A study on the false positive ratio of Stegdetect

Omed Saleem Khalind
School of Computing
University of Portsmouth

Julio C. Hernandez-Castro
School of Computing
University of Kent

Benjamin Aziz
School of Computing
University of Portsmouth

{Omed.khalind, Benjamin.Aziz}@port.ac.uk
J.C.Hernandez-Castro@kent.ac.uk

Abstract

In this paper we analyse Stegdetect, one of the well-known image steganalysis tools, to study its false positive ratio. In doing so, we process more than 40,000 image files randomly downloaded from the internet using Google images, together with 25,000 images from the ASIRRA (Animal Species Image Recognition for Restricting Access) public corpus. The aim of this study is to help digital forensic analysts aiming to study a large number of image files during an investigation. The results obtained shows that the ratio of false positive generated by Stegdetect depends highly on setting the sensitivity value, and it is generally quite high. This should inform the forensic expert and help to better interpret results, particularly false positives. Additionally, we have provided a detailed statistical analysis for the obtained results to study the difference in 'difference in detection' between selected groups, close groups and different groups, of images. This method can be applied to any other steganalysis tools, which gives the analyst a better understanding of the results, especially when he has no prior information about the false positive ratio of the selected tool.

Keywords: Stegdetect, steganalysis, steganography, digital forensics, computer forensics, detection, false positive.

1 Introduction

The word steganography is derived from two Greek words (*stegano* and *graphos*) that respectively mean covered and writing. It can be defined as the art and science of hiding secret messages in different media (image, audio, video, text, etc.) so that it can be correctly received by another party without raising suspicion by any observer (Chandramouli & Memon, 2003). The main difference between steganography and cryptography is that the former tries to hide the very existence of the information exchange, while the latter is only interested with the secrecy of the exchanged contents, not of the exchange itself.

To perform steganography we need both an embedding and an extraction process. Hiding of the message is done by embedding it into the object called cover-object and the extraction of the message is done by feeding the stego-object (cover-object + secret message) and the key to the extraction algorithm.

Steganography has some points in common with digital watermarking, they are both part of the larger field - information hiding, but there are differences between the two. The main one is that steganography focuses more on the imperceptibility property of the stego object, while robustness is the most important property for digital watermarking.

1.1 Basic Terminology

In this section we explain the terms that we use in the rest of the paper, like secret message; which is the information to be hidden from the third party. Cover-object; is the carrier of the secret message and could be any digital medium (text, image, video, audio …etc.). Stego-object; is the modified cover-object after embedding the secret message. Stego-algorithm; is the procedure of embedding the secret message into the cover-object. Stego-key; it is the key used in the embedding process and it is required from the second party to provide for the extraction process to correctly recover the secret message. Steganalysis; is the art and science of detecting hidden contents. Steganalyst; is the one who applies steganalysis techniques for detecting hidden messages. False positive, is when a tool or algorithm incorrectly detects the presence of hidden contents.

1.2 Steganography in images

Almost every digital media, where there is some sort of redundancy, could be used for steganography. Multimedia objects are considered an excellent media for hiding secret messages because of numerous formats having a high degree of redundancy (Chandramouli & Memon, 2001). Moreover, using digital images as a cover-object generally provides with a large capacity and could easily go unnoticed. Image steganography could be applied in spatial and transform domains. In spatial domain data embedding is done by manipulating image's pixel values bit-by-bit, whereas in transform domain data is embedded after transforming the image to coefficients after applying a discrete cosine transform (DCT) or a discrete wavelet transform. As mentioned by (Eggers et al., 2002) the final stego image should look very similar if not identical to the cover image, and no difference should be noticed by the human eye.

1.3 Steganalysis

To easily define steganalysis we can imagine the scenario of the Simon's prisoner's problem; Alice and Bob are in jail and monitored by the warden, Wendy. Alice and Bob wanted to discuss an escape plan and they can do it only if they could make their communication hidden by using a steganographic method for hiding their secret messages. In this scenario (Kharraz et al., 2004) wrote that steganalysis can be defined as a set of methods that help Wendy to detect the existence of a secret message inside the stego-object without needing any knowledge of the secret key and, in some cases, even the algorithm of embedding process. The absence of previous knowledge makes the steganalysis process in general very complex and challenging. In this setting, Wendy can sometimes actively stop and modify any messages she feels uncomfortable with (called active warden) and in other scenarios is only supposed to pass them through between the two communicating parties (passive warden).

Similarly to cryptanalysis, we can classify steganalysis techniques into; stego only attack, when the steganalyst only has the stego-object for analysis. Known cover attack, when the steganalyst has both stego and cover objects for analysis. Known message attack, which is the case when the steganalyst knows the hidden message. Chosen stego attack, is the case when the steganalyst has both the stego-object and the embedding algorithm. Chosen message attack, is when the steganalyst uses a known message and steganography algorithm for future analysis after creating a stego-object. Finally, the

known steganography attack, the steganalyst has the cover-object, steganography algorithm, and stego-object for analysis (Kessler, 2004).

1.4 Steganalysis in digital images

Despite the difficulties in defining a *normal* or a *clean* image, it is one of the requirements of statistical based image steganalysis, in order to decide whether the image under investigation departs significantly from the *average*. To arrive to this, a number of different image characteristics are usually observed after the evaluation of many cover and stego images (Johnson & Jajodia, 1998). The idea is that the insertion of data will inevitably alter some of the image characteristics, and for spotting those the steganalyst generally checks many of them and tried to find those that are consistently and significantly changed.

So image steganalysis could be defined as applying any of the multiple steganalytic techniques on image files. Of course, there are ready-to-use steganalysis tools (software) that implement many different techniques for detecting hidden contents. We have chosen the Stegdetect as a tool for this study because it is one of the well-known freely available image steganalysis tools, detects a number of steganographic methods, and specifies the level of confidence in detection.

1.5 Stegdetect

A number of steganalysis tools (software) could be found on the web for different types of algorithms and for various digital media. In this paper we focus on Stegdetect, an automated tool developed to detect hidden contents in digital images. It can detect secret contents in images embedded with a number of different steganographic tools like; jsteg, jphide, outguess, f5, appendX, camouflage, and alpha-channel (Provos, 2008). Moreover, it also shows the level of confidence in detection by appending stars; (*), (**), (***) - one; less confidence and three; quite confidence.

Stegdetect uses statistical test for detecting hidden contents and is capable of finding the method used in the embedding process. It is a very popular tool among security and forensic practitioners, and can be considered a de facto standard, due to its excellent capabilities, and the fact that it is free and open source. There are some options that could be set during the testing phase, and in this paper we focused on the sensitivity option as it greatly affects the sensitivity of the detection algorithm. The default value is 1.0 and we explored the whole range (0.1 – 10.0) permitted by Xsteg- the GUI interface of Stegdetect. As claimed by (Cole, 2003, p. 209), the value of the sensitivity parameter should be set carefully as it affects both the false positive and false negative ratios.

Stegdetect outputs the list of all steganographic methods found in each image which could be; negative, appended alpha-channel, camouflage, false positive or others like; jphide, outguess, jsteg, and f5 with the confidence level shown by appended stars. (Provos & Honeyman, 2001) have tested stegdetect tool on two million images linked to eBay auctions and they showed that there are over 1% of the total images appeared to have hidden content, but their study did not show all the results and the details of testing process like the results we showed in the section of results. We have provided our results with all

details in simplified tables, took every result into consideration, and analysed all the results which we believe this is the first such detailed study in literature.

1.6 Digital forensics investigations

A wide range of criminal investigations use digital evidence that shows the commitment of the crime, leads to some investigation, supports witness statements or disproves it. Computer or digital forensics in its simplest definition, derived from (Carrier, 2002), refers to the science of recovering materials found in digital media to be used as a digital evidence for further investigation especially in relation to computer crimes.

Nowadays steganalysis is considered as an important and essential tool to law enforcement and media especially in cybercrime and copyright related cases (Fridrich & Goljan, 2002). However, as it hides information in a plain sight, it became a big challenge for law enforcement to detect the existence of hidden contents in digital images through visual examination (Craiger et al., 2005). There are several automated steganalysis tools, but they should be used carefully by digital forensic analyst because they are not accurate that much.

As stated by (Reith et al., 2002) the methods of obtaining reliable and analysed evidence should be well proved. So the ratio of the false positives in any tool should be known at the beginning of the investigation process, otherwise there would be a biased investigation and may end with a catastrophic result.

(Orebaugh, 2004) Have tested Stegdetect with 100 images from a digital camera and got 6% false positive in their study where all the images were clean, and all detection methods were jphide content.

2 Methodology

We have chosen Stegdetect for analysis to study the false positive ratio aiming to help digital forensics analyst who wants to make some investigation on analysing a bulk of digital images. For that purpose we have downloaded the Stegdetect0.6-4 as a debian package and installed on an Ubunto11.10 operating system on a laptop with 2.10 GHz Intel Core2 Duo processor and 3 GB of RAM. Also we have downloaded more than 40000 random image files from Google images with Multi Image Downloader (ver 1.5.8.4) and tested them with Stegdetect with different sensitivity value ranged between (0.1 – 10). In this study we have assumed that almost all downloaded images are clean due to the randomness in selection and variation of the source. Additionally, we have downloaded 25000 images from ASIRRA pet images in a compressed folder.

2.1 Finding and downloading images

We have used the most popular search engine (Google images) to collect more random images with no restrictions to a particular website. The process of searching and downloading of images were done on 9^{th}-13^{th} of February 2012 using Google's advanced image search. We started first by searching for a single English letters (a, b, c …z) and then some common keywords like (nature, people, sport, animal, computer, technology, cars, and jpg). The resulted images are downloaded by feeding the search's URL

to the Multi Image Downloader. The Multi Image Downloader downloads the image after refining the URL, adding the start parameter, and getting image links. The followings are two examples of the search URL with a single letter 'a' and turning safe search option On, Off respectively.

- http://www.google.com/search?tbm=isch&um=1&hl=en&biw=1366&bih=673&cr=&safe=images&q=a&tbs=ift:jpg
- http://www.google.com/search?tbm=isch&hl=en&biw=1366&bih=673&gbv=2&cr=&safe=off&q=a&tbs=ift:jpg

The purpose behind turning the safe search on and off with the same keywords is to get two close sets of images, this will help us to analyse the difference in detection between close groups and different ones.

After downloading all image files we started filtering out the duplicated images and some non-jpg files to make our results more robust. Additionally, we have repeated the same process, finding and downloading of images, mentioned above twice; with and without turning off the safe search option.

All other parameters stayed unchanged as shown below:

- Image attribute:
    - Image size: Any
    - Aspect ratio: Any
    - Type of image: Any
    - Source of image: Any
    - Color in image: Any
- Usage rights: All images, regardless of license labeling.
- File type: JPG files
- Region: Any region

The other group of images, ASIRRA pet images, was downloaded in a compressed folder from the link (ftp://research.microsoft.com/pub/asirra/petimages.tar) on 11th of June 2012.

3 Results

After analysing and recording the results of all (40303) random images from Google images, we have distinguished the detection results changed with sensitivity value from sensitivity independent results to do further investigations on their detection ratio. Additionally, we have noticed from the two groups of image results, enabling and disabling the safe search during the search, that there is no significant

change (for more detail see the appendix section). So we have summed up all the values from the above mentioned groups and presented as the overall result.

Sensitivity independent results; error, appended, alpha-channel, camouflage, false positive likely, jsteg, and f5 stayed unchanged during the analysis with different sensitivity values, as shown in table 1.

Table 1: The ratio of sensitivity independent results of 40303 images from Google

Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	jsteg			f5		
						(*)	(**)	(***)	(*)	(**)	(***)
0.1-10	3.16%	0.76%	0.01%	0.02%	10.76%	0.02%	0.00%	0.00%	0.00%	0.00%	0.01%

The errors are the cases where Stegdetect couldn't analyse the image because of the image format incompatibility (for example non-RGB images). The highest ratio from the sensitivity independent results goes to false positive likely, which is quite high 10.76%. Other results were low and nothing special exists to be discussed.

Sensitivity dependent results; negative, jphide, and outguess(old) were changed according to the sensitivity value, there were a change in the level of confidence as well for jphide and outguess(old) as shown in table 2.

Table 2: Sensitivity dependent results of 40303 images from Google

Sensitivity	negative	jphide			outguess(old)		
		(*)	(**)	(***)	(*)	(**)	(***)
0.1	84.80%	0.25%	0.03%	0.00%	0.14%	0.06%	0.03%
0.2	83.73%	0.87%	0.22%	0.07%	0.21%	0.08%	0.16%
0.4	82.19%	1.35%	0.56%	0.59%	0.19%	0.12%	0.33%
0.8	78.80%	3.17%	0.88%	1.63%	0.23%	0.10%	0.54%
1.0	**77.41%**	**3.80%**	**0.88%**	**2.08%**	**0.24%**	**0.13%**	**0.57%**
1.6	69.55%	9.01%	2.17%	3.52%	0.34%	0.14%	0.72%
3.2	50.52%	19.20%	6.65%	8.05%	0.21%	0.23%	0.97%
6.4	32.29%	18.63%	11.00%	22.90%	0.02%	0.02%	1.39%
10	26.90%	6.41%	17.64%	33.96%	0.01%	0.01%	1.41%

- Negative results were high (84.8%) at the beginning with low value of sensitivity parameter (0.1) and there were a gradual decrease between (0.1 – 1.0), then it decreased dramatically between (1.0 – 6.4) and went back to its normal decrease ratio afterwards. Here it means that the tool is

more critical in detecting hidden contents between (1.0 – 6.4) of the sensitivity value as shown in figure 1.

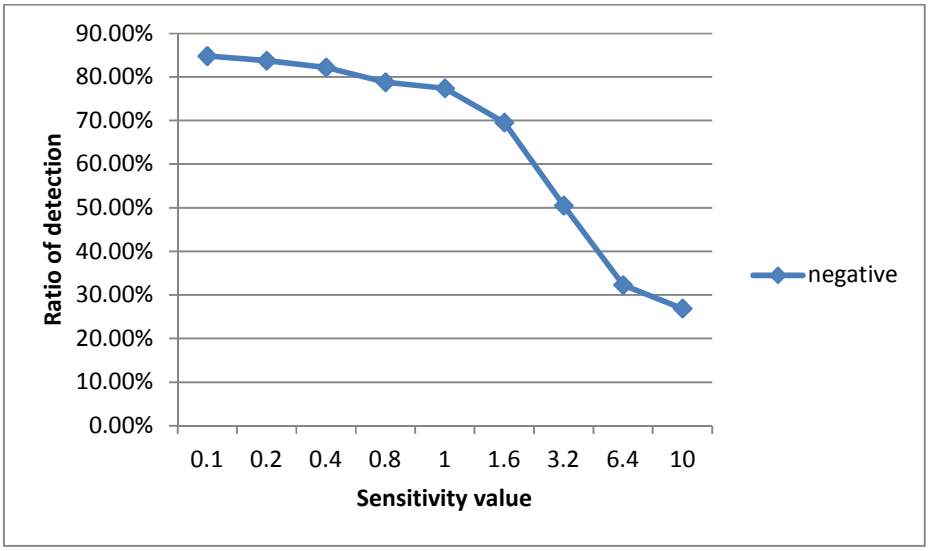


Figure 1: Changes in Negative ratio with sensitivity value

- There is a slight change in jphide results between (0.1 – 1.0) and the overall detection of jphide (*, **, ***) increased very much between (1.0 – 3.2). For jphide(**) the rate of change were stable till (10) and jphide(*) were stable between (3.2 – 6.4), then it went down afterwards while jphide(***) remained on its rapid increasing ratio as shown in figure 2.

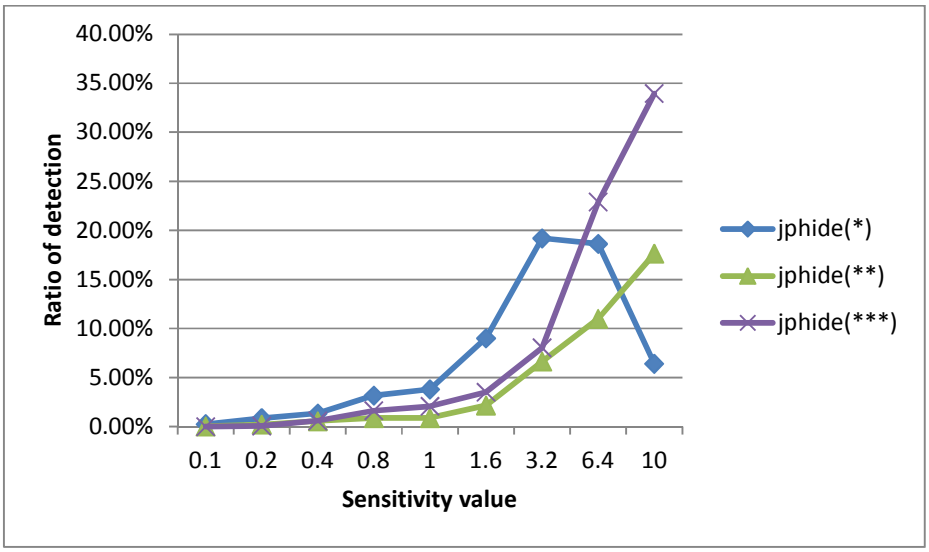


Figure 2: Changes in jphide ratio with sensitivity value

From the above graph description we can conclude that the level of confidentiality is increasing directly with the value of sensitivity and there is a great increase in overall detection confidence between the sensitivity values (3.2 – 10).

- Outguess results were different, the outguess(old)(*) increased between (0.1 – 1.6) and fallen down between (1.6 – 6.4) while outguess(old)(**) increased between (0.1 – 3.2) and then fallen down afterwards. Finally outguess(old)(***) were increased rapidly between (0.1 – 6.4) and the overall outguess(old) nearly became stable between (6.4 – 10) as shown in figure 3.

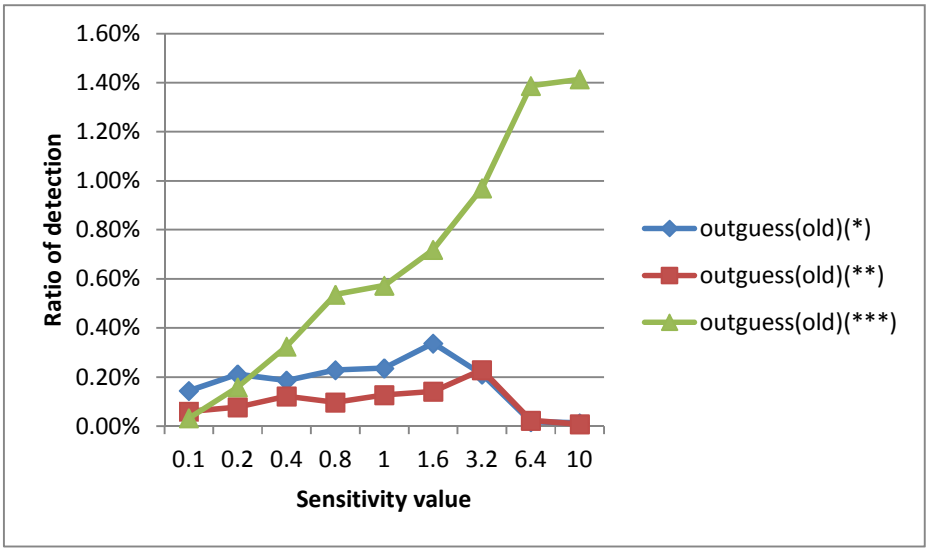


Figure 3: Changes in outguess(old) ratio with sensitivity value

The overall comment to be given on the above graph description is that the level of detection confidence is quickly increased between (0.1 – 6.4) and it nearly became stable between the sensitivity values (6.4 – 10).

- Detecting multi-methods of steganography, detection of multiple methods of steganography in the same image, was one of the interesting results in relation to the change in sensitivity value as shown in table 3.

Table 3: Examples of detecting multi-methods of steganography

Sensitivity	No. of images	Detected steganographic methods
0.1	27	Appended + false positive likely
	1	F5(***) + false positive likely
0.8	27	Appended + false positive likely
	1	F5(***) + false positive likely
	2	Jphide(*) + appended
	1	Jphide(*) + outguess(old)(***)

	1	Jphide(**) + appended
	1	Jphide(**) + outguess(old)(*)
	2	Jphide(***) + appended

The followings are some images where multi-methods of steganography are detected.

Table 4: Examples of detecting multi-methods of steganography

	Sensitivity	Detection result
	0.1	appended(575)<[nonrandom][data][……JFIF……]>
	0.2	appended(575)<[nonrandom][data][……JFIF……]>
	0.4	appended(575)<[nonrandom][data][……JFIF……]>
	0.8	appended(575)<[nonrandom][data][……JFIF……]>
	1.0	appended(575)<[nonrandom][data][……JFIF……]>
	1.6	outguess(old)(*) appended(575)<[nonrandom][data][……JFIF……]>
	3.2	outguess(old)(**) appended(575)<[nonrandom][data][……JFIF……]>
	6.4	outguess(old)(***) jphide(*) appended(575)<[nonrandom][data][……JFIF……]>
	10	outguess(old)(***) jphide(**) appended(575)<[nonrandom][data][……JFIF……]>
	Sensitivity	Detection result
	0.1	negative
	0.2	negative
	0.4	negative
	0.8	negative
	1.0	negative
	1.6	negative
	3.2	outguess(old)(*) jphide(*)
	6.4	outguess(old)(***) jphide(**)
	10	outguess(old)(***) jphide(***)
	Sensitivity	Detection result
	0.1	negative
	0.2	negative
	0.4	outguess(old)(*)
	0.8	outguess(old)(***) jphide(*)
	1.0	outguess(old)(***) jphide(*)
	1.6	outguess(old)(***) jphide(**)
	3.2	outguess(old)(***) jphide(***)

	6.4	outguess(old)(***) jphide(***)
	10	outguess(old)(***) jphide(***)

To simplify the results of detecting multi-methods of steganography, we only show the relation between the sensitivity value and the ratio of detecting multi-methods of steganography in the following graph:

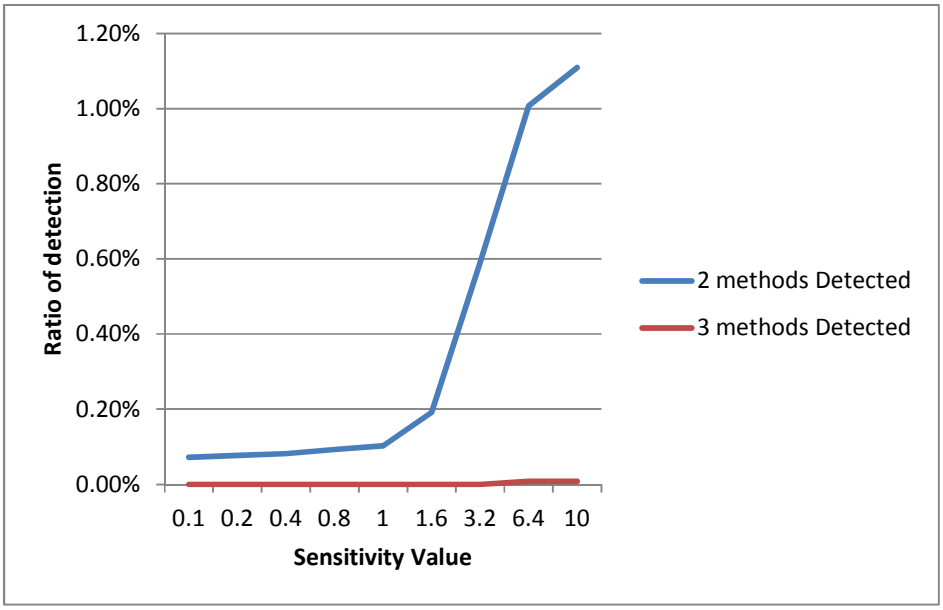


Figure 4: The detection ratio of multi-methods of steganography

It is noticeable that the sensitivity value directly affects the detection of multi-methods of steganography especially two-methods of steganography for sensitivity values (1.6 – 6.4).

- Of course considering all downloaded images as clean is not very accurate, at least for probability of having watermarked images. Still we have quite high overall false positives, after excluding 'errors' and the 'false positives' considered by the tool itself, especially between the sensitivity values of (1.0 – 10). Moreover, the highest ratio of false positive comes from jphide with different levels of confidence. However, the overall false positive, in the worst case (sensitivity = 10.0), excluding the jphide will reach 2.25% which is much lower than jphide only ratio (58.01%). This result would be quite useful for digital forensics analyst in examining a bulk of images and they should take this high ratio of false positives into account for further investigation. Figure 5 will clarify the overall picture of the false positive ratio:

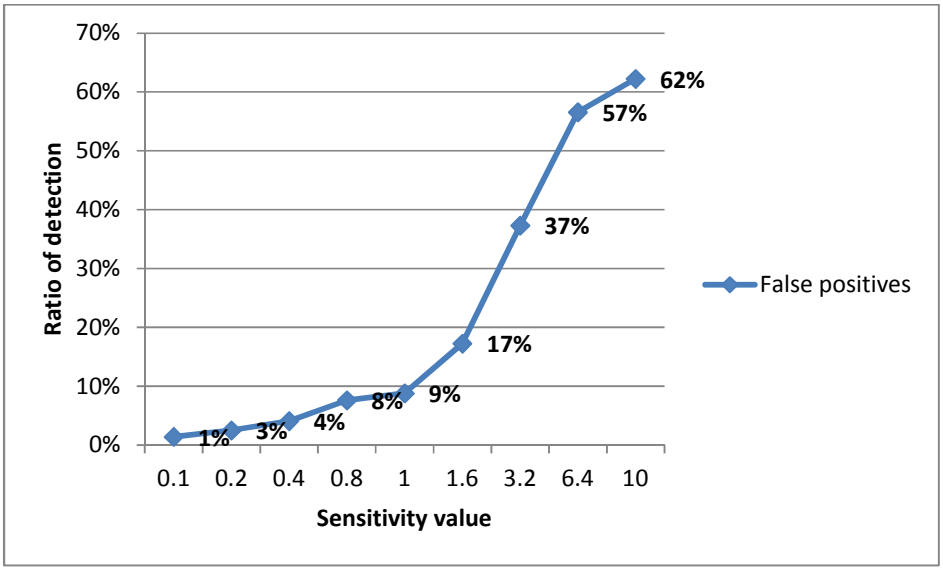


Figure 5: The overall false positive ratio

For the other group of images, ASIRRA pet images (Cat and Dog), again the error, appended, alpha-channel, camouflage, false positive likely, jsteg, and f5 stayed unchanged during the analysis with different sensitivity values, as shown in table 5.

Table 5: The ratio of sensitivity independent results of 25000 images from ASIRRA pets

Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	jsteg			f5		
						(*)	(**)	(***)	(*)	(**)	(***)
0.1-10	0.96%	0.08%	0.35%	0.00%	3.50%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

Again the highest ratio from the sensitivity independent results goes to false positive likely, which is 3.5%. Other results were low and nothing special exists to be discussed.

Also the ratio of negative, jphide, and outguess(old) were changed according to the sensitivity value and there were a change in the level of confidentiality as well for jphide and outguess(old) as shown in table 6.

Table 6: Sensitivity dependent results of 25000 images from ASIRRA pets

Sensitivity	negative	jphide			outguess(old)		
		(*)	(**)	(***)	(*)	(**)	(***)

0.1	94.26%	0.54%	0.04%	0.01%	0.16%	0.04%	0.04%
0.2	91.42%	2.70%	0.44%	0.16%	0.16%	0.11%	0.14%
0.4	88.20%	3.13%	1.97%	1.34%	0.11%	0.09%	0.32%
0.8	85.46%	2.61%	1.59%	4.85%	0.16%	0.06%	0.46%
1.0	**83.72%**	**4.91%**	**1.29%**	**5.75%**	**0.14%**	**0.10%**	**0.50%**
1.6	70.86%	14.58%	1.81%	7.23%	0.12%	0.07%	0.61%
3.2	37.45%	33.57%	11.35%	12.27%	0.06%	0.05%	0.75%
6.4	21.67%	15.97%	17.76%	39.44%	0.02%	0.00%	0.86%
10	15.08%	7.51%	15.06%	57.30%	0.01%	0.01%	0.86%

- The graphs of the sensitivity dependent results were very similar to the ones we got from Google images in both shape and rate of change perspectives however there is a difference between ratio of detections, that is why we didn't describe them again in detail, but the graphs are still shown below:

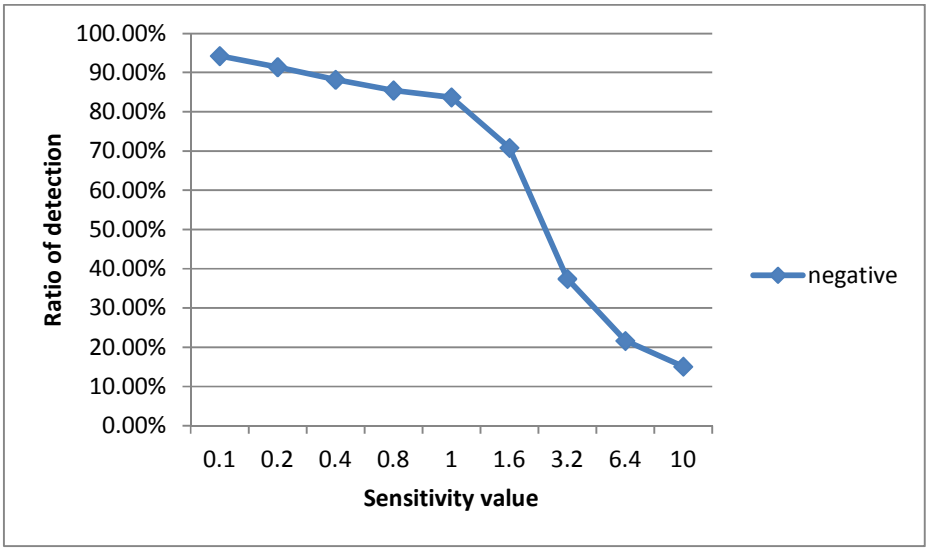


Figure 6: Changes in Negative ratio with sensitivity value

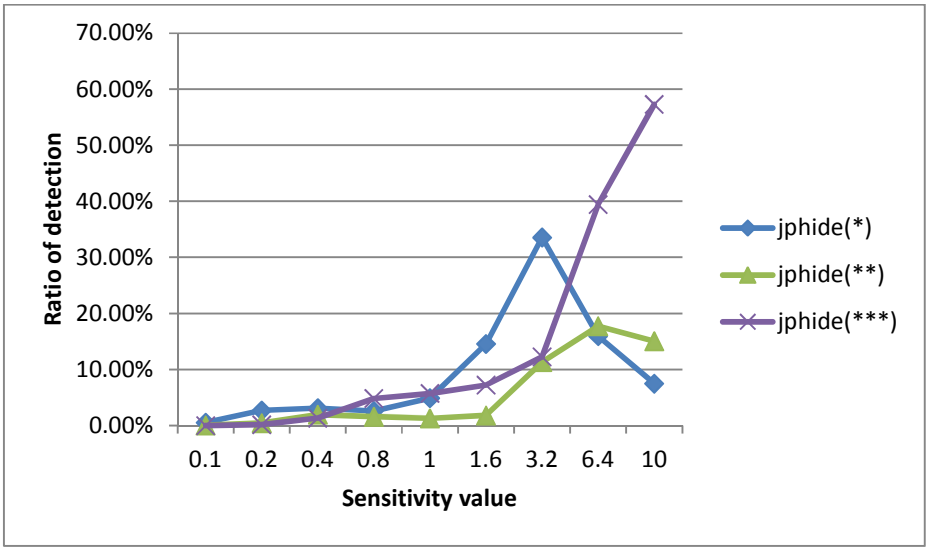


Figure 7: Changes in jphide ratio with sensitivity value

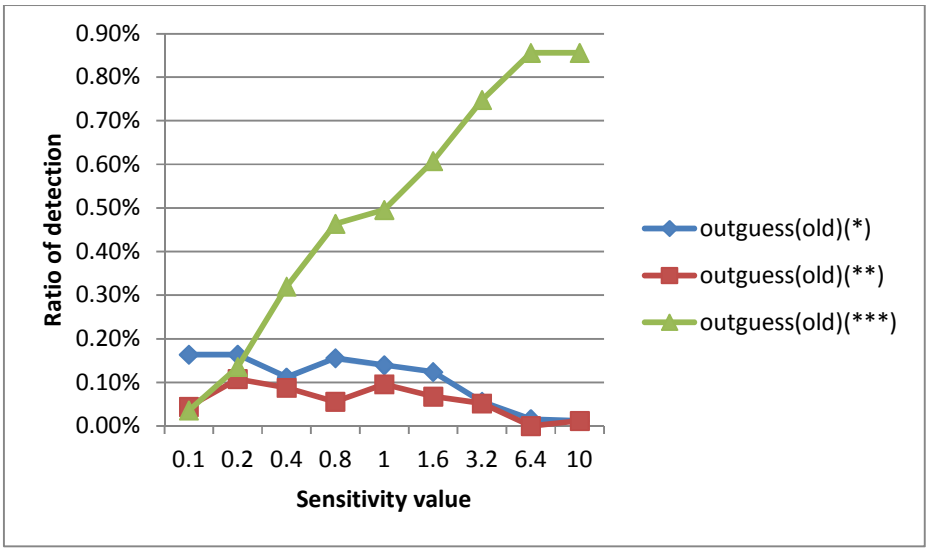


Figure 8: Changes in outguess(old) ratio with sensitivity value

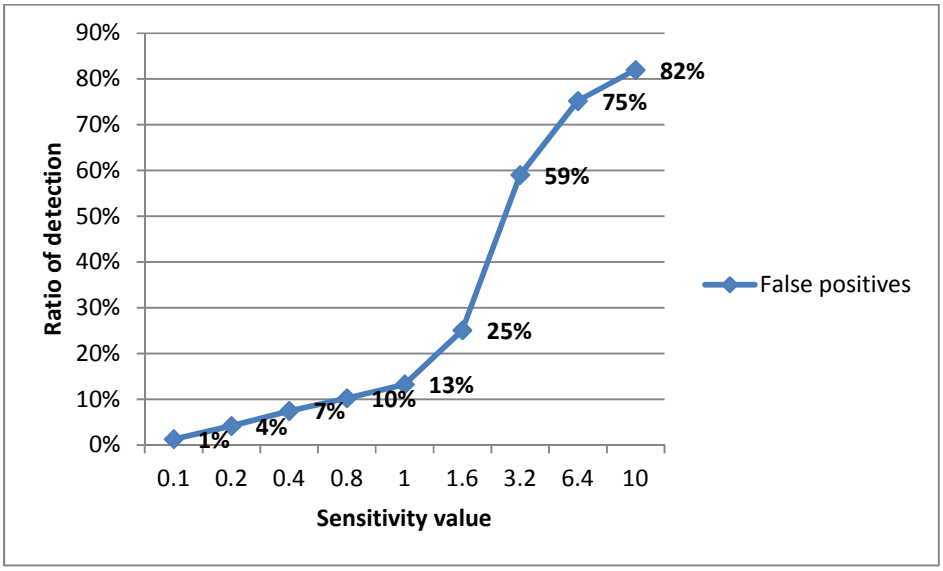


Figure 9: The overall false positive ratio

3.1 Statistical analysis

Assessing or evaluating the accuracy of steganalysis tools and the reliability of their results are not easy, especially for digital forensics analyst. Doing such kind of work needs a good knowledge in steganalysis methods, which is not very interesting to the forensics analyst, as they use the steganalysis tools as a black box; they give it inputs and get results back from it. So providing this method, statistical analysis, would be a simple and useful tool in doing that.

To study the difference between results we have got so far we have used a statistical method, two-proportion z-test, to test our hypothesis (the two samples are identical or not). We set the null hypothesis H_0; as there is no difference between the two results proportion and the alternative hypothesis H_a; as there is a difference.

H_0: $p_1=p_2$

H_a: $p_1 \neq p_2$

We set the significant level to 0.05; in this case the error rate of 5% is accepted. Here we compute p-value (the probability associated with the z-score) and compare it with the significant level. If the p-value was less than the significant level, we reject the null hypothesis i.e. there is a difference between the proportions of detection results, otherwise they would be equal.

According to the resulted p-value we can notice the significance of the difference in detection proportions like the following:

Significant: p-value <0.05

Non-Significant: p-value ≥ 0.05

We have done a statistical test for two sets of images and showed the result in tables similar to the ones used in showing the Stegdetect results. Here we coloured the non-significant p-values with green and the significant ones with red. There are some cells with not applicable (N/A), resulted from having the value of zero from both results (Off and On), which is also coloured with green as there is no significant difference.

The two groups of images from Google with Safe search option (Off and On) were taken for the test and got only 0.617% (1/162) of red cells, which is less than 5%, as shown in table 7.

Table 7: The difference of detection between Safe search (Off and On) images

Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
							(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
0.1	0.725	0.090	0.678	0.740	0.745	0.773	0.750	0.458	N/A	0.797	0.798	0.572	0.798	0.482	0.486	N/A	N/A	0.798
0.2	0.725	0.090	0.678	0.740	0.745	0.662	0.248	0.767	0.530	0.789	0.788	0.780	0.798	0.482	0.486	N/A	N/A	0.798
0.4	0.725	0.090	0.678	0.740	0.745	0.773	0.542	0.452	0.345	0.660	0.786	0.780	0.798	0.482	0.486	N/A	N/A	0.798
0.8	0.725	0.090	0.678	0.740	0.745	0.710	0.219	0.590	0.396	0.773	0.784	0.782	0.798	0.482	0.486	N/A	N/A	0.798
1	**0.725**	**0.090**	**0.678**	**0.740**	**0.745**	**0.654**	**0.417**	**0.002**	**0.506**	**0.750**	**0.786**	**0.770**	**0.798**	**0.482**	**0.486**	**N/A**	**N/A**	**0.798**
1.6	0.725	0.090	0.678	0.740	0.745	0.388	0.289	0.392	0.787	0.576	0.626	0.796	0.798	0.482	0.486	N/A	N/A	0.798
3.2	0.725	0.090	0.678	0.740	0.745	0.191	0.497	0.354	0.443	0.746	0.483	0.753	0.798	0.482	0.486	N/A	N/A	0.798
6.4	0.725	0.090	0.678	0.740	0.745	0.105	0.764	0.517	0.195	0.740	0.751	0.798	0.798	0.482	0.486	N/A	N/A	0.798
10	0.725	0.090	0.678	0.740	0.745	0.107	0.759	0.730	0.093	0.719	0.672	0.798	0.798	0.482	0.486	N/A	N/A	0.798

It shows that the two groups had got similar detection proportions and no significant difference has been found. It shows the acceptance of the null hypothesis ($p_1=p_2$), by this the digital forensic analyst shouldn't be worry about these two groups of images.

For further investigation we have taken the ASIRRA pet images and tested the Cat and Dog images, we got 20.37% (33/162) of red cells that rejects the null hypothesis ($p_1 \neq p_2$). The red cells are resulted from error, negative, and jphide as shown in table 8.

Table 8: The difference of detection between ASIRRA (Cat and Dog) images

Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
							(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
0.1	0.016	0.161	0.691	0.484	0.535	0.017	0.002	0.086	0.294	0.788	0.762	0.484	0.484	0.484	N/A	N/A	N/A	N/A
0.2	0.016	0.161	0.691	0.484	0.535	0.001	0.052	0.029	0.005	0.788	0.783	0.630	0.484	0.484	N/A	N/A	N/A	N/A
0.4	0.016	0.161	0.691	0.484	0.535	0.007	0.796	0.067	0.005	0.743	0.352	0.323	0.484	0.484	N/A	N/A	N/A	N/A
0.8	0.016	0.161	0.691	0.484	0.535	0.181	0.011	0.138	0.000	0.091	0.798	0.605	0.484	0.484	N/A	N/A	N/A	N/A
1	0.016	0.161	0.691	0.484	0.535	0.765	0.001	0.177	0.001	0.787	0.572	0.690	0.484	0.484	N/A	N/A	N/A	N/A
1.6	0.016	0.161	0.691	0.484	0.535	0.000	0.000	0.001	0.022	0.690	0.612	0.787	0.484	0.484	N/A	N/A	N/A	N/A
3.2	0.016	0.161	0.691	0.484	0.535	0.000	0.000	0.000	0.084	0.450	0.768	0.779	0.484	0.484	N/A	N/A	N/A	N/A
6.4	0.016	0.161	0.691	0.484	0.535	0.000	0.000	0.000	0.254	0.484	N/A	0.733	0.484	0.484	N/A	N/A	N/A	N/A
10	0.016	0.161	0.691	0.484	0.535	0.548	0.000	0.001	0.000	0.675	0.675	0.733	0.484	0.484	N/A	N/A	N/A	N/A

It helps the digital forensics analyst to indicate the area of difference for further investigation process. Here error, negative, and jphide may be considered for further study by the digital forensics analyst. A certain image processing and filtering process may have been applied before publishing the ASIRRA pet images, which also should be considered by the digital forensics analyst.

4 Conclusion

In this study we have analysed one of the well-known digital image steganalysis tools (Stegdetect) to examine the false positive ratio. This could greatly benefit the digital forensic analyst in their investigation. We conclude that the value of the sensitivity parameter strongly affects the detection rate for jphide and outguess(old), especially when the sensitivity value is between (1.0 – 6.4). Another conclusion, possibly the most important one, is that we have noticed a high ratio of false positives particularly between sensitivity values of (1.0 – 10). For that reason we can indicate the sensitivity value of 1.0 as an optimum value for detection, as the detection of 'negative' is sharply fall down after this point. This high ratio of false positive should be taken into consideration by the digital forensic analyst when they analysing, as is frequently the case, a large number of images during an investigation using Stegdetect. Finally, we have proposed a statistical tool to show the difference in proportion of detection between two groups of images. The most random group of images could act as a baseline for this comparison, the Google images in our case. This would help the digital forensic analyst to take further informed decisions during an investigation process, likely arriving at better conclusions. This statistical method could be applied to any other steganalysis tools, especially when the digital forensics analyst has no prior information about the false positive ratio of the chosen tool.

There are two other related studies we intend to achieve as a future works: one is based on studying the false negative ratio of Stegdetect, the other on doing similar analysis for other steganalysis tools.

References

Carrier, B. Defining Digital Forensic Examination and Analysis Tools. Digital Forensics Research Workshop II 2002.

Chandramouli, R., & Memon, N. Analysis of LSB based Image Steganography Techniques. Proceedings of ICIP 2001. Greece: Thessaloniki.

Chandramouli, R., & Memon, N. Steganography Capacity: A Steganalysis Perspective. SPIE Security and Watermarking of Multimedia Contents V 2003; 5020.

Cole, E. Hiding in Plain Sight: Steganography and the Art of Covert Communication. Indianapolis: Wiley Publishing, Inc.; 2003.

Craiger, P. J., Pollitt, M., & Swauger, J. Law Enforcement and Digital Evidence. In To appear in H. Bidgoli (Ed.), Handbook of Information Security (pp. 17-18). New York: John Wiley & Sons; 2005.

Eggers, J., Bäuml, R., & Girod, B. A Communications Approach to Steganography. Proceedings of SPIE 2002; 4675: 26-49.

Fridrich, J., & Goljan, M. Practical Steganalysis of Digital Images – State of the Art. Proceedings of the SPIE Security and Watermarking of Multimedia Contents IV 2002; 4675: 1-13.

Johnson, N. F., & Jajodia, S. Steganalysis of Images Created Using Current Steganography Software. in Information Hiding: 2nd Int. Workshop 1998; 1525: 273–289.

Kessler, G. C. An Overview of Steganography for the Computer Forensics Examiner. Frorensic Science Communications 2004; 6(3).

Orebaugh, A. D. Steganalysis: A Steganography Intrusion Detection System; 2004. Retrieved June 4, 2012, from http://securityknox.com/Steg_project.pdf

Provos, N. OutGuess - Steganography Detection; 2008. Retrieved May 2012, from OutGuess: http://www.outguess.org/detection.php

Provos, N., & Honeyman, P. Detecting Steganographic Content on the Internet. CITI Technical Report 2001: 01-11.

Reith, M., Reith, C., & Gunsch, G. An Examination of Digital Forensic Models. International Journal of Digital Evidence 2002; 1(3).

Appendices

A. The followings tables are the raw results of detection for each group of images:

Table A.1: The detection results of Safe search option (On)

No. of Images	Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
								(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
20063	0.1	626	170	1	4	2148	17023	49	5	0	29	12	5	4	1	0	0	0	3
20063	0.2	626	170	1	4	2148	16821	160	42	12	43	15	31	4	1	0	0	0	3
20063	0.4	626	170	1	4	2148	16500	282	105	109	40	25	64	4	1	0	0	0	3
20063	0.8	626	170	1	4	2148	15790	665	184	312	47	20	109	4	1	0	0	0	3
20063	1	626	170	1	4	2148	15504	785	144	403	49	26	117	4	1	0	0	0	3
20063	1.6	626	170	1	4	2148	13898	1849	452	709	63	31	145	4	1	0	0	0	3
20063	3.2	626	170	1	4	2148	10051	3891	1366	1644	44	41	198	4	1	0	0	0	3
20063	6.4	626	170	1	4	2148	6384	3749	2236	4665	4	5	278	4	1	0	0	0	3
20063	10	626	170	1	4	2148	5308	1279	3555	6912	3	2	284	4	1	0	0	0	3

Table A.2: The detection results of Safe search option (Off)

No. of Images	Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
								(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
20240	0.1	647	135	2	3	2190	17155	53	9	0	29	12	8	4	0	1	0	0	3
20240	0.2	647	135	2	3	2190	16924	190	45	17	42	16	33	4	0	1	0	0	3
20240	0.4	647	135	2	3	2190	16626	264	122	130	35	24	67	4	0	1	0	0	3
20240	0.8	647	135	2	3	2190	15969	614	171	345	45	19	107	4	0	1	0	0	3
20240	1	647	135	2	3	2190	15694	748	210	434	46	25	114	4	0	1	0	0	3
20240	1.6	647	135	2	3	2190	14132	1783	421	709	73	26	145	4	0	1	0	0	3
20240	3.2	647	135	2	3	2190	10310	3848	1314	1599	41	51	193	4	0	1	0	0	3
20240	6.4	647	135	2	3	2190	6630	3759	2197	4564	3	4	281	4	0	1	0	0	3
20240	10	647	135	2	3	2190	5534	1306	3554	6775	2	1	286	4	0	1	0	0	3

Table A.3: The detection results of ASIRRA pet images (Cat)

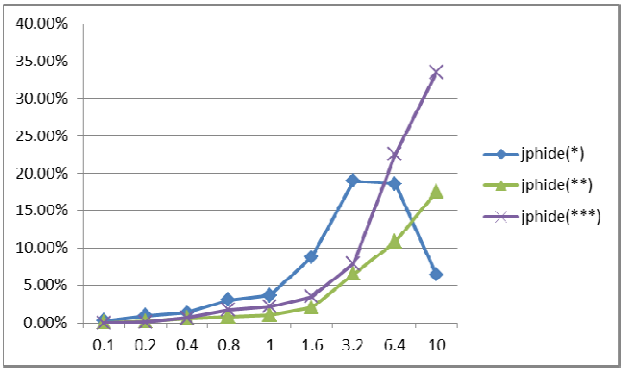
No. of Images	Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
								(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
12500	0.1	98	14	46	0	425	11834	48	2	0	21	6	6	0	0	0	0	0	0
12500	0.2	98	14	46	0	425	11507	308	42	10	21	14	19	0	0	0	0	0	0
12500	0.4	98	14	46	0	425	11103	390	222	138	15	8	46	0	0	0	0	0	0
12500	0.8	98	14	46	0	425	10731	363	217	533	13	7	62	0	0	0	0	0	0
12500	1	98	14	46	0	425	10457	552	177	651	17	10	65	0	0	0	0	0	0
12500	1.6	98	14	46	0	425	8698	2030	264	849	17	7	75	0	0	0	0	0	0
12500	3.2	98	14	46	0	425	4942	3777	1554	1589	5	7	92	0	0	0	0	0	0
12500	6.4	98	14	46	0	425	2854	2110	1932	4988	3	0	104	0	0	0	0	0	0
12500	10	98	14	46	0	425	1909	1073	1986	6929	2	2	104	0	0	0	0	0	0

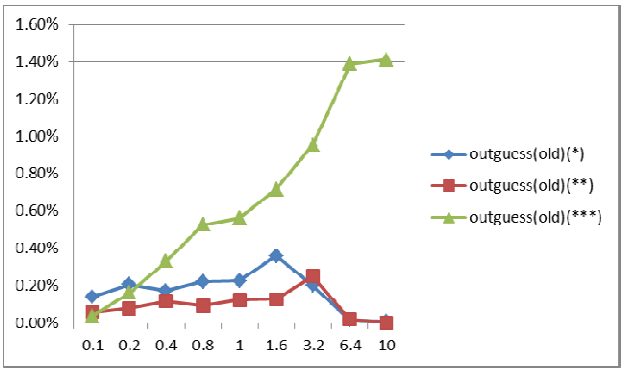
Table A.4: The detection results of ASIRRA pet images (Dog)

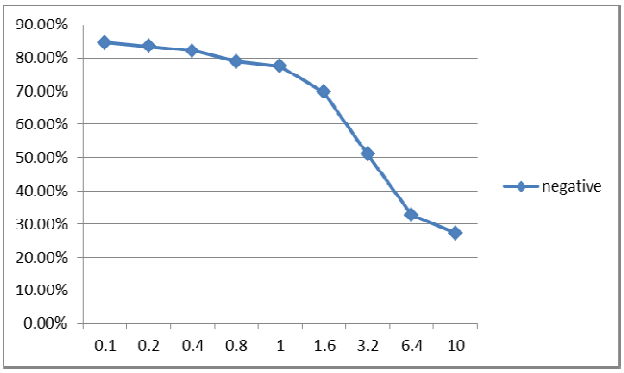
No. of Images	Sensitivity	error	appended	Alpha-channel	camouflage	skipped (false positive likely)	negative	jphide			outguess(old)			jsteg			f5		
								(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)	(*)	(**)	(***)
12500	0.1	141	6	41	1	451	11732	88	9	2	20	5	3	1	1	0	0	0	0
12500	0.2	141	6	41	1	451	11347	368	69	30	20	13	15	1	1	0	0	0	0
12500	0.4	141	6	41	1	451	10946	392	271	196	13	14	34	1	1	0	0	0	0
12500	0.8	141	6	41	1	451	10635	289	180	679	26	7	54	1	1	0	0	0	0
12500	1	141	6	41	1	451	10474	675	146	787	18	14	59	1	1	0	0	0	0
12500	1.6	141	6	41	1	451	9016	1615	189	959	14	10	77	1	1	0	0	0	0
12500	3.2	141	6	41	1	451	4421	4615	1284	1479	9	6	95	1	1	0	0	0	0
12500	6.4	141	6	41	1	451	2563	1882	2507	4871	1	0	110	1	1	0	0	0	0
12500	10	141	6	41	1	451	1860	804	1778	7397	1	1	110	1	1	0	0	0	0

B. The followings graphs are the graphs of detection ratio for each separate group of images:

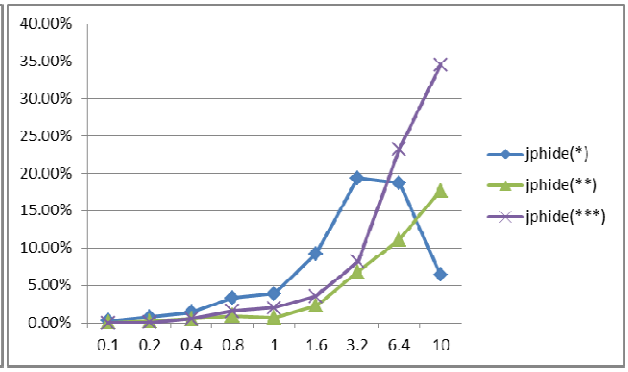
Safe search option Off

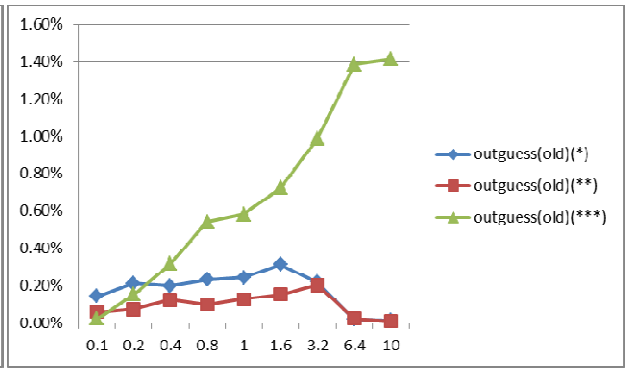


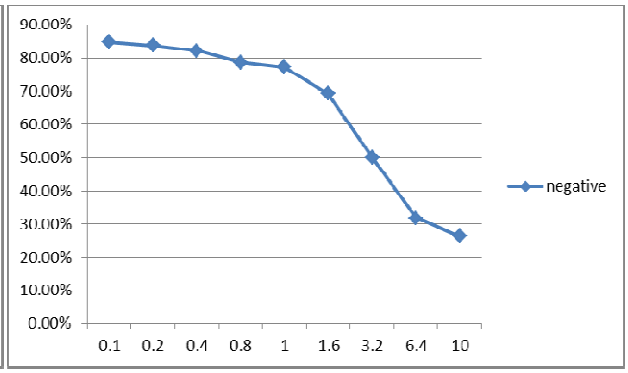




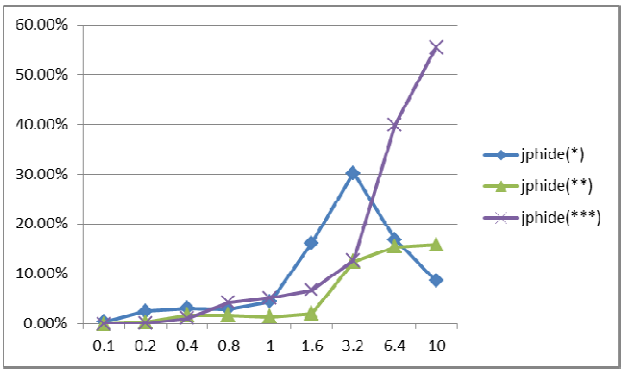
Safe search option On

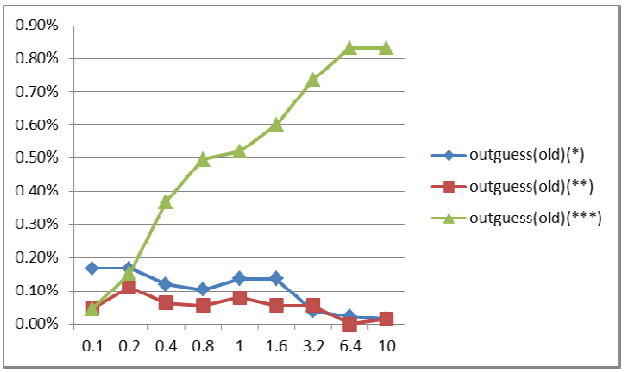


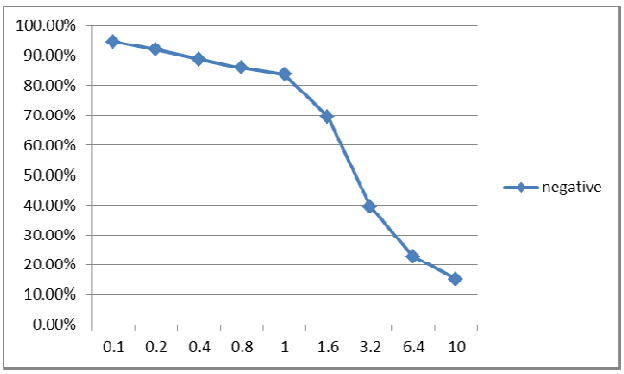




ASIRRA Cat images







ASIRRA Dog images