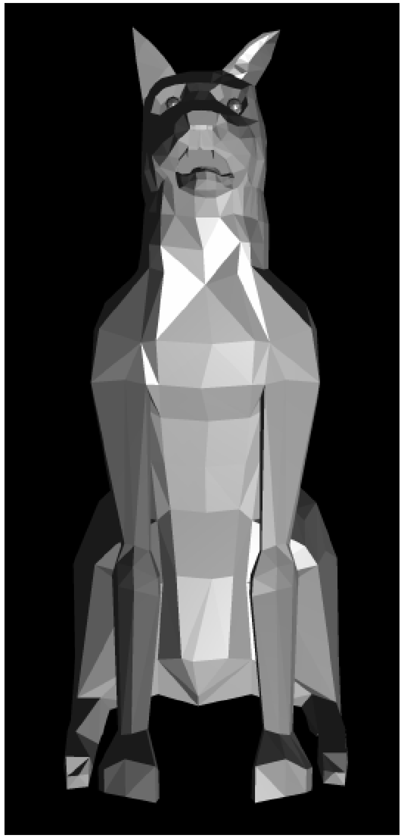
Embedding the code **010**

The vertices are changed along the direction of the normal to the ellipsoids

Experimental results



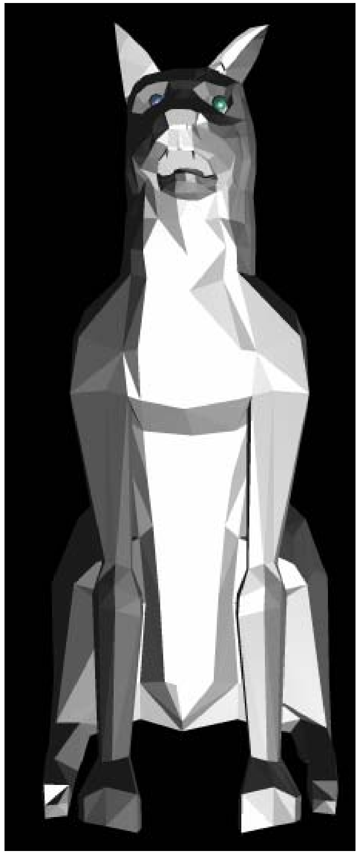
Original



“Dog”

Bounding planes

embedding

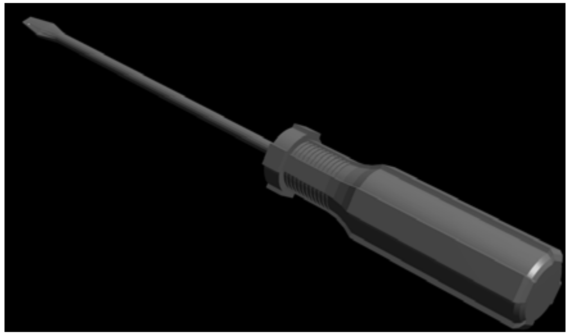


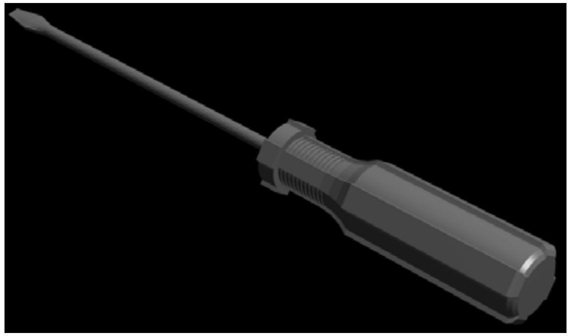
Bounding ellipsoids

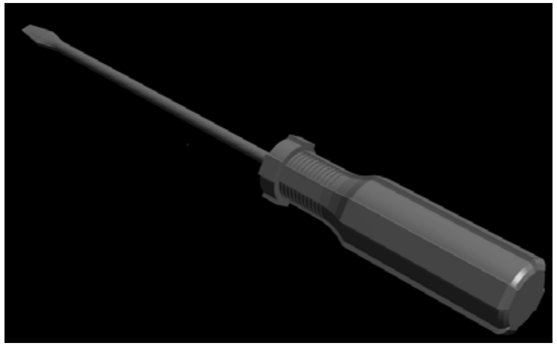
embedding

Experimental results

“Screwdriver”



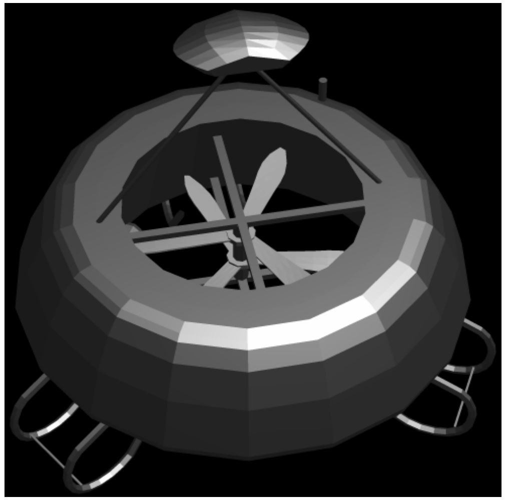


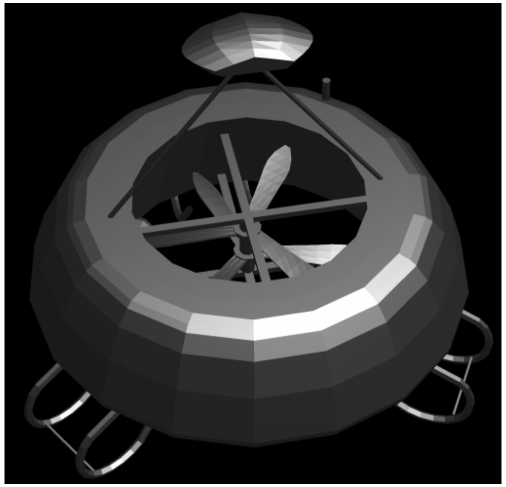


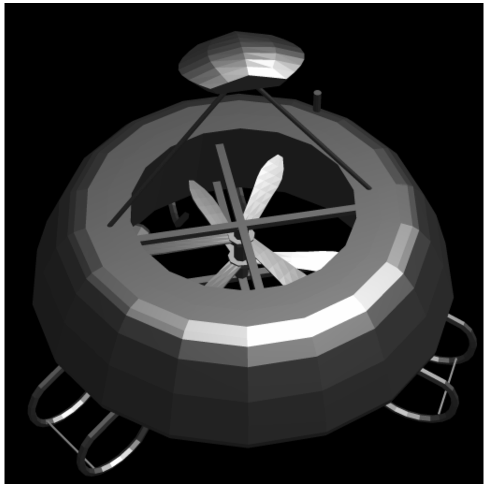
Original

Bounding planes embedding

Bounding ellipsoids embedding



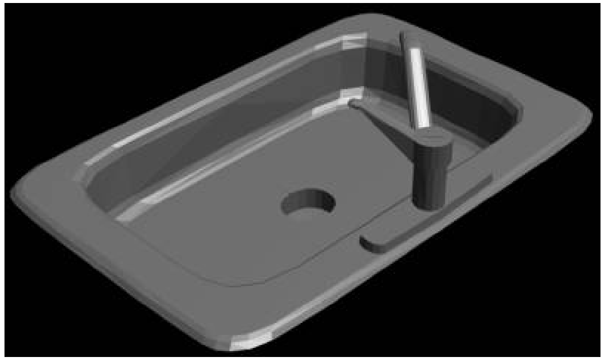




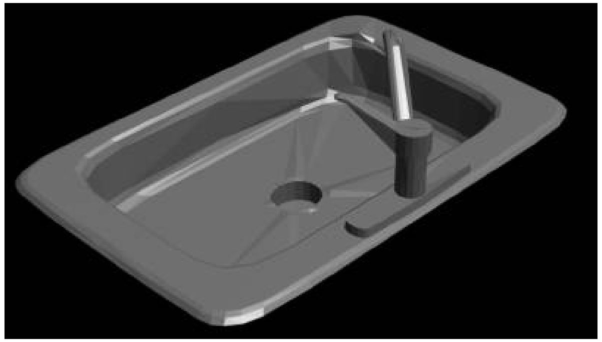
“Fan”

Experimental results

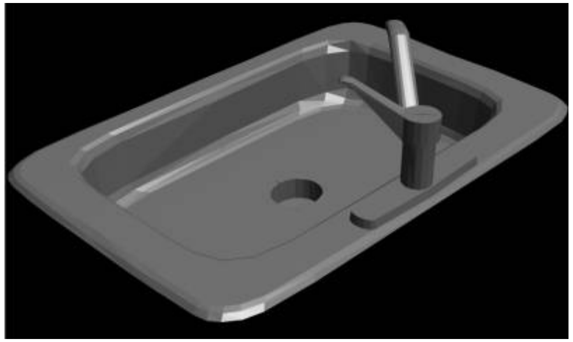
“Sink”



Original



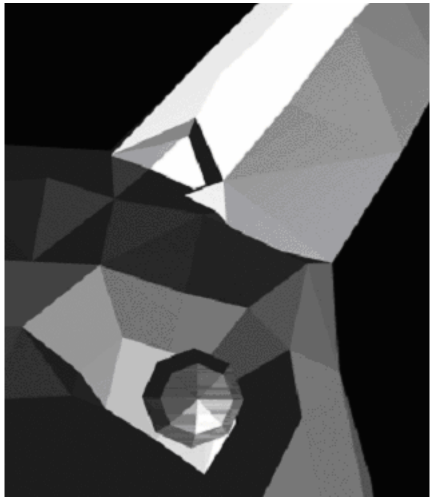
Bounding planes embedding



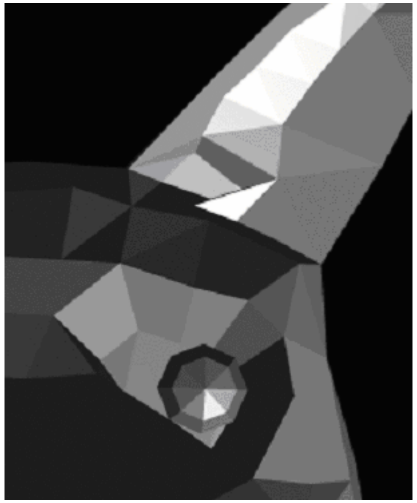
Bounding ellipsoids embedding

Experimental results

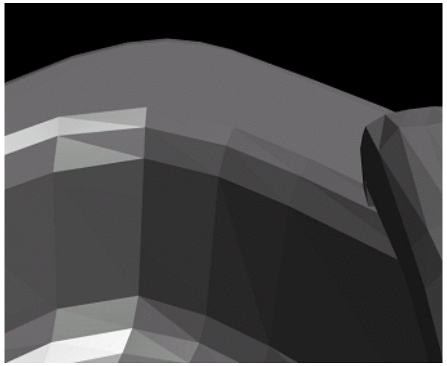
“Dog” Details “Sink”



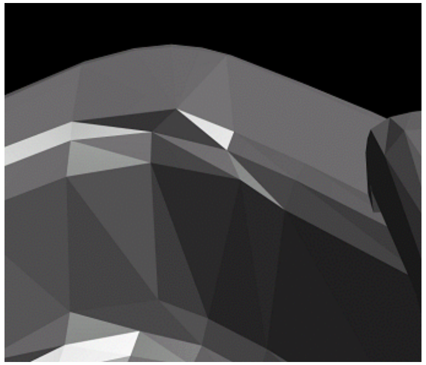
Original



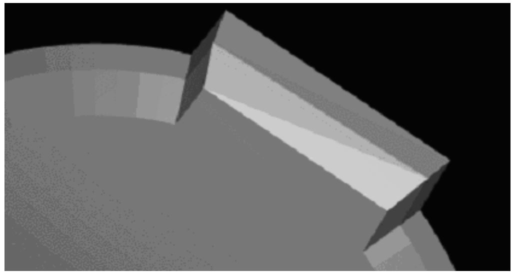
Bounding planes embedding



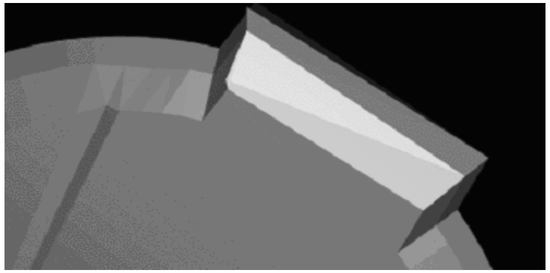
Original



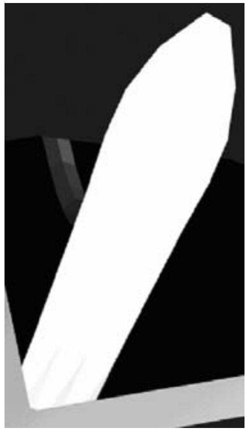
Bounding planes embedding



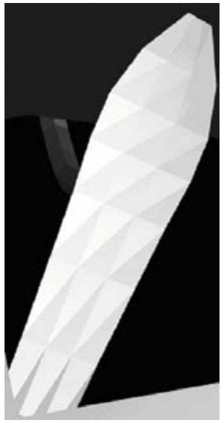
Original



Bounding ellipsoids embedding



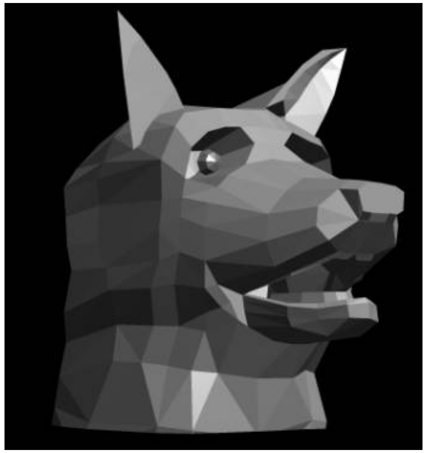
Original



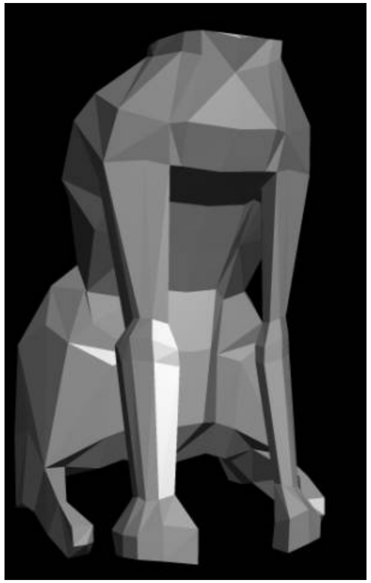
Bounding ellipsoids embedding

Experimental results

Cropping the “Dog” object



“Dog head”



“Dog body”

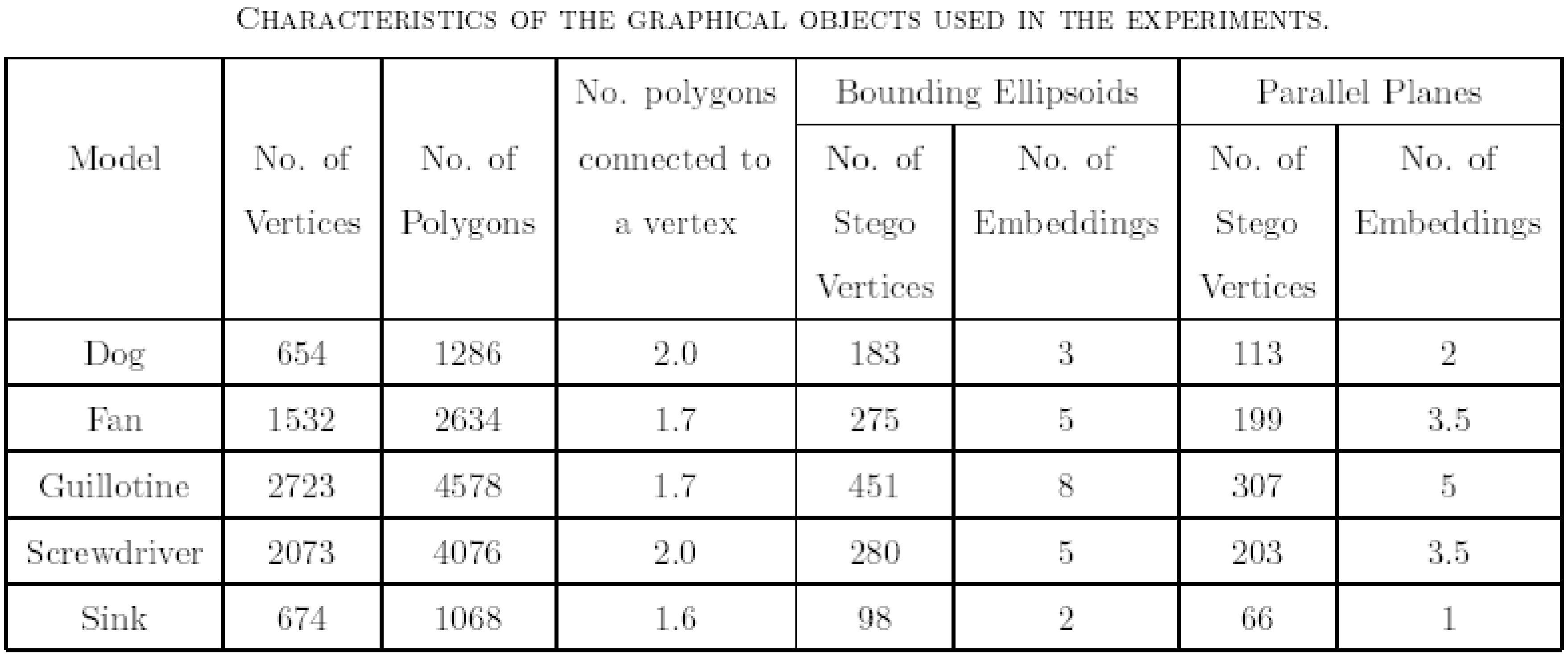
Watermark detected

100%

Watermark detected in

proportion of 50%

Experimental results

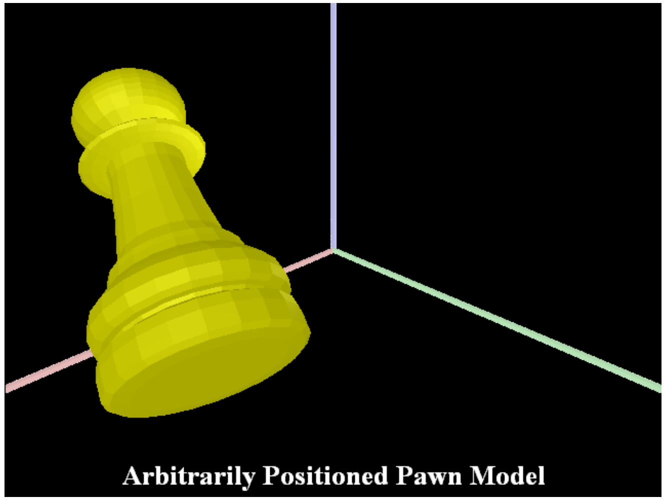


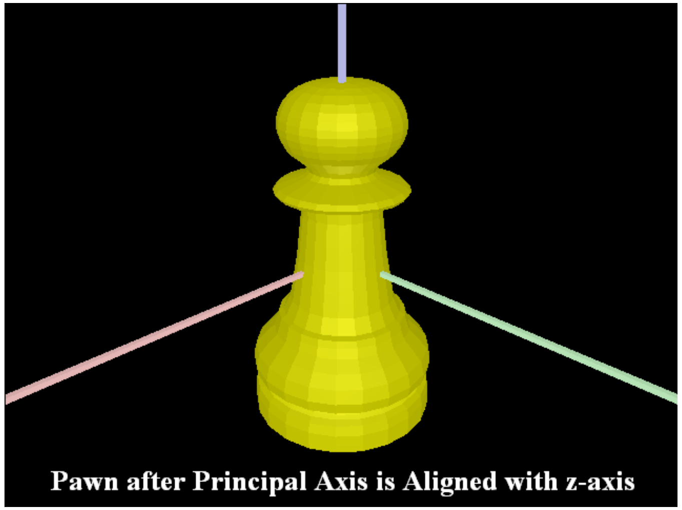
Characteristics of the graphical objects used in the experiments.

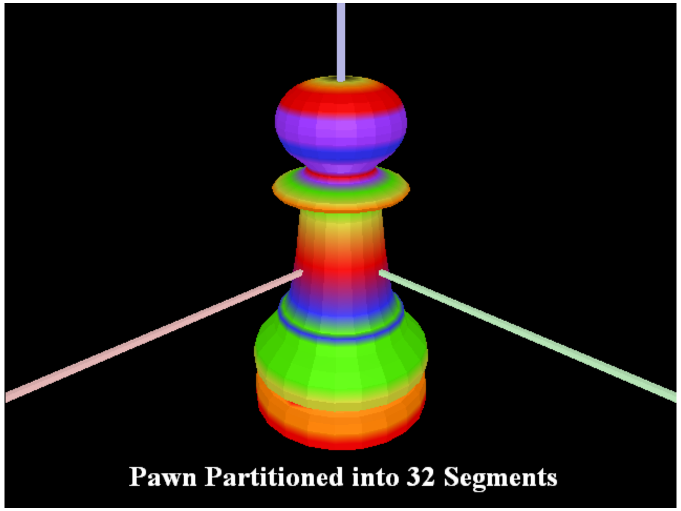
Model	No. of Vertices	No. of Polygons	No. polygons connected to a vertex	Bounding Ellipsoids		Parallel Planes	
				No. of Stego Vertices	No. of Embeddings	No. of Stego Vertices	No. of Embeddings
Dog	654	1286	2.0	183	3	113	2
Fan	1532	2634	1.7	275	5	199	3.5
Guillotine	2723	4578	1.7	451	8	307	5
Screwdriver	2073	4076	2.0	280	5	203	3.5
Sink	674	1068	1.6	98	2	66	1

3D watermarking using the principal axis alignment

- S. Zafeiriou, A. Tefas, I. Pitas, “Blind Robust Watermarking Schemes for Copyright Protection of 3D Mesh Objects”, *IEEE Trans. on Vis. and Comp. Graphics*, vol 11, no 5, pp 596-607, 2005.
- Stages:
  - Principal axis alignment of the graphical object
  - Conversion to spherical coordinates
  - Object partition in slices, each to embed a bit of the code
  - Detection - statistical


Arbitrarily Positioned Pawn Model


Pawn after Principal Axis is Aligned with z-axis


Pawn Partitioned into 32 Segments

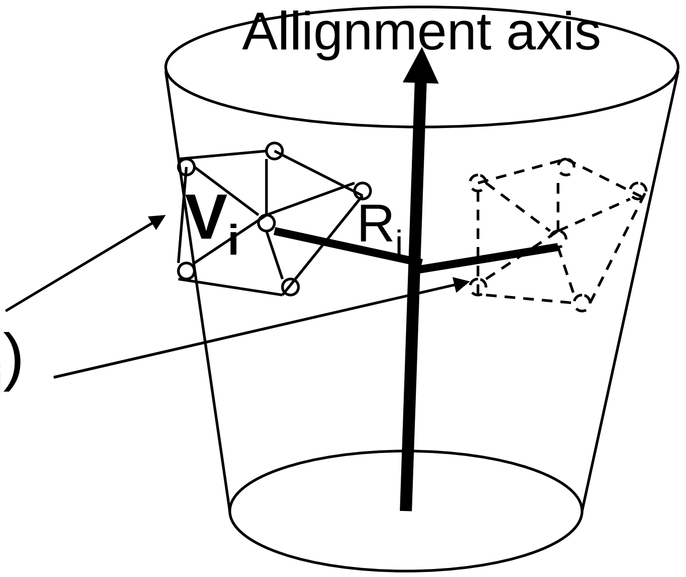
3D watermarking using the principal axis alignment

- Watermarking idea - Introduce a change in the symmetry
- For each section:
  - Choose a vertex $\mathbf{V}_i$ defined by the spheric coordinates (R,f,t).
  - Define a neighbourhood for each vertex $\mathcal{N}(\mathbf{V}_i)$
  - Average the length of distance from a vertex to the principal axis:

$S\ R_k/N$, k in $\mathcal{N}(\mathbf{V}_i)$, where N is the number of data in the neighbourhood

  - Calculate the difference

$$d_i = R_i - S\ R_k/N$$



Symmetrical neighbourhoods

$\mathcal{N}(\mathbf{V}_i)$

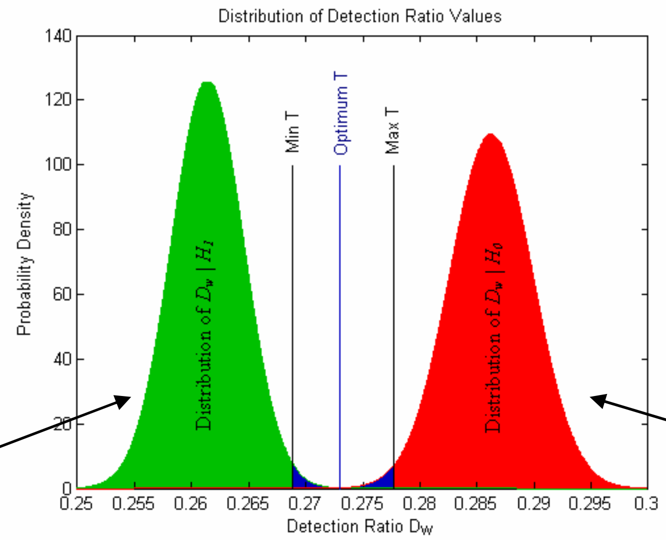
Statistical watermark detection

- The key determines:
  - how the object is split in transversal segments along the main axis
  - which vertices are chosen
- Detection – statistical testing of symmetry for the distribution of d_i
- If the key is wrong the algorithm will be “looking” for the wrong labels and in the wrong places

Hypothesis

testing for

detection



A threshold T is chosen

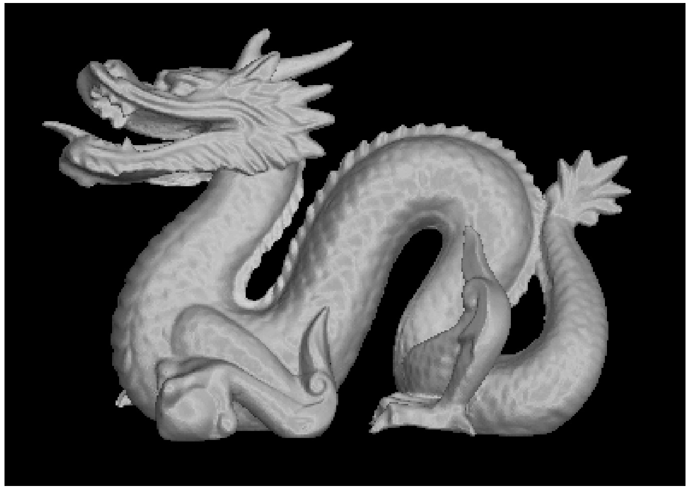
in order to minimize

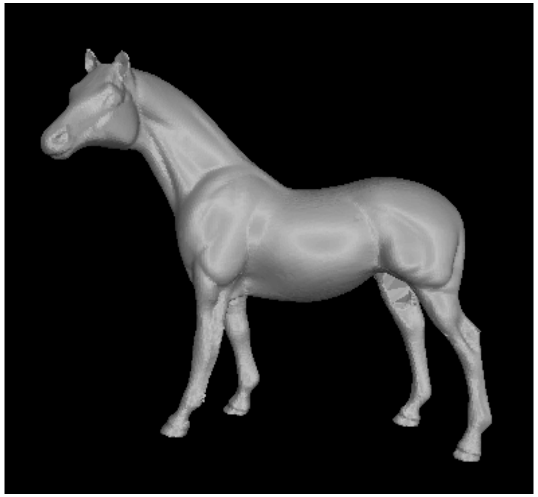
the error

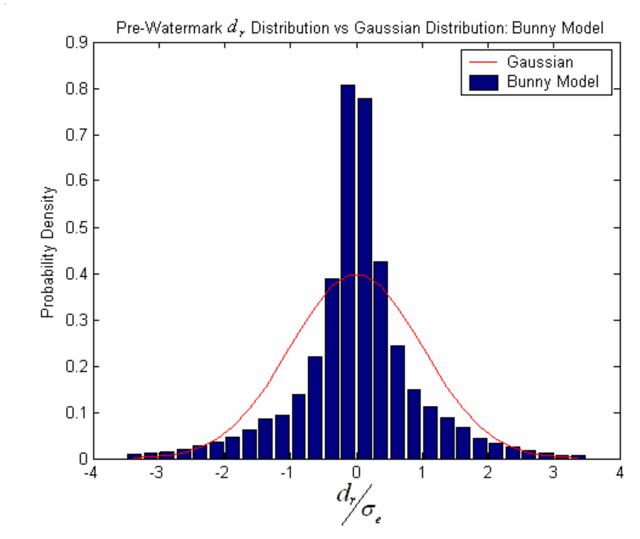
Hypothesis 1 – Watermark detected

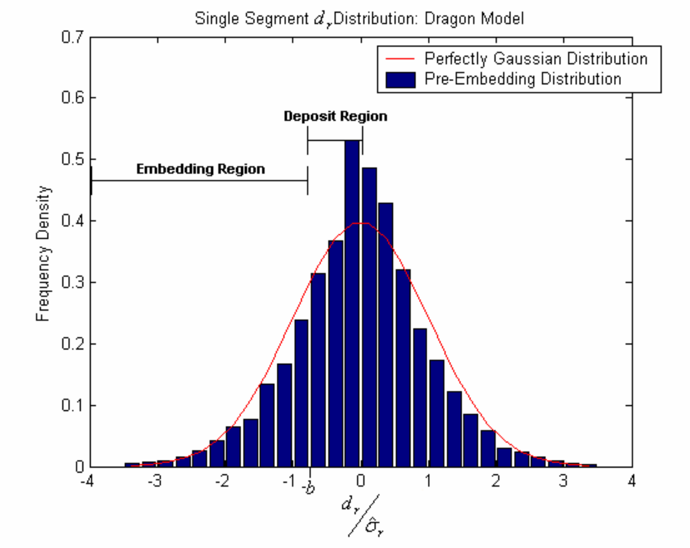
Hypothesis 0 – No watermark

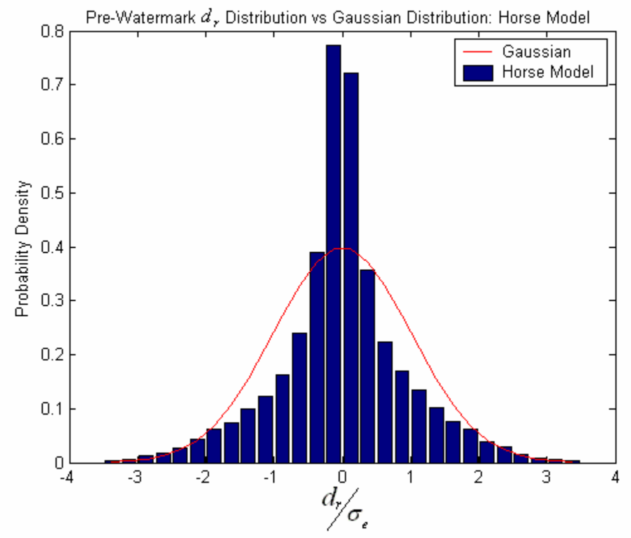
Distribution of differences D_i











Watermark embedding

Model	***Number of Vertices in Model***	***Percentage Modified***
Happy Buddha	543,652	1%
Dragon	437,645	1%
Skeletal Hand	327,323	1%
Horse	48,485	2%
Stanford Bunny	34,834	3%
Porsche	5,247	24%
Pawn	1,036	Not possible
Mickey Mouse	960	32%

The Percentage of an Objects Vertices That Must Be Modified by the Embedding Algorithm for a Watermark to be Successfully Created

Measuring distortions

- Signal to noise Ratio calculation

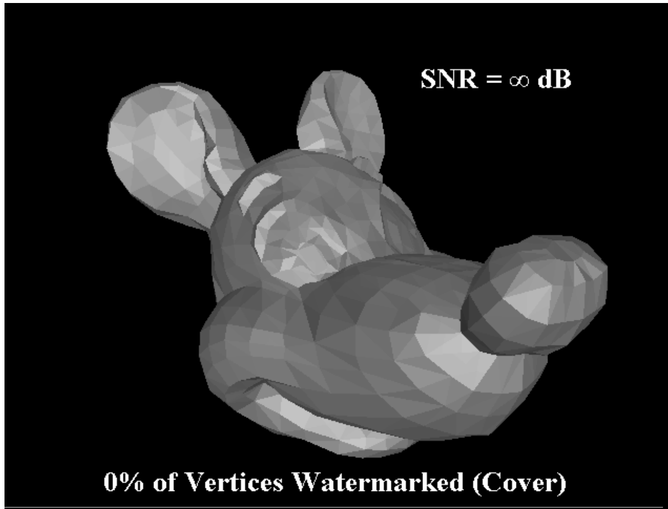
$$\mathrm{SNR} = 10\log_{10}\left(\frac{\sum_{i=1}^{n}(x_i^2 + y_i^2 + z_i^2)}{\sum_{i=1}^{n}(x_i' - x_i)^2 + (y_i' - y_i)^2 + (z_i' - z_i)^2}\right)$$

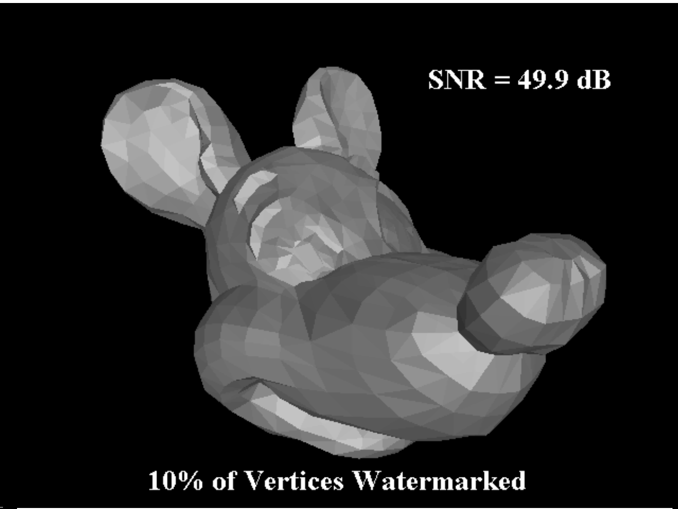
where (x,y,z) are the vertex coordinates

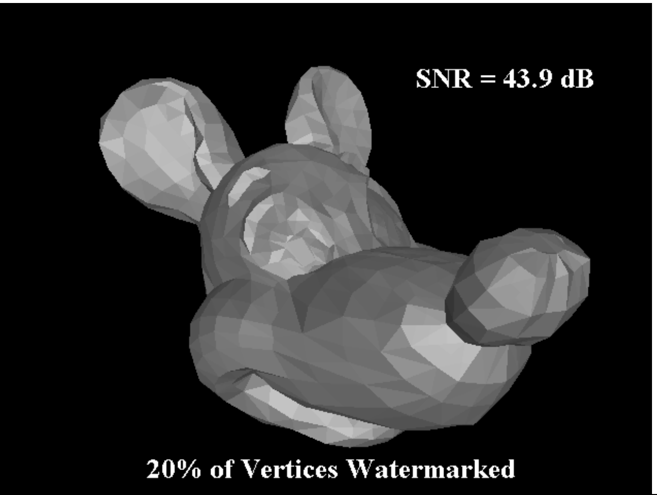
- Local measure of variance – for example calculating the variance in a neighbourhood

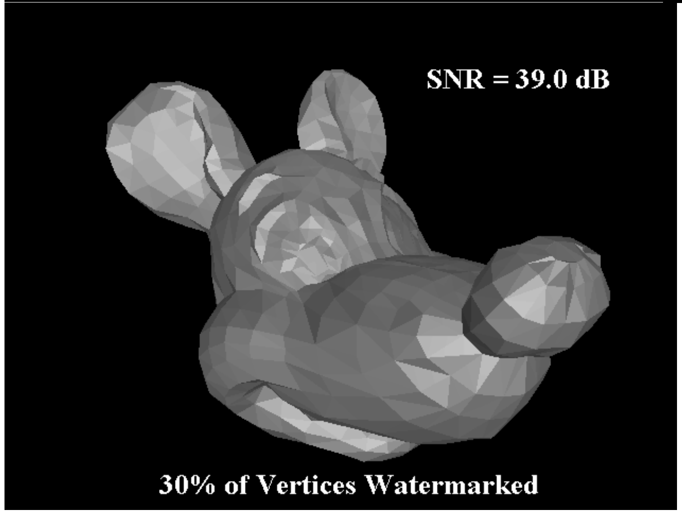
- Hausdorff distance between 3-D shapes

Visibility tests of 3D watermarking

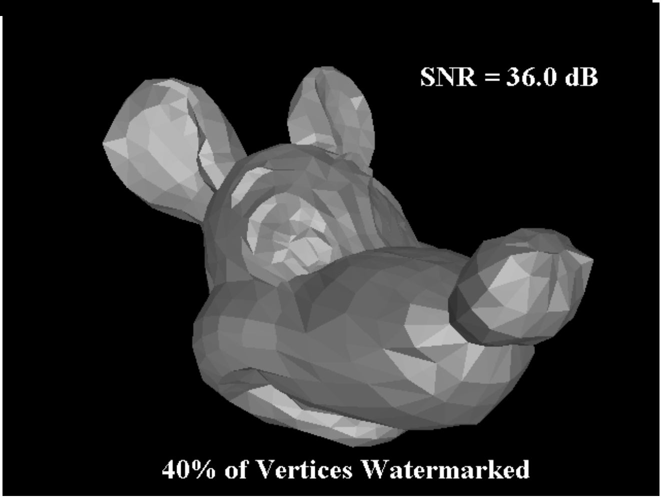




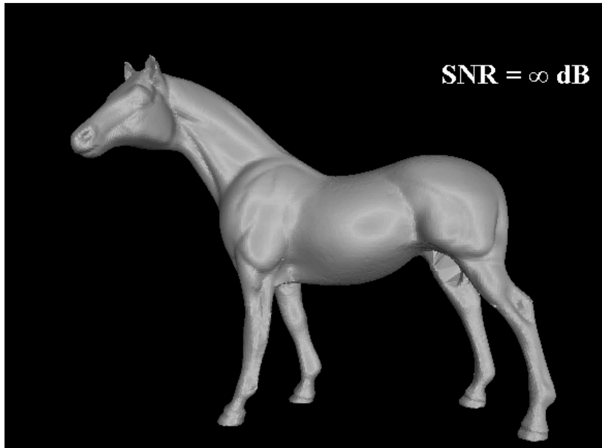




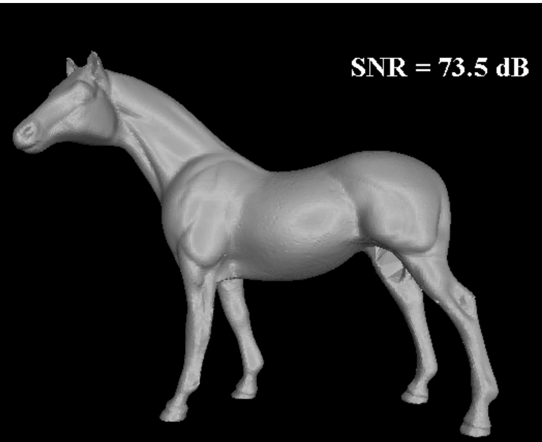
“Mickey”



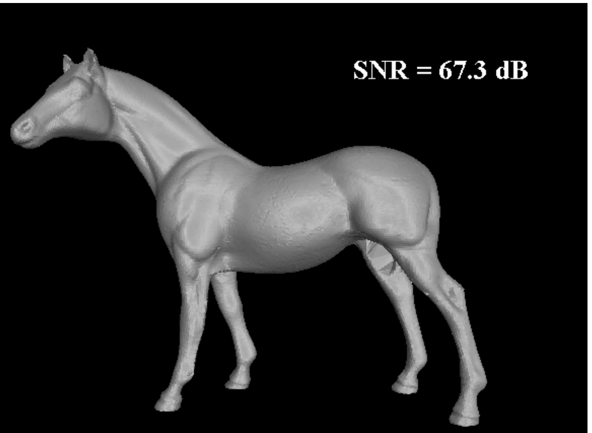
Visibility tests of 3D watermarking



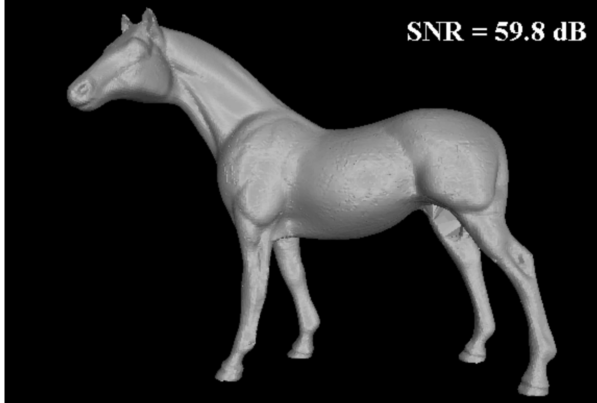
0% of Vertices Watermarked (Cover)



10% of Vertices Watermarked

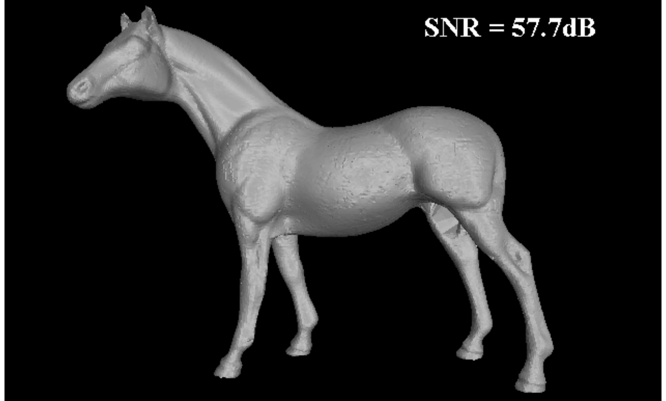


20% of Vertices Watermarked



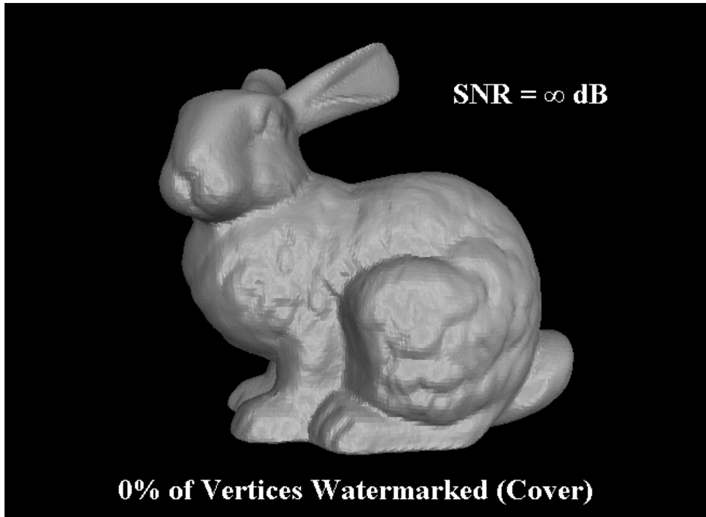
30% of Vertices Watermarked

“Horse”

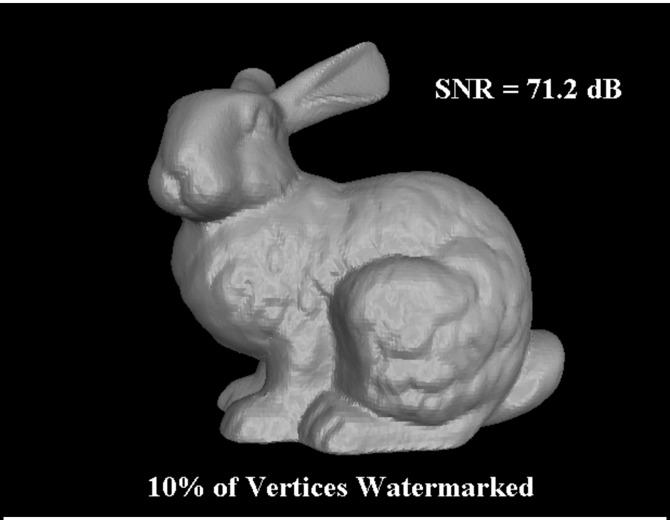


40% of Vertices Watermarked

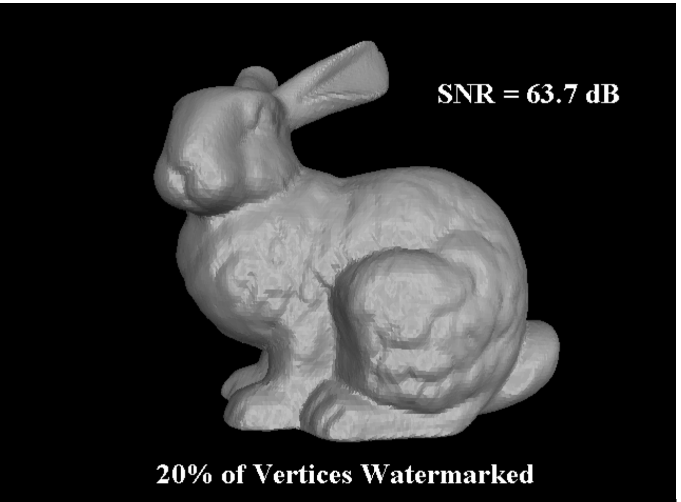
Visibility tests of 3D watermarking



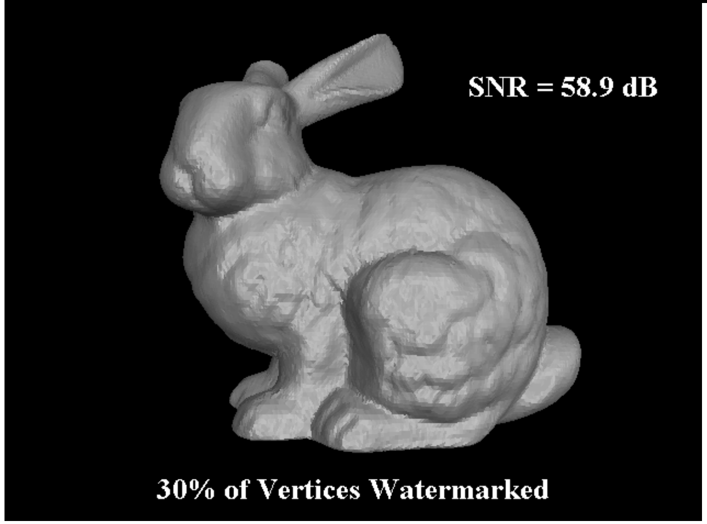
0% of Vertices Watermarked (Cover)



10% of Vertices Watermarked

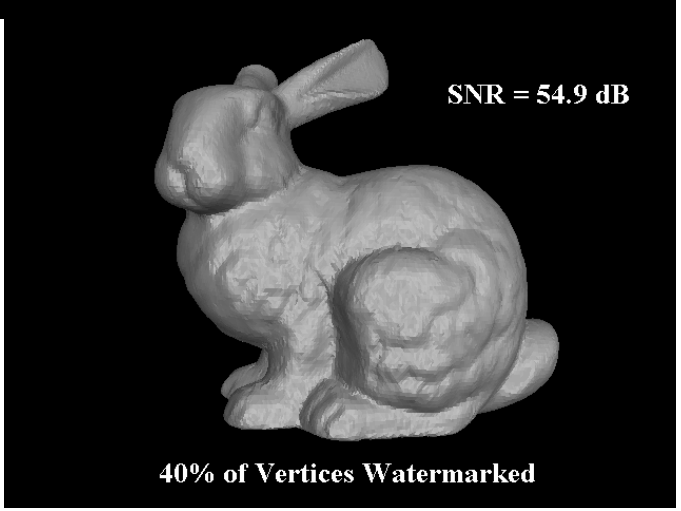


20% of Vertices Watermarked



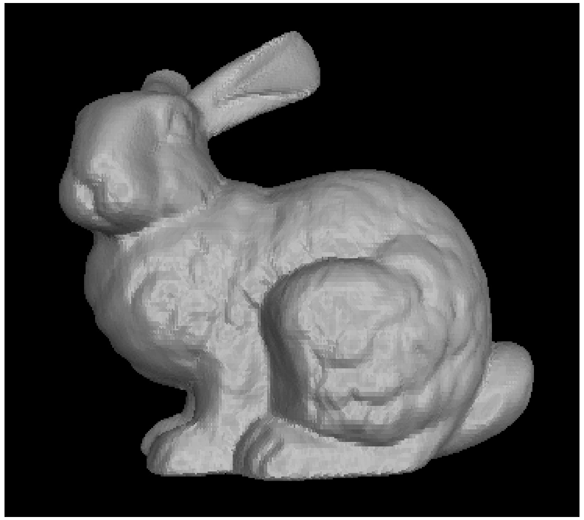
30% of Vertices Watermarked

“Bunny”



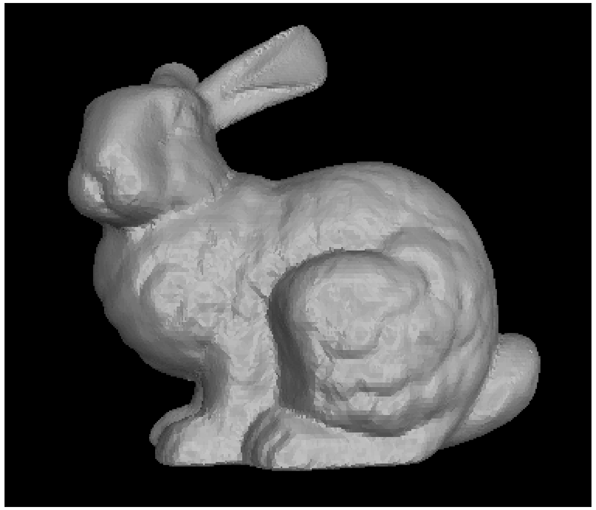
40% of Vertices Watermarked

Attacks on graphical objects

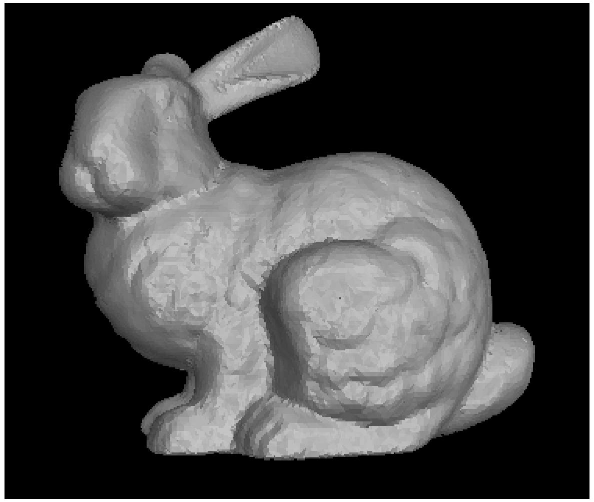


Original

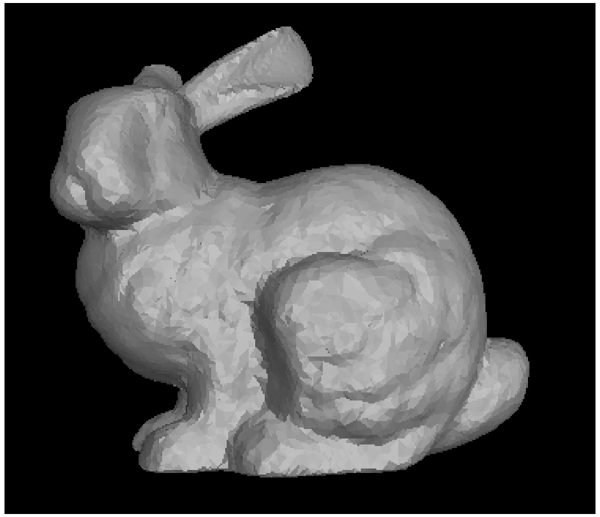
Fails>10%



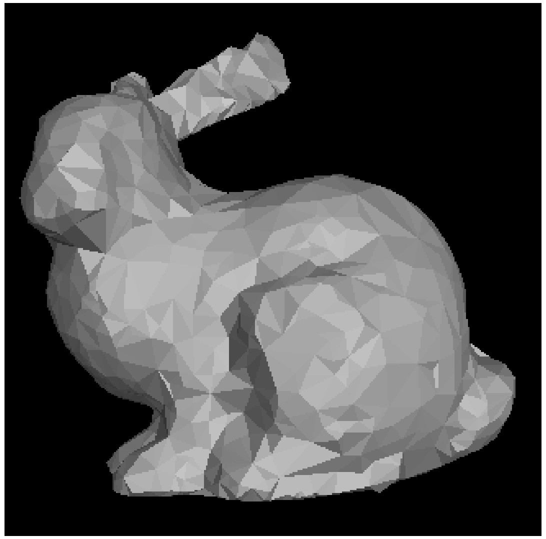
20% reduction



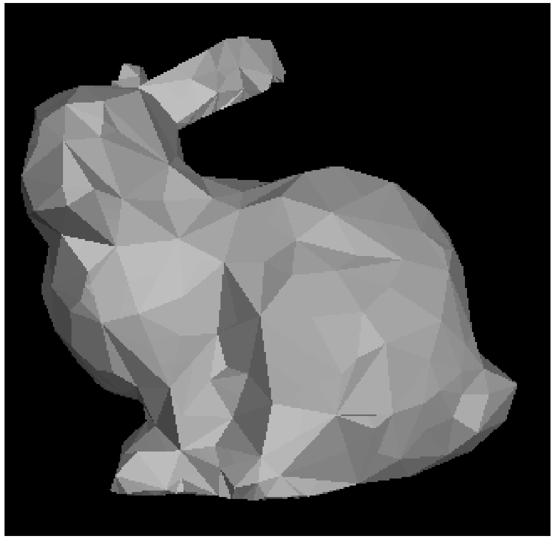
40 % reduction



70 % reduction



95% reduction

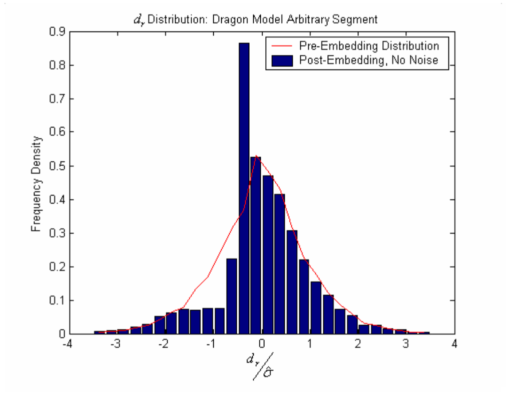


98 % reduction

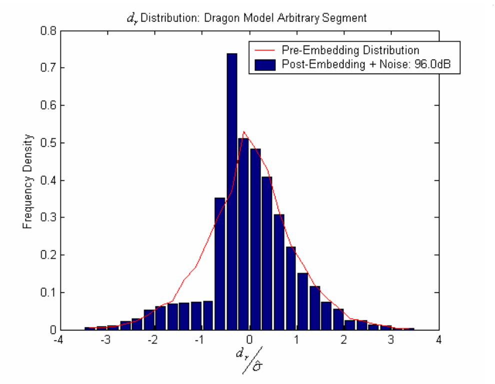
Robustness tests

- Noise addition – effects of adding Gaussian noise to Dragon model

Distribution of distances

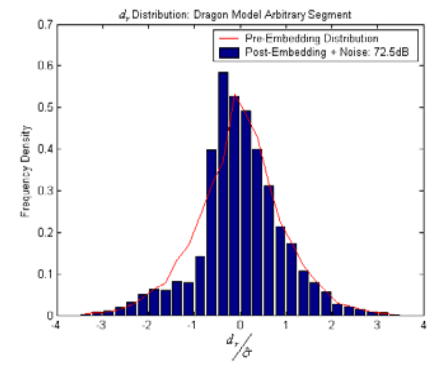


Watermarked model



Noise

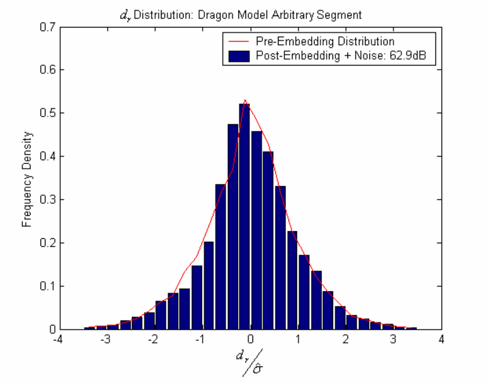
SNR = 96 dB



Noise

SNR = 73 dB

Fails

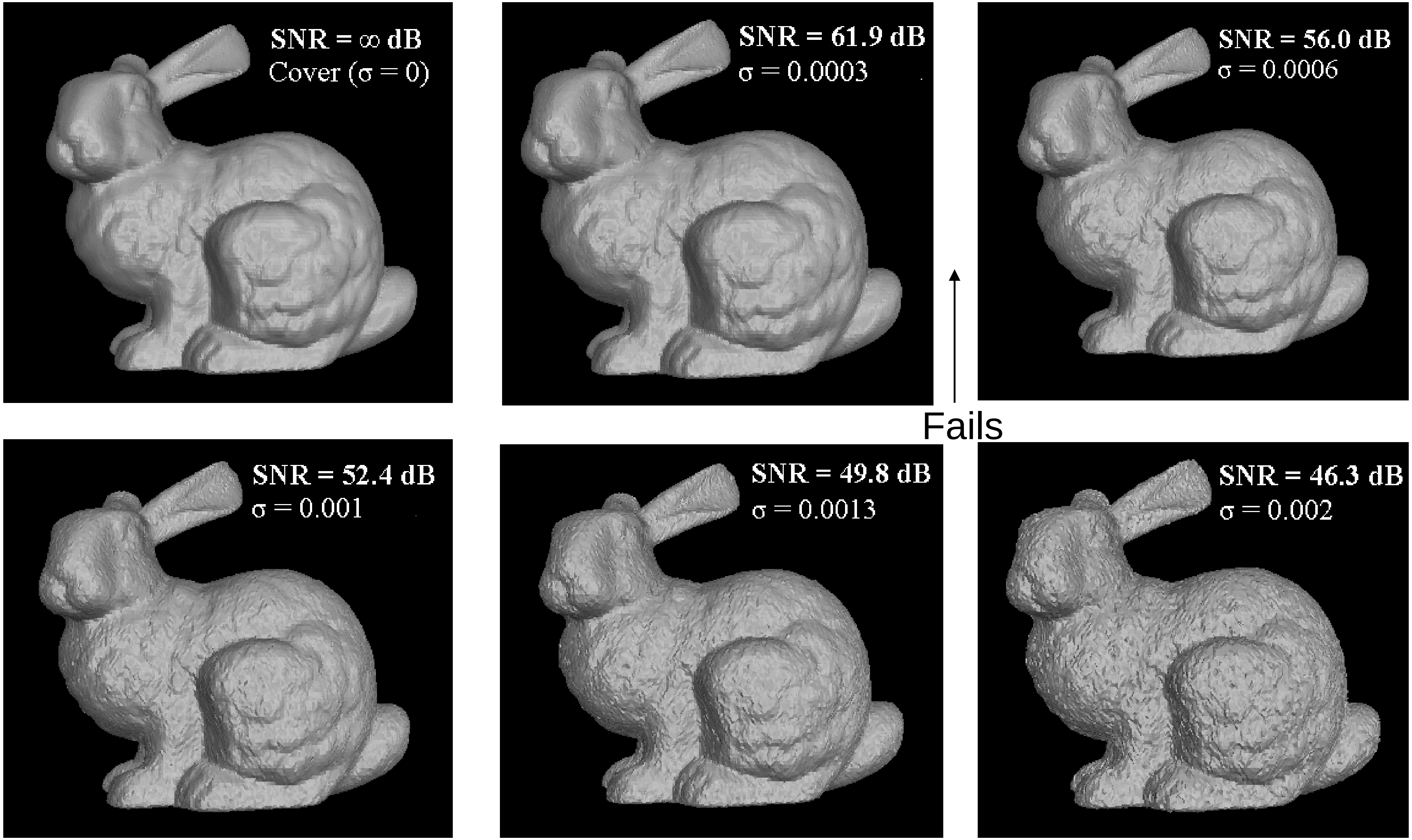


Noise

SNR = 63 dB

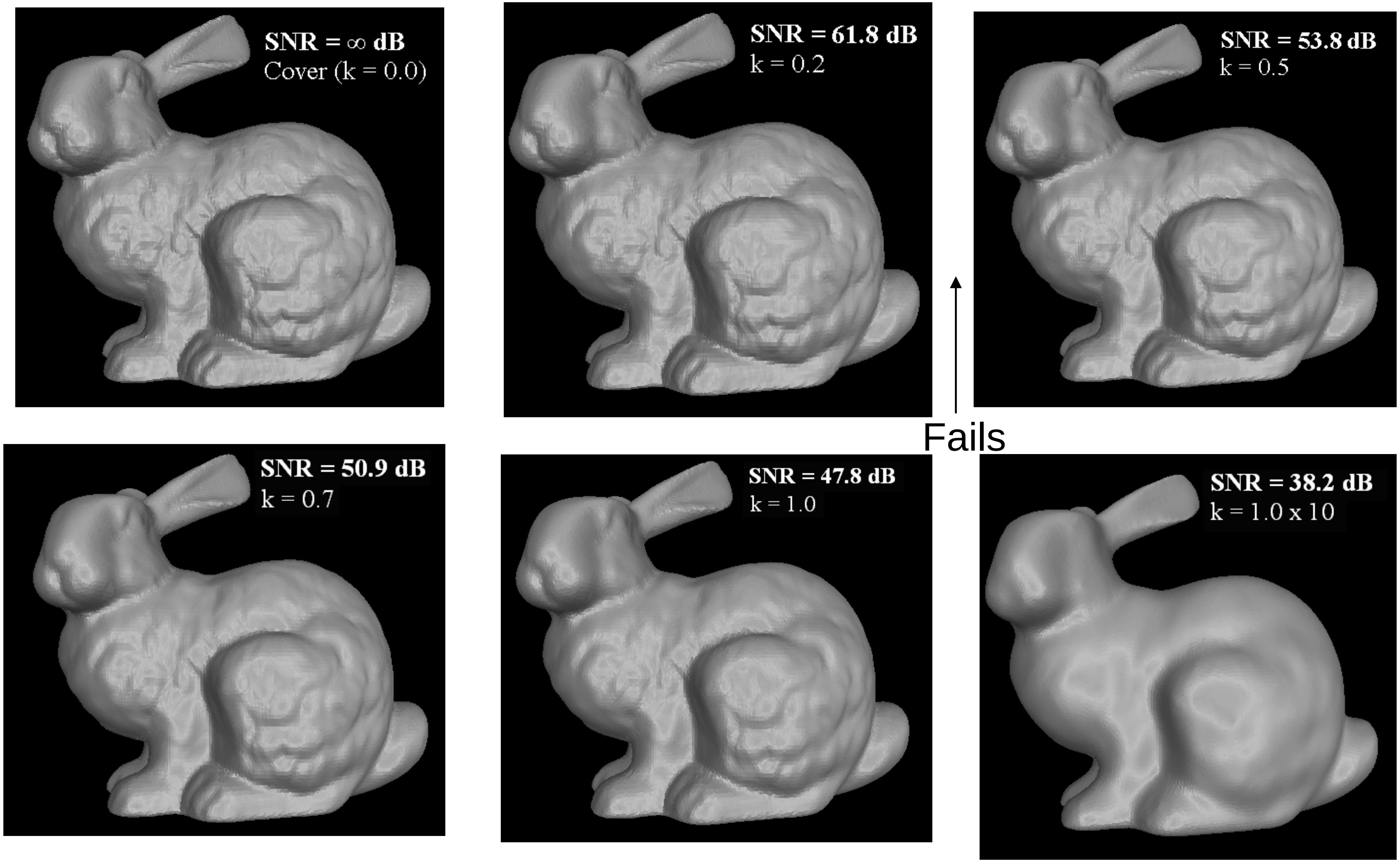
- The effect is to lead towards re-normalization of the distribution of distances – thus towards erasing the watermark
- Robustness to smoothness of the mesh – achieved by simply averaging the vertex locations – up to 62 dB for the Dragon model.
- Cropping that is not consistent with the direction of the principal axis would affect the orientation of the orientation of the principal axis and consequently the watermark.

Attacks on graphical objects



Gaussian noise addition

Attacks on graphical objects



Fails

Mesh surface smoothing by averaging coordinates

Performance assessment

Spatial techniques	***Bit-Capacity***	***Similarity transform***	***Signal process. attacks***	***Local deformation and cropping***	***Connectivity attacks (mesh simpl.)***
(Yeo, Yeung, 1999)	++	--	--	Localization	--
(Lin et. al., 2005)	++	--	|	Localization	--
(Ohbuchi, Masuda and Aono, 1998)	++	++	-	+	--
(Benedens, 1999) *	|	|	-	-	-
(Cayre,Macq,2003)	++	++	-	--	--
(Bennour,Dugelay,2006)	|	|	+	+	-
(Bors, 2006)	+	++	|	-	--
(Zafeiriou, Tefas, Pitas, 2005) *	--	|	+	-	+
(Cho, Prost, Jung, 2007)	--	|	+	-	+

Performance assessment

Transform domain techniques	***Bit-Capacity***	***Similarity transform***	***Signal process. attacks***	***Local deformation and cropping***	***Connectivity attacks (mesh simpl.)***
(Kanai, Date, Kishinami, 1998) **	|	+	**-**	**-**	**--**
(Praun, Hoppe, Finkelstein, 1999) **	**--**	|	++	++	|
(Yin et. al, 2001) **	**-**	|	+	**-**	|
(Ohbuchi, Mukaiyama and Takahashi, 2002) **	**--**	|	++	++	|
(Cayre et. al., 2003)	**--**	+	+	++	**--**
(Uccheddu, Corsini, Barni, 2004)	**--**	**-**	+	**-**	**-**
(Wu, Kobbelt, 2005) **	**--**	|	++	++	|

Non-blind methods – marked with **   Semi-blind methods – marked with *

Applications

- Depending on the application the respective watermarking algorithm should have certain properties enhanced.

- Copyright protection (digital rights management)
  - Assisting outsourcing and artists rights in graphical object markets
- Authentication
- Recording of parameters for various reasons
  - For 3-D graphical object database management
  - For controlling certain properties of the graphical object such as motion, interaction, “behaviour”, etc.
- Information hiding
  - For example hiding the attributes (colour, texture) into the geometry
  - Steganography

Conclusions

- Several 3-D watermarking approaches are analysed
  - Spatial techniques – have better capacity of embedding and usually good robustness to the affine and other similarity transforms
  - Transform domain techniques – they are more robust but many of them are non-blind
- Visual hiding methods can be used to mask the distortions introduced by watermarks – using postprocessing
  - Applying texture
  - Applying colour, texture or shading (particularly Phong shading)
  - Applying 3-D mesh variations – without affecting the watermark
- Other methods – could disregard the connectivity information – for example by embedding the information into clouds of vertices.
- There is no “perfect” 3-D watermarking method

- Using new shape modelling techniques !