Списание за наука
„Ново знание“
ISSN 2367-4598 (Online)
ISSN 1314-5703 (Print)
Академично издателство „Талант“
Висше училище по агробизнес и развитие на регионите - Пловдив

New Knowledge
Journal of Science
ISSN 2367-4598 (Online)
ISSN 1314-5703 (Print)
Academic Publishing House „Talent“
University of Agribusiness and Rural Development
Bulgaria

http://science.uard.bg

WEB BASED EXTRACTION METHOD OF STEGANOGRAPHIC MESSAGE FROM 3D GRAPHIC MODEL

Konstantin Tsvetkov, Delyan Sarmov
Bishop K. Preslavski University of Shumen, Bulgaria

WEB БАЗИРАН МЕТОД ЗА ИЗВЛИЧАНЕ НА СТЕГАНОГРАФСКО СЪОБЩЕНИЕ ОТ 3D ГРАФИЧЕН МОДЕЛ

Константин Цветков, Делян Сърмов
Шуменски университет „Еп. К. Преславски“

Abstract: In steganography, the use of various raster graphic file formats, as containers, has been explored. The present work aims to propose a method for inserting a message into a model built using 3D modeling software and stored in vector file format. These formats use standard libraries, supported from browsers compatible with WebGL, without any other additional software.

Keywords: obj, 3D modeling, steganography, javascript, webgl.

INTRODUCTION

In steganography, the use of various raster graphic file formats, as containers, has been explored. Methods for inserting messages have been developed as well as for a research of files about the presence of some added information [1,2,3,4,5].

The aim of the present study is to introduce a web based method extracting a message from a model built by 3D modelling software and stored in vector file format. This is the purpose for analyzing some libraries for visualization of 3D models which have been controlled by modern browsers. The support of „WebGL“, as well as the development of some additional javascript libraries used for ready-to-use models storage made 3D content more and more popular. Modelling is a creative process in which the amount and type of information included is subjective. This is the reason for the difficulties in the development of methods for standard automated search for hidden information.

Examples for program libraries: webgl-obj-loader [6], COLLADA importer [7], p5js, tree.js, etc. There are also libraries working in node.js. Such libraries are not part of the operating system but supported by all modern browsers. They might be used in desktop and mobile systems.

The present work introduces the demonstration algorithm for loading of Wavefront .obj file format, based on libraries which have been worked out by webglfundamentals.org team..

STRUCTURE OF WAVEFRONT OBJ FILES

Obj format supports some description of lines, polygons, curves and surfaces. The lines and polygons are described by points. The curves and surfaces are described by control points some information depending from the curve type. Points in the vector graphics are called vertices and in obj files vertices coordinates are the main part of the file [8,9,10,11,12]. Every not empty and uncommented line in obj files consists of a key word describing the following data:

v - geometric vertices
vt - textured vertices
vn - vertex normals

There are key words for groups:

g - name of group
s - Smoothing group
mg - Merging group
o - Object name

There are also saved words for curve parameters, surfaces, texture, material of objects, shadows, etc.

CHOOSING SOME INFORMATION FROM OBJ FILE TO BE USED AS A CONTAINER

In one standard obj file the biggest part of the information presents the description of vertices by which the model has been built. It has got the following format:

```
v -0.083380 -0.160176 -0.064195 0.105882 0.462745 0.333333
vn 0.000593 0.000000 -0.005565
vt 0.055954 1.000000
v -0.088054 -0.160176 -0.065120 0.105882 0.462745 0.333333
vn 0.000296 0.000000 -0.002782
vt 0.055954 1.000000
v -0.088054 0.160176 -0.065120 0.105882 0.462745 0.333333
vn -0.001791 0.000000 0.002408
vt 1.000000 1.000000
```

Lines in the file begin with the saved word „v“, followed by the three coordinates of the point in the surface. One line in a file is one vertex. The three number values are the coordinates in the surface by the axis „x“, „y“ and „z“. The following values to the end of the line are the color in the RGB scheme.

Each of the three values of geometric vertices, textured vertices, vertex normals[9] and colour can be used as a message container. Such type of algorithms exists and can be applied using data from an existing obj file or be executed using the model during the first file save. In the current example a modification of the sixth character after the decimal point in the z coordinate axis is used. The value is modified with 1/100000, as the even designates „0“, the odd one - „1“.

Here is used a text message, from which every sign has to be presented with a binary code by some code table. In the given example the standard ASCII code table will be used.

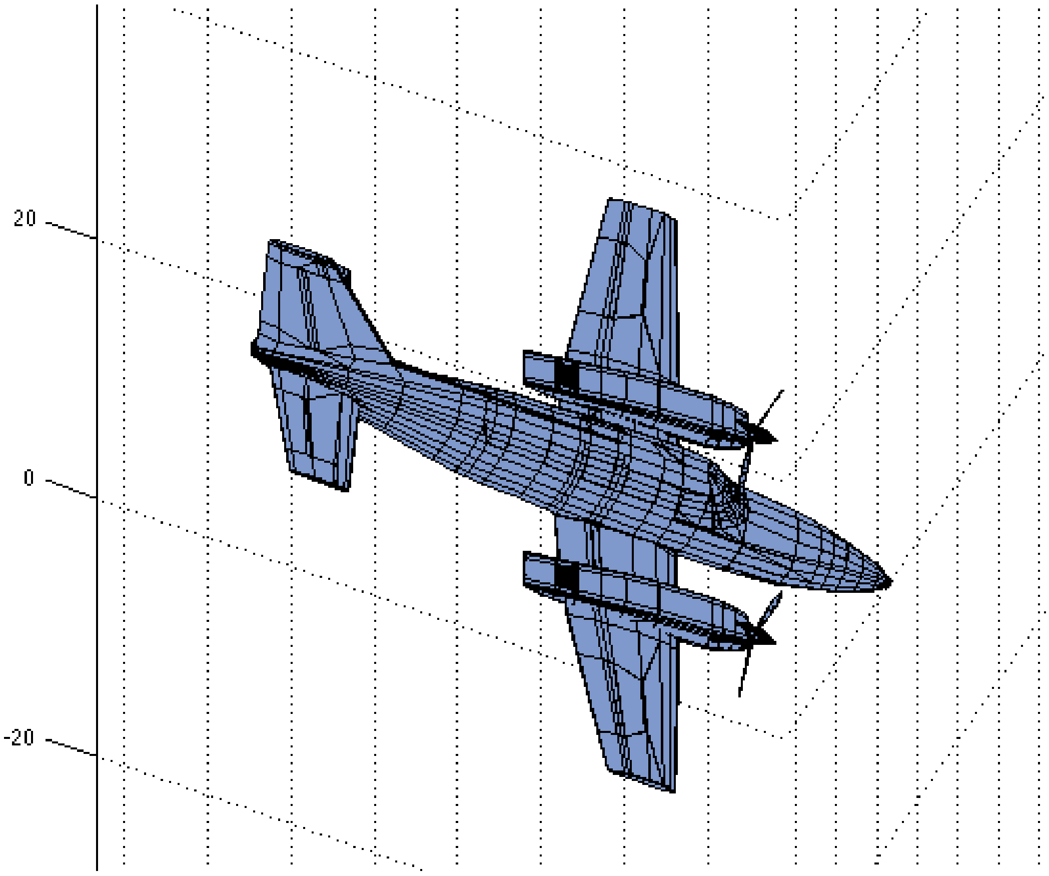
For example, the text message „Hello_there“ is presented in decimal format as:

72 101 108 108 111 32 116 104 101 114 101 4

in binary format:

01001000 01100101 01101100 01101100 01101111 00100000 01110100 01101000 01100101 01110010 01100101 00000100

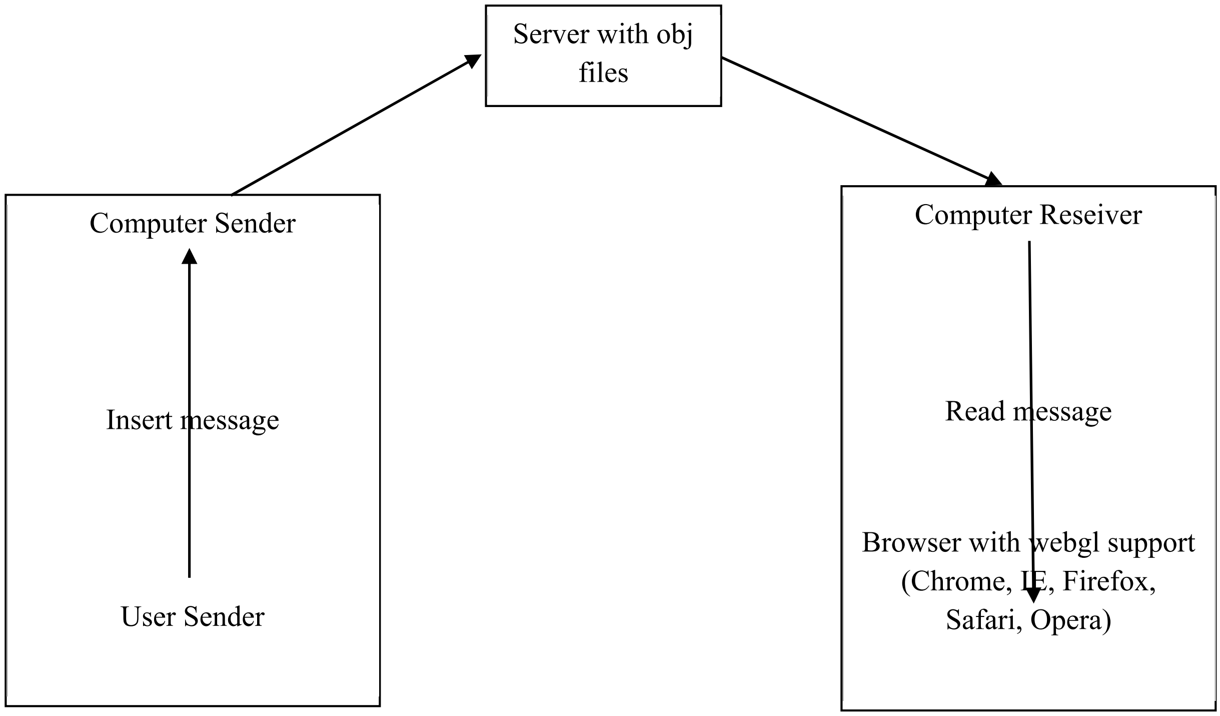
Where the last byte is the symbol EOT (end of transmission) with ASCII code 4. It serves as the end of the message.



Pic. 1. An example of 3D modeling

In the given model (Pic. 1) are used 3745 vertices for plane modelling. The file contains information about points set to six decimal places. In practice, however, due to the peculiarities of the used rendering algorithms, as well as the hardware used, the information about the vertices is rounded before being visualized. The example discussed above 96 vertexes will be used to store the “Hello_there” message.

Pic. 2 shows the communication process. The present work focuses on extracting a steganographic message using only a browser with standard libraries (WebGL). This allows the use of the method of any desktop or mobile system without the need of additional software. Depending on the usage of obj files, there may be some variation in the parameter used to insert the message.



Pic. 2. Communication sender-receiver

The following information in the article points out a method for extracting of a hidden message, with a simple modification of the javascript script using WebGL when visualizing an obj file.

EXTRACTION OF INFORMATION FROM AN OBJ FILE

A code fragment, in which the extraction of the hidden information form an obj file is executed:

```
global definitions:
var mess = '';
var mess_end = false;

variable in main function:
var ch_code = '';

In loading the coordinates of vertexes.
const keywords = {
 v(parts) {
/* insert code here*/
 if (parts.length > 3) {
 objPositions.push(parts.slice(0, 3).map(parseFloat));
 objColors.push(parts.slice(3).map(parseFloat));
 } else {
 objPositions.push(parts.map(parseFloat));
 }
 }
```

The following program code is inserted at the indicated location on the third line.

```
if ( !mess_end) {
 if (ch_code.length > 7) {
 if (parseInt((ch_code),2) == 4) {
 mess_end = true;
 }
 else {
 mess = mess + String.fromCharCode(parseInt((ch_code),2));
 ch_code = '';
 }
 }
 ch_code=ch_code+((Math.trunc(parts.slice(2,3)*1000000)%2)+2)%2;
}
```

The variable "mess" contains the extracted message and can be visualized out of the script. Sample code that can do this:

At the end of the main function the value of the variable "mess" is written.

```
document.getElementById('myMess').innerHTML = mess;
```

In the body of the Web page is visualised.

```
<canvas id="canvas"></canvas>
<p></p>
<p><div id="myMess"></div></p>
```

A full list of the unmodified libraries used in the experiment is available at: webglfundamentals.org

After downloading the html file to the local computer and making the above given corrections.

After copying the HTML file and adding the above mentioned changes it can be used for receiving the hidden information in the obj file. It is possible for the program code to be realized in a function in a separate library. Then, it is necessary only inclusion of the library and a function call. The rest javascript libraries, obj files, css scripts and other resources remain with the same URL addresses.

DISCUSSION

1)It is possible to use pre-coding of the information before inserting in obj file. More research is needed on the effectiveness of the available coding algorithms for some realization with standard javascript libraries and their compatibility with WebGL.

2)The process of inserting the message is similar but for the automation of the process in a Web based environment some access to a server architecture is required using protocols such as HTTP, FTP, SSH, etc. It is necessary an implementation of algorithm with some "server side" scripting languages such as PHP, Python, Perl.

3) For constant active support of a permanent steganographic channel is possible with the use of third party infrastructure. With this purpose, in a three-dimensional scene as a container for information, a scene parameter which is constantly synchronized between the participants has to be used. These parameters are most often the position of the camera, the direction of the camera, etc.

CONCLUSION

This method extracts information from Wavefront obj files by modifying a standard script that can be easily inserted into one of the common Web-based graphic libraries. With the given method, available third party Web sites can be used to provide a hidden transmission of information. The given 3D vector models contribute the approach of using the standard syntax of the files without interfering or changing their use. The other x and y coordinates or other parameters of the 3D model can be used with the same approach.

REFERENCES

1. Stoyanov, Borislav P., Stanimir K. Zhelezov, and Krasimir M. Kordov. "Least significant bit image steganography algoritm based on haotic rotation equations." Comptes rendus de l'Academie bulgare des Sciences 69.7 (2016)..
2. Железов, Ст., Х.Параскевов и др. Един подход за откриване на стеганографски манипулирани аудиофайлове чрез разлагане с wavelet. Научна конференция смеждународно участие, посветена на 105 годишнината от рождението на Джон Атанасов и Джон фон Нойман, Том 2, Шумен 2009., стр.126.
3. Станев, С. Стеганологична защита на информацията, Университетско издателство „Епископ Константин Преславски". Шумен, 2013. ISBN 978-954-577-825-4. 320.
4. Параскевов, Хр.,Стефанов, Ал., Съвременни стеганографски подходи в социалните мрежи, MATTEX 2018, Том 1, стр. 197-203.
5. F. Petitcolas, R. Anderson, M. Kuhn, "Information hiding a survey", Proceedings of the IEEE, vol. 87, pp. 1062-1078.
6. URL: https://www.npmjs.com/package/webgl-obj-loader
7. Miroslav Milivojević, I. Antolovic, Dejan Rancic, Evaluation and visualization of 3D models using COLLADA parser and WebGL technology ISBN: 978-1-61804-000-8 URL: https://www.researchgate.net/publication/229050567_Evaluation_and_visualization_of_3D_models_using_COLLADA_parser_and_WebGL_technology
8. Doria D., A Wavefront OBJ Writer for VTK, The VTK Journal, 01-28-2010 ISSN 2328-3459
9. B. Madoš, J. Hurtuk, M. Copjak, P. Hamaš, M. Ennert, "Steganographic algorithm for information hiding using scalable vector graphics images", Acta Electrotechnica et Informatica, vol. 14, no. 4, pp. 42-45, 2014, ISSN 1335-8243.
10. Branislav Madoš, Anton Balaz, Norbert Adám, Ján Hurtuk, Information Hiding into OBJ Format File Using Vector Steganography Techniques, 2018 IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI)
11. URL: https://www.fileformat.info/format/wavefrontobj/egff.htm
12. URL:http://homepages.gac.edu/~hvidsten/courses/MC394/projects/final/obj_io_modif232b.py