A New Cyber Forensic Philosophy for Digital Watermarks in the Context of Copyright Laws

Vinod P. Bhattathiripad
Cyber Forensic Consultant, GJ Software Forensics, vinodpolpaya@gmail.com

Sneha Sudhakaran
Cyber Forensic Consultant, GJ Software Forensics, sneha14888@gmail.com

Roshna K. Thalayaniyil
College of Engineering, Kallooppara,Kerala India, roshnakhalidt@gmail.com

A NEW CYBER FORENSIC PHILOSOPHY FOR DIGITAL WATERMARKS IN THE CONTEXT OF COPYRIGHT LAWS

Vinod Polpaya Bhattathiripad Ph D
Cyber Forensic Consultant
GJ Software Forensics
Kozhikode - 673004, Kerala, India
vinodpolpaya@gmail.com

Sneha Sudhakaran
Cyber Forensic Consultant
GJ Software Forensics
Kozhikode - 673004, Kerala, India
sneha14888@gmail.com

Roshna Khalid Thalayaniyil
College of Engineering
Kallooppara,Kerala
India
roshnakhalidt@gmail.com

ABSTRACT

The objective of this paper is to propose a new cyber forensic philosophy for watermark in the context of copyright laws for the benefit of the forensic community and the judiciary worldwide. The paper first briefly introduces various types of watermarks, and then situates watermarks in the context of the idea-expression dichotomy and the copyright laws. It then explains the forensic importance of watermarks and proposes a forensic philosophy for them in the context of copyright laws. Finally, the paper stresses the vital need to incorporate watermarks in the forensic tests to establish software copyright infringement and also urges the judiciary systems worldwide to study and legalize the evidential aspects of digital watermarks in the context of copyright laws.

Keywords: Digital Watermarks, Software Copyright, Idea-Expression Dichotomy, Programming Blunders, Copyright Infringement, AFC, POSAR

1. INTRODUCTION

Software can be copyright protected. When an infringement of the copyright is suspected, the copyright owner has every moral and legal right to ensure the exclusivity of their property rights to the software. It is only natural that when such rights have been flagrantly violated, particularly for commercial profits (and uses), the injured parties will invariably resort to legal measures both for the protection of their property and for the restitution of damages involved therein. Such an issue can trigger a legal battle.

In the process of legally establishing copyright infringement, the watermark (contained in the software) can play an important role. In order to use watermark as an evidence to establish the criminal activity behind the infringement allegation, both the forensic procedure (used as part of the investigation) and the judge's decision-making process need to be sensitive to the forensic role of watermarks.

Although much has been done on the design, programming and implementation aspects of watermarks (Cox et al, 2008), there has not been any effort from cyber forensic researchers to explain the forensic locus standi and philosophical rationalle of watermarks for the benefit of the entire forensic community and also for the benefit of the judiciary across the world. As a result of this deficiency, a cognitive (or an expertise) gap can exist between the forensic community and the judiciary and the goal of this work is to fill this gap. As several different forms of digital watermarks exist, it is the duty of forensic professionals to explain the forensic roles of various different watermarks separately and then generalize these different roles to form a single forensic philosophy which can be ultimately used by the judiciary for effective decision making in any software copyright infringement litigation.

Before getting into the forensic philosophy of watermarks, a quick overview of digital watermarks will help readers to situate this work properly.

2. OVERVIEW OF WATERMARKS

File watermarking is not uncommon in the digital world. It is a widely used mechanism worldwide in order to protect the ownership of a digital file,

including software. A digital watermark (or, simply a watermark) in a digital file (whether it is a text or image or an audio or a video file) is some kind of electronic thumb impression introduced by the owner into the file for easy establishment of his / her creativity (Nagra et al, 2002).

Since any digital file has a source code (or a hex dump) as part of it (see fig 1), file watermarking virtually becomes a process of embedding some kind of information into the source code (of the file) for the purpose of introducing some degree of personalization (or identity) into the source code (Cox et al, 2008). When a watermark is embedded into any digital file, the source code of the watermark also gets embedded into the source code of the digital file (see fig 2 & 3).

Watermarks can exist in different forms like text, image, audio and video (and also combinations of these forms). The best way to further explain a watermark is to quickly demonstrate the technicalities of an image file, first using its non-watermarked form and then, its watermarked form.

There is a general feeling that a watermark is always a single, identifiable and easily separable entity in a watermarked file and that a watermarked file always differs from its non-watermarked form by only a few hexdumps. This is not true. Most watermarks do not remain as single, identifiable and easily separable entities in the watermarked file. Also, the hexdump of any non-watermarked image (for example, see fig 2) differs in a big way from that of the watermarked form of the same image (see fig 3) and this difference can be easily verified by comparing the corresponding hex values in fig 2 and fig 3. This big difference is because the watermarking algorithm not only inserts the hex values of watermarks into the original (non-watermarked) image but also modifies most hex values of the original image. In the same manner, the hexdumps of any particular non-watermarked audio, video or a text file also differ largely in the same fashion from those of the watermarked form of the file.

Just as there are different forms of digital files (image, audio, video and text form and also their different combinations), watermarks can also exist in many forms. Further, watermarks can be classified in many different ways based on several factors. An overview of two sample classifications will help readers to situate the forensic aspects of watermarks properly. Based on their techniques of generation, watermarks are classified into two types and they are static watermarks (which are embedded as code segments within the source code of a digital file) and dynamic watermarks (which are watermarks generated during the runtime with the help of code segments embedded within the source code of a digital file) (collberg and Thomborson, 1999). Again, based on the roles played by different persons involved in the development of the software, watermarks can be classified as authorship mark, fingerprinting mark, validation mark and licensing mark (which are unique identities of the author, distributor, publisher and consumer, respectively, of the software that contains the watermark) (Nagra et al, 2002).

Every watermark has certain desirable features like effectiveness (or the correctness and aptness of the intended purpose of the watermark), integrity (or the watermark's ability to not to interfere with the performance of the source code), fidelity (or how closely the watermark accurately or truthfully helps to identify the 'owner' of the software), robustness (or the watermark's ability to withstand any kind of alteration of the content of the file in which the watermark is embedded) etc. (Nagra et al, 2002; Cox et al, 2008; Marcella and Menendez, 2008).

3.THE IDEA-EXPRESSION DICHOTOMY AND WATERMARKS

The idea-expression dichotomy (Walker, 1996) provides an excellent theoretical perspective to look at and further delineate watermarks embedded as part of the source code of any software. Any software is (or consists of) a collection of code segments and each code segment is an expression of one or more ideas. This being so, software, as a whole, can be considered a collection of expressions of one or more ideas.

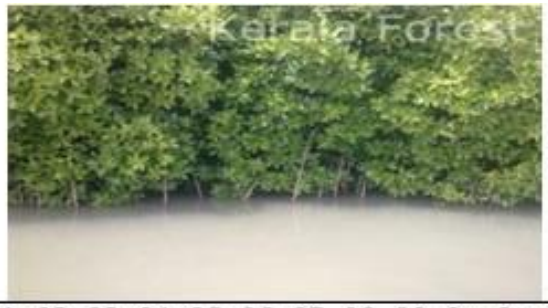
```
/* E:\images for paper watermarks\03072011099.jpg (7/3/2011 7:12:44 AM)
  StartOffset: 00000000, EndOffset: 00173627, Length: 00173628 */
unsigned char rawData[1521192] = {
        0xFF, 0xD8, 0xFF, 0xE1, 0x6C, 0xCB, 0x45, 0x78, 0x69, 0x66, 0x00, 0x00,
        0x49, 0x49, 0x2A, 0x00, 0x08, 0x00, 0x00, 0x00, 0x08, 0x00, 0x0F, 0x01,
        0x02, 0x00, 0x06, 0x00, 0x00, 0x00, 0x6E, 0x00, 0x00, 0x00, 0x10, 0x01,
        0x02, 0x00, 0x06, 0x00, 0x00, 0x00, 0x74, 0x00, 0x00, 0x00, 0x12, 0x01,
        0x03, 0x00, 0x01, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x1A, 0x01,


        0x04, 0x76, 0x14, 0x8E, 0x3E, 0x70, 0x48, 0x57, 0x69, 0x06, 0x0B, 0x18,
        0xF1, 0xF8, 0xFD, 0x2A, 0x2D, 0xE4, 0x60, 0x71, 0x90, 0x07, 0xDD, 0x4D,
        0xC3, 0xAF, 0x3D, 0x29, 0x1C, 0x9F, 0x25, 0x99, 0x15, 0xCC, 0x9B, 0xF6,
        0x85, 0x46, 0xC7, 0x19, 0xE2, 0x9F, 0x23, 0x23, 0x64, 0x7F, 0xFF, 0xD9
};
```

Figure 1. A JPEG file and its source code in C, generated using the HxD tool. (Only the beginning and the end of the C code are shown here and the hidden portion is indicated by a thick white space) (Picture courtesy: Kadalundi Mangrove Reserve preserved by Kerala Forests, Kozhikode district, India)

offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000000	FF	D8	FF	E1	6C	CB	45	78	69	66	00	00	49	49	2A	00	ÿØÿálËExif..II*.
00000010	08	00	00	00	08	00	0F	01	02	00	06	00	00	00	6E	00	.............n.
00000020	00	00	10	01	02	00	06	00	00	00	74	00	00	00	12	01	..........t.....
00000030	03	00	01	00	00	00	01	00	00	00	1A	01	05	00	01	00	....
00000040	00	00	7A	00	00	00	1B	01	05	00	01	00	00	00	82	00	..z.............,.
00000050	00	00	28	01	03	00	01	00	00	00	02	00	00	00	13	02	..(..............
00000060	03	00	01	00	00	00	01	00	00	00	69	87	04	00	01	00	..........i‡....
00000070	00	00	8A	00	00	00	44	32	00	00	4E	6F	6B	69	61	00	..Š...D2..Nokia.
00000080	45	37	32	2D	31	00	2C	01	00	00	01	00	00	00	2C	01	E72-1.,.......,.
00000090	00	00	01	00	00	00	1A	00	9A	82	05	00	01	00	00	00	........š,......
000000A0	C8	01	00	00	9D	82	05	00	01	00	00	00	D0	01	00	00	È...,.....Ð...
000000B0	27	88	03	00	01	00	00	00	64	00	00	00	00	00	90	07	00	'^.....d.....
000000C0	04	00	00	00	30	32	32	30	03	90	02	00	14	00	00	00	....0220........
000000D0	D8	01	00	00	04	90	02	00	14	00	00	00	EC	01	00	00	Ø.............ì...
000000E0	01	91	07	00	04	00	00	00	01	02	03	00	01	92	0A	00	.'.............'..
001735F0	B1	6B	5D	CB	0B	8E	17	03	04	76	14	8E	3E	70	48	57	±k]Ë.Ž...v.Ž>pHW
00173600	69	06	0B	18	F1	F8	FD	2A	2D	E4	60	71	90	07	DD	4D	i...ñøý*-ä`q..ÝM
00173610	C3	AF	3D	29	1C	9F	25	99	15	CC	9B	F6	85	46	C7	19	Ã¯=).Ÿ%™.Ì›ö…FÇ.
00173620	E2	9F	23	23	64	7F	FF	D9									âŸ##d.ÿÙ

Figure 2. The hexdump (generated using the HxD tool) of the non-watermarked JPEG image shown in Figure.1



offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000000	FF	D8	FF	E0	00	10	4A	46	49	46	00	01	01	01	01	2C	ÿØÿà..JFIF.....,
00000010	01	2C	00	00	FF	E1	32	96	45	78	69	66	00	00	4D	4D	.,..ÿá2–Exif..MM
00000020	00	2A	00	00	00	08	00	09	01	0F	00	02	00	00	00	06	.*..............
00000030	00	00	00	7A	01	10	00	02	00	00	00	06	00	00	00	80	...z...........€
00000040	01	12	00	03	00	00	00	01	00	01	00	00	01	1A	00	05	.....
00000050	00	00	00	01	00	00	00	86	01	1B	00	05	00	00	00	01	.....†.....
00000060	00	00	00	8E	01	28	00	03	00	00	00	01	00	02	00	00	...Ž.(.....
00000070	01	31	00	02	00	00	00	3D	00	00	00	96	02	13	00	03	.1.....=...–....
00000080	00	00	00	01	00	01	00	00	87	69	00	04	00	00	00	01	.....‡i.....
00000090	00	00	00	D4	00	00	00	00	4E	6F	6B	69	61	00	45	37	...Ô....Nokia.E7
000000A0	32	2D	31	00	00	00	01	2C	00	00	00	01	00	00	01	2C	2-1.....,.....,
000000B0	00	00	00	01	54	53	52	20	57	61	74	65	72	6D	61	72	....TSR Watermar
000000C0	6B	20	49	6D	61	67	65	20	53	6F	66	74	77	61	72	65	k Image Software
000000D0	20	33	2E	34	2E	31	2E	34	20	77	77	77	2E	77	61	74	3.4.1.4 www.wat
000000E0	65	72	6D	61	72	6B	2D	69	6D	61	67	65	2E	63	6F	6D	ermark-image.com
00177F30	AD	90	C4	0F	5A	8A	46	FD	E7	CF	B0	B6	4F	5C	77	34	..Ä.ZŠFýçÏ°¶O\w4
00177F40	51	49	2B	19	09	CA	FF	00	12	E4	92	08	1D	31	49	71	QI+..Êÿ..ä'...1Iq
00177F50	FB	C8	F0	4E	DD	DF	ED	63	A5	14	51	63	42	4B	CB	88	ûÈðNÝßíc¥.QcBKË
00177F60	DB	3B	B0	D0	B2	02	41	3C	E7	3E	D4	CB	8B	87	8E	69	Û;°Ð².A<ç>ÔË‹‡Ži
00177F70	BD	36	29	E0	63	91	C7	5A	28	A2	C0	3F	ED	63	73	00	½6)àc'ÇZ(¢À?ícs.
00177F80	15	F6	F4	0C	C4	7D	69	B1	DC	79	DB	38	50	D1	6E	C1	.öô.Ä}i±Üy08PÑnÁ
00177F90	0E	3B	D1	45	34	66	24	93	BB	6F	F2	49	0D	8C	02	48	.;ÑE4f$"»oòI.Œ.H
00177FA0	3F	8D	48	B3	99	25	C2	82	C0	8E	01	EF	F8	D1	45	4B	?.H³™%Â‚À.ï0ÑEK
00177FB0	D3	63	58	EB	B9	5E	3C	70	B8	18	23	B0	A4	99	7F	7C	ÓcXë¹^<p¸.#°¤™.|
00177FC0	09	0A	ED	20	C1	25	31	F8	FD	28	A2	A8	85	B1	FF	D9	..í Á%1øý(¢¨…±ÿÙ

Figure 3. The watermarked form of the image shown in Figure.1 and its hexdump (First, the picture was watermarked using the tool TSR Image Watermark and then the corresponding hexdump was generated using the HxD tool)

From the idea-expression perspective, any watermark (embedded as part of any software) is a genuine idea which is properly expressed in a manner that does not adversely affect the syntax (or sometimes even the semantics) of the software. It is a part of the source code (of the software) which is not a functional requirement of the software. In other words, watermark in any software is part of the requirements marking and identifying the original ownership of the software and not part of the requirements of the potential users of the software.

The above explanation of watermarks in terms of the idea-expression dichotomy clearly opens the door to linking watermarks directly to copyright infringements of any software because the idea-expression perspective is the basis of formulation of software copyright laws of several countries (Newman, 1999; Hollar, 2002). The idea-expression basis of copyright laws of several countries (especially the US copyright laws) says that if there is only one way or a limited number of (exclusive) ways of effectively expressing an idea, this idea and its expression tend to "merge" (Walker, 1996) and in such instances an idea and its expression are not protectable through copyright (Hollaar, 2002). In other words, if the same idea can be realized through more than a limited number of expressions, all such different realizations are protected by copyright laws. Thus, if the idea behind the expressions in a watermark (which is embedded in any particular copyrighted software) can be expressed in more than a limited number of ways, then the copyright obtained for the software can extend to the watermark contained in it. Thus, watermarks are directly linked to copyright.

This link requires further explanation. Even if the copyright of the main software can be extendable to the watermark contained in it as well, the copyright may not be extendable to all the elements of the watermark. This non-extendability is because a watermark can contain several legally unprotectable elements such as globally common mnemonics, names and expressions, globally shared notations, codes or expressions due to shared nature of technology, and globally common functional area elements.

If all the elements in the watermark are unprotectable, then the copyright obtained for the software will not be extendible to the watermark contained in the software. Finally, if there is at least one protectable element in the watermark contained in software, the copyright of any software will extend to the watermark contained therein as well.

To summarize, watermarks can be perceived in terms of idea-expression dichotomy, and thus, can be directly linked to copyright and can also be an indicator of software copyright infringement.

4. FORENSIC IMPORTANCE OF COPYRIGHTED WATERMARKS

Despite their apparent functionally irrelevant and thus innocuous status in any software, watermarks, when copyrighted (that means, when there is at least one protectable element in a watermark), can be of great value / assistance to the cyber forensic expert and a discussion of this evidence is the prime objective of this article.

The approach to the forensic importance of a watermark can be best done in the context of the concept of programming blunders (Bhattathiripad, 2012). A programming blunder has been defined as a "variable in a program or a code segment in a program … which is …. unnecessary for the user's functionality". Looking from this definitional point of view, a watermark is technically (or can be explained in terms of) a programming blunder because a watermark in any software is not part of the functional requirements of the software or (in other words) is unnecessary for user's functionality. The locus standi and functionality of watermarks can thus be best situated through their inclusion in the category of blunders.

Even so, unlike a typical programming blunder, watermark is neither unintentional nor accidental. Rather, it is an intentional 'programming blunder', introduced into the software by its developer for a specific purpose. In general, every watermark is an intentionally introduced software element and is technically an intentional programming blunder. Because watermarks are intentionally introduced code segments in any software, the three etiological factors of programming blunders (see Bhattathiripad, 2012) are not sufficient enough to explain the etiology of watermarks. All the

existing etiological factors of programming blunders assume that programming blunders can happen only due to inability or inattention of the programmer (or the quality engineer) to completely remove those statements that are not required for user's functionality. This also means that the existing etiological aspects of programming blunders do not consider the possibility of programming blunders happening due to software developer's intentional effort to introduce (into a software), a code segment (like a watermark) which is not required for user's functionality.

While doing a juxtaposed comparison of two sets of software to establish possible copyright infringement, the existence of a particular watermark in identical contexts in both the complainant's and the defendant's versions can be a more positive indication of illegal copying (than other kinds of blunders), as the watermark was deliberately inserted into but not carelessly leftover in the complainant's version. It is highly unlikely that two programmers will design and insert exactly same watermarks exactly in the same position and exactly in the same way, and this elevates the similarity into potential evidence of copyright infringement.

Thus, most watermarks can provide direct evidence (or at least probable, corroborative or supporting evidence) to establish copyright infringement more decisively than other programming blunders. In the absence of other direct evidence of copyright infringement, watermarks can form the only basis of the expert opinion to the judiciary about the possibility of copyright infringement.

5. WATERMARKS AS EVIDENCE IN COPYRIGHT INFRINGEMENT FORENSIC TEST

The importance of watermarks has not been given any role or status in the forensic procedure of the Abstraction-Filtration-Comparison (AFC) test (which is the only judiciary-accepted procedure for establishing software copyright infringement in the US) (Bhattathiripad, 2014). Watermarks are not even considered during this test because during the abstraction of the software, only the functionally active or relevant parts (of the two sets of software) will be considered for abstraction and used for further investigation (Hollaar, 2002). As a result, the functionally irrelevant parts (or those items that are irrelevant for user's functionality, like watermarks) may not be considered for abstraction. In such case of unfortunate non-consideration, the watermarks will not be available for final comparison and this unavailability certainly adversely affects the rigour of the AFC test and thus, can affect its reliability.

Hence, this paper proposes that, along with the AFC test results, the evidence concerning watermarks, if any, should also be identified and gathered separately by the forensic expert, before the final findings and inferences are presented to the court.

The software forensic research community is encouraged to take on this proposal and find ways to incorporate watermarks in the AFC test.

The judiciary systems worldwide also need to be encouraged to study and legalize the evidential aspects of digital watermarks in the context of copyright laws. Some preliminary suggestions are presented below.

During the forensic analysis as part of any software copyright infringement litigation, any watermark (embedded into a software package by the developer and identified and detected by the forensic expert) needs to be considered as a separate program segment. In other words, during the forensic test in any copyright infringement litigation, the embedded watermark needs to be first separated[1] from the main software and then

[1] The task of separation of the source code of a watermark from the source code of the main software (or any digitally watermarked file) can be easy if and only if the source code of the watermark can be perfectly identified in the original source code as a single unit of code segments. To put it clearer, the task of separation of watermarks from an image / audio / video file can be complicated and strenuous, for many reasons. Two such potential reasons are (a) the hex values of the watermark get fragmented (as against an identifiable single unit) in the ocean of hex values of any watermarked image / audio / video file and (b) the watermarking algorithm not only inserts the hex values of watermarks into the original (non-watermarked) image / audio / video file but also modifies a few, if not all, hex values of the original. Even so, this task of

subjected to the forensic test separately. This is in order to ensure that the watermark has (or does not have) protectable elements. The ultimate goal here is to establish whether the copyright of the main software is (or is not) extendable to the watermark as well. For instance, if the test used is AFC, then the watermarks in both the software packages need to be separated first, and then separately abstracted. Subsequently, the unprotectable elements in both watermarks need to be filtered out and removed. Finally, the comparable elements in the remaining "golden nuggets" (Walker, 1996) need to be compared and the resulting evidence (or evidence of infringement of protectable elements) needs to be reported to the court. If the test used is POSAR (Bhattathiripad, 2014), watermarks need to be separately subjected to this 5-stage forensic test process and the resulting evidence[2] need to be reported to the court.

Although outside the purview of AFC and POSAR, the evidence of copyright infringement of watermark will form part of the evidence of copyright infringement of the main software as well (because watermark is a part of the main software) and sometimes, can turn out to be valuable evidence to establish copyright infringement of the main software.

Before concluding, a note on what a judge expects from a forensic expert would add value to the special attention and consideration given to watermarks. In any software comparison report, what the judge would expect from the forensic expert is a set of details that helps the court in arriving at a decision on the copyrightable aspects of the elements in both software packages (Newman, 1999). So, what is expected in the case of watermarks is not a mere statement on the extendability of the copyright to the watermarks. Rather, the statement should be substantiated and supported by a set of details on the merger aspects of the ideas and expressions contained in the watermarks.

It needs to be stated here that the future research on forensics of watermarks should not ignore all these complex aspects that determine the status and role of watermarks in copyright cases.

6. CONCLUSION

In the process of legally establishing copyright infringement, the watermark (contained in the software) can play an important role. As any watermark can be considered to be technically a programming blunder, the forensic importance and philosophy of programming blunders (explained in the context of the idea-expression dichotomy) can be extendible to every watermark as well. While doing a juxtaposed comparison of two sets of software to establish possible copyright infringement, the existence of watermarks in identical contexts in both versions can be a more positive indication of illegal copying (than other kinds of blunders), as they were deliberately inserted into and not carelessly leftover in the complainant's version.

In order to use watermark as evidence to establish the criminal activity behind the infringement allegation, both the forensic procedure (used as part of the investigation) and the judge's decision making process need to be sensitive to the forensic role of watermarks. Hence the forensic tests (to establish software copyright infringement) need to be re-designed so as to ensure that possible evidence like watermarks are procedurally collected, forensically analyzed and then properly reported to the court. The forensic report should be substantiated and supported by a set of details on the merger aspects of the ideas and expressions contained in the watermarks. Future research in this area should ensure not to leave out the importance of watermarks as well as their role in establishing software copyright infringement.

separation is not impossible if the algorithm for separation is sensitive to both the insertions and the modifications done by the watermarking algorithm.

[2] The evidence set here contains the evidence of infringement of protectable elements along with the evidence of post-piracy modifications and the evidence of infringement of programming blunders as part of the watermark

REFERENCES

[1] Bhattathiripad, P. V. (2012), Software Piracy Forensics: A Proposal for Incorporating Dead Codes and other Programming Blunders as Important Evidence in AFC Test, Proceedings of the IEEE 36th International Conference on

Computer Software and Applications Workshops, Turkey

[2] Bhattathiripad, P. V. (2014). Judiciary-friendly forensics of software copyright infringement, IGI-Global

[3] Collberg, C., & Thomborson, C. (1999). Software watermarking: Models and dynamic embeddings, Proceedings of the 26th ACM SIGPLAN-SIGACT symposium on Principles of programming languages (pp. 311-324). ACM.

[4] Cox, I. J., Miller, M. L., & Bloom, J. A. (2008). J. Fridrich, T. Kalker, Digital Watermarking and Steganography, Morgan Kaufmann Publishers

[5] Hollaar, L. A. (2002) "Legal Protection Of Digital Information", BNA Books, 2002

[6] Nagra, J., Thomborson, C., & Collberg, C. (2002). A functional taxonomy for software watermarking. In *Australian Computer Science Communications* (Vol. 24, No. 1, pp. 177-186). Australian Computer Society, Inc..

[7] Marcella, A. J. Jr., and Menendez, D. (2008), Cyber Forensics, Auerbach Publications, page 3

[8] Newman, J. O. (1999), "New Lyrics For An Old Melody, The Idea/Expression Dichotomy In The Computer Age", 17, Cardozo Arts & Ent. Law Journal, p.691

[9] Walker, J., (1996), Protectable 'Nuggets': Drawing the Line Between Idea and Expression in computer Program Copyright Protection, 44, Journal of the Copyright Society of USA, Vol 44, Issue 79