개선된 Lowbit Encoding 방법을 이용한 StegoWaveK의 구현

(Implementation of StegoWaveK using an Improved Lowbit Encoding Method)

김 영 실 † 김 영 미 †† 백 두 권 †††

(Young-Sil Kim) (Young-Mi Kim) (Doo-Kwon Baik)

요 약 멀티미디어 데이타중에서 오디오데이타를 이용한 상용화된 오디오 스테가노그라피(audio steganography) 소프트웨어들은 시각적인 측면에서 비밀 메시지가 은닉되어 있다는 것을 쉽게 인지할 수 있다는 것과 숨길 정보의 크기에 제한이 있다는 문제점을 가지고 있다. 이러한 문제점을 해결하기 위해 동적으로 메시지를 은닉하는 방법을 제안하였다. 또한 비밀 메시지의 보안수준을 향상시키기 위해 파일 암호화 알고리즘도 적용하였다. 본 논문에서는 제안한 오디오 스테가노그라피를 수행해주는 StegoWaveK 시스템을 상용화된 오디오 스테가노그라피 시스템의 5가지 측면으로 비교 분석하였으며, 성능면에서 우수함을 보였다. StegoWaveK 는 상용화된 시스템에 비해 시각적 공격 측면이나 은닉할 메시지 크기 측면에서는 좋으나 인터페이스 측면에서는 사용자 위주의 편리성을 제공할 수 있도록 보완되어야 한다. 그리고 StegoWaveK와 상용화된 시스템이 견고성이 약하다는 단점을 보완해야 하며, 다양한 멀티미디어 데이타를 이용한 스테가노그라피로의 추후연구가 필요하다.

키워드 : 스테가노그라피, 암호화 알고리즘, 비밀 메시지, 암호문, 스테고 데이타

Abstract The steganography is one of methods that users can hide data. Some steganography softwares use audio data among multimedia data. However, these commercialized audio steganography softwares have disadvantages that the existence of hidden messages can be easily recognized visually and only certain-sized data can be hidden. To solve these problems, this study suggested, designed and implemented Dynamic Message Embedding (DME) algorithm. Also, to improve the security level of the secret message, the file encryption algorithm has been applied. Through these, StegoWaveK system that performs audio steganography was designed and implemented. Then, the suggested system and the commercialized audio steganography system were compared and analyzed on criteria of the Human Visible System (HVS), Human Auditory System (HAS), Statistical Analysis (SA), and Audio Measurement (AM).

Key words : steganography, encryption algorithm, cover-data, secret message, stego-data

1. 서 론

컴퓨터와 통신시스템의 비약적인 발전으로 인하여 데이타를 안전하게 보호하기 위한 방안들이 개발되고 있다. 가장 근원이 되는 기법이 암호화이며 이와 더불어 다양한 형태의 정보 은닉 기술이 연구되고 있다[1]. 이중 대표적인 응용기술이 스테가노그라피(steganography)이다. 스테가노그라피는 데이타를 다양한 형태의 자료(텍스트, 이미지, 동영상, 오디오 등)에 은닉함으로써 숨겨진 데이타를 찾아 내지 못하도록 지원하는 기술이다[2,3,4,5]. 일반적으로 비밀 메시지를 암호화하여 은닉하게 되면 은닉된 비밀 데이타가 공격자(attacker)에 의해 발견된다 하여도 공격자는 그 암호화된 메시지를 복호화하기 위해 노력을 해야 하기 때문에[6] 그 비밀 메시지에 대한 보안 수준이 한층 더 안전하고 높아지게 된다. 또한 은닉하려고 하는 비밀 메시지의 크기에

† 정 회 원 : 대림대학 컴퓨터정보계열 교수
pewkys@daelim.ac.kr
†† 비 회 원 : (주)쎄스 암호화연구실장
rose@cest.co.kr
††† 종신회원 : 고려대학교 컴퓨터학과 교수
baik@dblab.korea.ac.kr
논문접수 : 2003년 2월 24일
심사완료 : 2003년 3월 21일

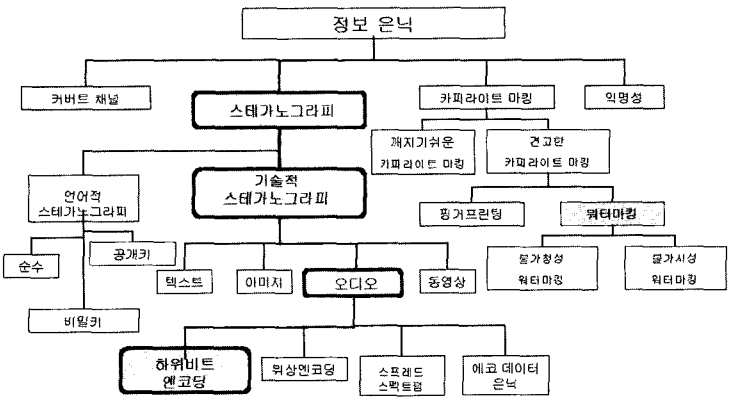


그림 1 정보은닉의 분류[9]

따라 필요한 데이타의 크기가 결정되므로 비밀 데이타를 압축하여 은닉하고 있다. 이 때 비밀 메시지를 숨기고자하는 데이타를 원본(cover-data)이라고 하고 스테가노그라피 방법을 이용하여 데이타를 은닉하고 있는 것을 스테고 데이타(stego-data)라고 한다. 현재 가장 일반적으로 발전된 스테가노그라피 분야가 이미지를 이용한 스테가노그라피이며, 또 다른 관심 영역이 오디오를 이용한 스테가노그라피이다. 일반적으로 오디오 스테가노그라피에서는 음악을 듣는 청취자가 실제로 청취하지 못하는 부분에 데이타를 숨기기 때문에 실제 데이타가 숨겨져 있다는 것을 일반 청취자들은 알지 못한다. 뿐만 아니라 스테고 데이타의 크기가 원본의 크기와 차이가 없기 때문에 차이를 인식하지 못한다. 오디오 스테가노그라피에서 견고성 측면과 숨기고자하는 정보의 크기 측면을 고려할 때 견고성 측면을 강조하는 방법은 디지털워터마킹과 같은 저작권 분야에서 사용이 되나 보다 많은 정보를 은닉하기 위한 방법으로는 하위비트 엔코딩(Lowbit Encoding)으로 상용화된 오디오 스테가노그라피에서 사용하는 것이다[7,8]. [2]에 의하면 오디오 스테가노그라피는 오프라인상태에서 정보은닉에 매우 유용하다. 암호화와는 달리 오디오 스테가노그라피에서는 비밀 메시지를 위한 추가적인 메모리를 요구하지 않으면서 오디오 파일의 크기 변화 없이 정보를 은닉하는 방법을 지원한다. 본 논문에서는 이러한 방법을 사용하면서 숨겨지는 정보의 손실 없이 음질의 차이를 인식하지 못하는 범위내에서 좀 더 큰 크기의 비밀 메시지를 은닉하면서 시각적 공격시에도 인지하지 않도록 기존의 방법을 개선한 StegoWaveK 시스템을 제안하고 설계 구현했다.

2. 관련 연구

암호화와 더불어 보안을 위해 활발히 연구되고 있는 분야가 바로 정보은닉 분야이다. 정보은닉(Information Hiding) 기술은 크게 그림 1과 같이 커버트 채널(covert channel), 스테가노그라피, 익명성, 카피라이트 마킹(Copyright Marking)의 네 가지로 분류되며[9], 이 중 스테가노그라피와 카피라이트 마킹의 경우에는 비밀 메시지를 은닉하기 위해 추가적인 저장 공간이 필요하지 않다는 특징을 가지고 있다[3].

이 중 기원전부터 사용된 대표적인 정보은닉 기술이 스테가노그라피이며, 현재 지적 재산권을 보호하기 위해 디지털 워터마킹과 함께 응용되고 있다. 정보은닉 기술의 적용분야들을 살펴보면 디지털 워터마킹은 지적재산권보호, 상표 및 로고 관리, 전자도서관, VOD 등과 같은 영역에서 사용된다. 핑거프린팅(fingerprinting)은 디지털데이타의 인증 식별, 소프트웨어 불법복제 방지 등에 활용되며, 익명성은 이동통신에서 이동기지국의 위치 추적장치, 전자상거래의 익명성 등에서 사용된다. 스테가노그라피는 군사적, 국가적 기밀 통신이나 정보은닉에 사용된다. 커버트 채널은 은 컴퓨터바이러스, 키복구 시스템 등에 사용된다[10].

스테가노그라피 시스템의 정보 모델에 관한 연구는 [11,12,13]에서 주로 이루어져 왔다. 또 [2]에 의하면 지금까지 스테가노그라피의 주관심 대상이 이미지에서 오디오 영역으로 전이되고 있으며, 특히 오프라인에서 정보은닉을 수행하는 더욱 중요한 멀티미디어 데이타로서 역할을 하게 될 것이라고 한다. 특히 응용 분야에 따라 적용되는 오디오 스테가노그래피가 다르다고 한다. 정보은닉과 디지털 워터마킹의 또 다른 접근방법인 QIM

(Quantization Index Modulation)[14]이 있으며 이것은 주로 이미지에 정보를 은닉하기 위해 사용된다.

2.1 스테가노그라피와 디지털 워터마킹

스테가노그라피와 디지털 워터마킹은 응용에서 요구하는 측면에 따른 구분이며, 그림 2는 스테가노그래피와 디지털 워터마킹의 차이이다. 그림 2에서 보는 바와 같이 응용에서 가장 강조되어야하는 측면에 따라 각 요소들은 장단점을 가지고 있으며 어떠한 요소에 무게를 두느냐가 구현하는 방법을 선정하게 한다.

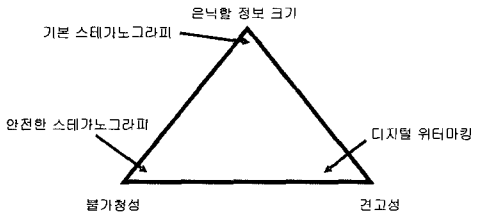


그림 2 스테가노그라피와 디지털 워터마킹[15]

오디오 스테가노그라피를 이용한 정보은닉에서 인간의 불가청성과 은닉 정보의 크기에 비중을 두는 영역에서는 기본(basic) 스테가노그라피 시스템이 요구된다. 특히 FFT와 같은 주파수를 이용한 방법을 사용하는 경우는 손실 변환이므로 숨길 메시지들이 일반적인 사용자들이 만들고 사용하는 다양한 문서들을 하나의 손실 없이 그대로 은닉한 후 다시 추출하고자할 때는 추가적인 방법들을 적용하는 여러 처리단계가 요구된다[16,17].

2.2 오디오 스테가노그라피(Audio Steganography)

상용화된 스테가노그라피 소프트웨어들은 [18]에서 볼 수 있으며 이 중에서 오디오 스테가노그라피는 인간의 가청 체계(Human Auditory System)에서 일정한 영역을 벗어나는 음은 사람이 구분하지 못한다는 특성을 이용하는 것이다. 그러므로 오디오 파일에 메시지를 숨길 수 있다면 쉽게 비밀 정보를 전달 할 수 있다는 개념에서 개발된 기술이다. 오디오 스테가노그라피가 가능하도록 해주는 기법은 하위비트 엔코딩, 위상엔코딩(phase encoding), 스프레드 스펙트럼(spread spectrum), 에코 데이타 은닉(echo data hiding) 등이 있다[2,6,19].

하위비트 엔코딩 기법은 다른 데이타 구조에 데이타를 삽입하기 위한 가장 간단한 방법으로 이진 스트림으로 샘플링된 비트의 마지막 비트를 숨기고자 하는 비밀 데이타로 대체한다. 이 방법은 오디오 시그널에 다른 방법에 비해 큰 크기의 데이타를 인코딩할 수 있지만 송수신 채널의 잡음이나 리샘플링, 압축등의 일반적인 신호처리 시 은닉된 데이타를 필터링하기 어렵다[4,5].

위상 엔코딩은 초기 오디오 세그먼트의 위상을 데이타를 표현하는 연관된 위상으로 바꾸는 방법이다. 결과 세그먼트의 위상은 세그먼트 사이에 관련된 위상을 유지하기 위해 조정되며 시그널과 감지할 수 있는 잡음 사이에 효과적인 코딩 방법 중에 하나이다. 하지만 메시지를 필터링하기 위해 주파수 영역의 길이와 시작 위치점을 파악해야 하며, 메시지 필터링 시 원본이 필요하다.

스프레드 스펙트럼은 대부분의 통신 채널의 대역폭을 보존하기 위해 가능한 한 스펙트럼의 정밀한 영역에 오디오 데이타를 집중시킬 수 있다는 특성을 이용한 방법으로 자연 잡음이나 고의적인 전파 방해에 강한 면을 가지고 있다.

에코 데이타 은닉방법은 에코의 삽입에 의해 원본 데이타에 데이타를 은닉하는 방법이다. 데이타를 초기 진폭(initial amplitude), 지연률(decay rate), 그리고 오프셋을 세 개의 매개변수로 변화를 주어 데이타를 숨기며, 원본과 에코사이의 오프셋은 두 신호를 섞는다[4,5,20]. 에코는 단지 추가된 소리로 들리며, 인간의 청력이 두 시그널 사이의 차이점을 구분하지 못한다. 표 1은 오디오에 메시지를 은닉하는 방법을 비교한 것이다.

표 1 오디오에 메시지를 은닉하는 방법[2]

방법	하위비트 엔코딩	에코 데이타 은닉	위상 코딩	스프레드 스펙트럼
메시지 크기 (bps)	높다 (44,100)	낮다 (16)	보통이다 (20)	매우 낮다 (4)
견고성	나쁘다	좋다	좋다	매우 좋다

은닉 정보의 크기 측면으로 표 1에 제시된 방법들을 비교해 보면 음악이 1분 정도 연주되는 곡이라고 가정했을 때 하위비트 엔코딩은 846000비트이고 에코 은닉은 960 비트, 위상 코딩은 1200 비트, 스프레드 스펙트럼은 240 비트를 저장할 수 있다. 따라서 표 1에서 제시된 방법 중에서 하위비트 엔코딩을 제외한 다른 방법들은 압축과 같은 공격에는 강하지만 저장할 수 있는 데이타의 양이 너무 적고, 신호 재생시 동기화가 중요한 관건이 된다. 또한 숨겨진 메시지를 숨겨지기 이전 상태와 같도록 삽입 추출하기 위해서는 추가적인 기술들이 필요하다[17,20]. 이러한 방법들은 일반적인 문서들을 은닉하기에는 부적당하며, 주로 저작권과 관련된 분야에 이용된다. 따라서 견고성은 나쁘지만 저장할 수 있는 데이타의 크기가 큰 방법을 선택해서 원하는 정보를 은닉하고 정보의 손실 없이 다시 추출할 수 있어야 한다.

2.3 상용화된 오디오 스테가노그라피 시스템의 문제점

상용화된 스테가노그라피 소프트웨어중에서 오디오를 지원하는 것은 PCM 방식의 웨이브 파일을 원본으로 사용하면서 하위비트 엔코딩 방법을 사용한다. PCM 방식의 웨이브 파일을 사용하는 이유는 잡음과 간섭에 강하고, 전송 중 코딩된 신호를 효과적으로 재생하며, 신호대잡음비인 SNR을 개선하기 위한 채널대역폭의 증가를 효과적으로 바꿀 수 있다. 또, 동일한 포맷으로 공통된 네트워크에서 다른 디지털 데이타와 합칠 수 있으며, TDMA 시스템에서 신호를 빼거나 삽입하기가 쉽고, 특수한 변조나 암호화를 적용하기가 쉽다[21].

다음 그림 3은 일반적인 하위비트 엔코딩을 이용한 데이타 삽입 방법을 나타낸 것이다. 오디오 샘플 안에 "A"를 인코딩한 경우의 예이다.

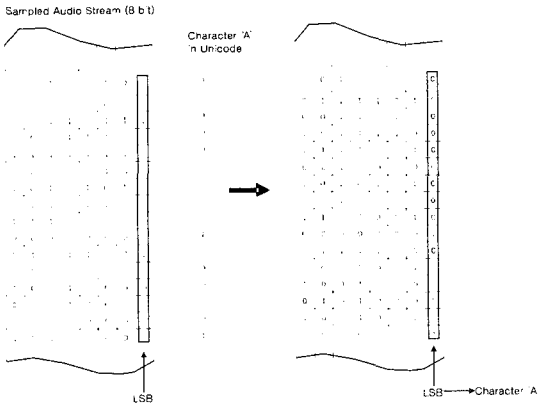
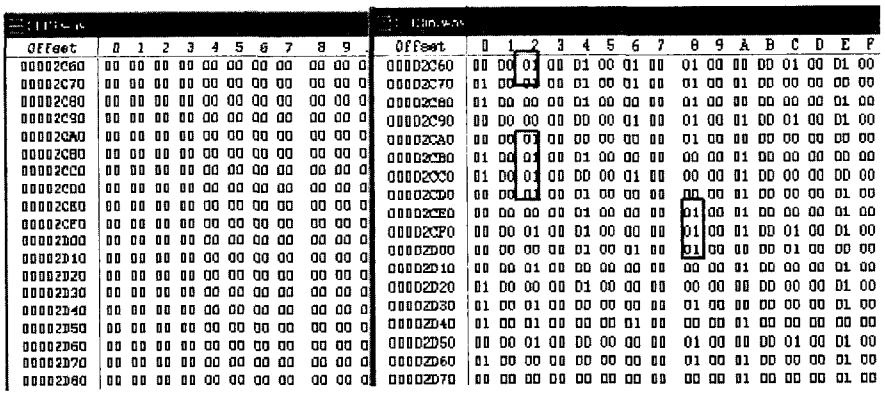


그림 3 오디오 샘플에 "A"를 엔코딩 예

기본적으로 오디오 스테가노그라피에서는 청각적으로 원음에 영향을 미치지 않는 범위 내에서 정보를 은닉해야 하며 스테고 데이타에서 은닉된 비밀 메시지를 손실없이 검출할 수 있어야 한다. 현재 오디오 스테가노그라피를 지원해주는 소프트웨어의 문제점을 살펴보면 다음과 같이 크게 두 가지이다.

첫째 하위비트 엔코딩의 경우 16비트 마다 비밀 메시지의 1비트를 은닉하므로 산술적으로 비밀 메시지의 16배에 해당하는 원본이 필요하다. 상용화된 시스템들을 테스트해 보면 웨이브가 45MB일 때 크기가 5MB인 한글워드 문서가 비밀메세지로 선택되면 은닉되지 않는다. 따라서 PCM 방식의 특성인 잡음에 강하다는 것을 이용하여 이러한 비밀 메시지의 크기를 좀 큰 것을 사용하면서도 음질의 차이를 느끼지 못하는 범위 내에서 데이타를 은닉하는 방법이 필요하다. 물론 비밀 메시지를 압축하게 되면 실제 16배 크기의 원본이 필요하지는 않다. 그러나 압축된 비밀 메시지를 잘 은닉하기 위해서는 16배 이상의 원본을 필요로 하기 때문에 비밀 메시지를 은닉하기 위해 사용되는 원본의 선정을 어렵게 한다. PCM 방식이 잡음에 강하므로 원본과 구별이 되지 않는 범위 내에서 가능한 많은 비밀 메시지를 은닉할 수 있는 기술의 개발이 요구된다. 이 외에도 상용화된 시스템은 웨이브 파일의 헤더크기에 따라 같은 16비트 PCM 방식이라고 하더라도 처리하지 못한다.

둘째 압축된 데이타를 PCM 방식의 데이타 포맷인 16비트의 가장 오른쪽 한 비트에만 저장함으로써 비록 일반 청취자들이 웨이브 파일을 듣거나 또는 단순히 파형을 눈으로 보는 것만으로는 정보가 은닉되어 있다는 사실을 알 수 없다 하더라도 데이타 영역인 데이타 청



원본 스테고 데이타

그림 4 헥사편집기로 읽어들인 원본과 스테고 데이타

크(data chunk)의 시작부분부터(대부분 0으로 되어 있다.) 순차적으로 비밀 메시지를 삽입하기 때문에 데이타 청크부분의 값을 살펴보면 쉽게 정보가 은닉되어 있다는 것을 알 수 있다. 그림 4는 상용화된 시스템(Invisible Secrete 2002)에서 만들어진 스테고 데이타와 원본을 헥사편집기에서 파일열기를 한 것인데, 원본과 스테고 데이타간의 차이가 있음을 눈으로 식별할 수 있다. 즉, 이것은 상용회된 시스템들이 숨길 메시지를 저장할 때 데이타 영역의 처음부터 저장하므로써 발생하는 문제이다. 이러한 문제점을 보완하기 위해선 비밀 메시지를 하위비트가 아닌 웨이브 파일의 특성을 변경하지 않는 범위 즉 가청도의 특성을 유지할 수 있는 범위 내에서 임의의 위치 비트를 기준으로 삽입할 필요가 있다. 뿐만 아니라 비밀 메시지를 삽입, 추출하는 알고리즘이 알려져도 불법적인 비밀 메시지 검출이 쉽게 이루어지지 않도록 구현되어야 한다.

3. 개선된 Lowbit encoding 알고리즘

상용화된 스테가노그라피 시스템들의 문제점을 해결하고 PCM 방식의 웨이브 파일을 이용한 오디오 스테가노그라피를 위해 능동적으로 비밀 메시지 은닉하는 알고리즘을 제안한다.

3.1 메시지 은닉 알고리즘

원본에 비밀 메시지를 은닉하는 여러 가지 방법들을 정의하는 은닉 함수(embedding function)들은 표 2와 같이 정의한다. 은닉함수들은 웨이브 파일들의 집합인 C와 은닉메시지 집합 M를 정의역으로 하고 C 집합에 있는 웨이브 파일과 같은 크기와 같은 음악을 제공하는 웨이브 파일들의 집합인 S로 가는 함수이다.

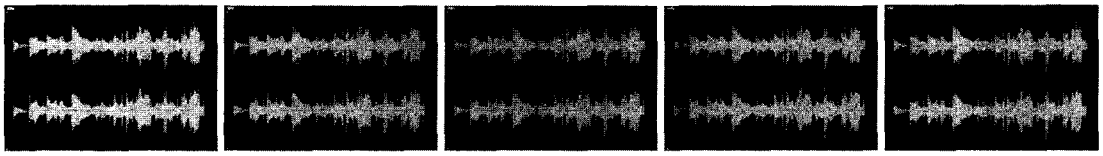
예를 들어, 원본에 1 비트씩 하위비트에 은닉하는 경우 S_{M1}이라고 정의하고, 2 번째 비트에 은닉 할 때는 S_{M2}라고 하며, 현재 상용화된 오디오 스테가노그라피도 하위비트 엔코딩으로 비밀 메시지를 은닉하고 있으므로, f_{M1}을 적용한 것과 같다. 일반적으로 이러한 방법이 사용되는 가장 큰 이유는 비밀 메시지를 하위비트에 은닉한 뒤에 생성된 스테고 데이타 즉 S_{M1}이 원본과 가장 유사한 특성을 가지며 또한 간단히 구현할 수 있기 때문이다. 그러나 이 방법은 16비트마다 1비트씩 비밀 메시지를 은닉하기 때문에 비밀 메시지를 은닉하기 위해 사용할 수 있는 원본을 선택하는데 제한을 갖는다는 문제점이 있다. 이를 보완하기 위해선 스테고 데이타가 원본과 유사한 특성을 가지는 범위 내에서 원본에 은닉되는 비밀 메시지를 16비트 당 1비트 이상으로 저장하여, 은닉된 정보의 크기를 향상시키기 위해 누적하여 삽입하는 함수 Af_{Mi}를 표 2와 같이 정의했다.

Embedding Function f_{Mi}과
Extraction Function Ef_{Mi} 알고리즘

Embedding Function f_{Mi}	Extraction Function Ef_{Mi}
k-insertion position i=1 ; j=1 do until i > $\ell(c)$ $S_i \leftarrow C_i$ i++ loop do until i > $\ell(m)$ $S_{jk} \leftarrow m_i$ i++; j++ loop	k-insertion position i=1 ; j=1 do until i > $\ell(c)$ i++ loop do until i > $\ell(m)$ $m_i \leftarrow S_{jk}$ i++; j++ loop

표 2 다양한 은닉함수

은닉 함수	함수표현	설명
f_{Mi}	$f_{Mi}(C_j, M_j) = S_{Mi}$	$i \in N$, S_{Mi}는 원본의 i비트 위치에 1비트씩 은닉한 스테고 데이타, $C_j \in C$는 원본의 집합
Af_{Mi}	$Af_{Mi}(C_j, M_j) = AS_{Mi}$	AS_{Mi} 는 원본의 i비트씩 누적하여 비밀 메시지를 은닉한 스테고 데이타
Sf_{Mi}	$Sf_{Mi}(C_j, M_j) = SS_{Mi}$	SS_{Mi} 는 원본의 i 번째 비트까지 사인곡선을 그리면서 한비트씩 비밀 메시지를 은닉한 스테고 데이타
ASf_{Mi}	$ASf_{Mi}(C_j, M_j) = ASS_{Mi}$	ASS_{Mi} 는 원본의 1비트씩 가장 오른쪽 비트부터 i 번째 비트까지 사인곡선 형태로 비밀 메시지를 은닉한 스테고 데이타



<원본> <AS_{M1}> <AS_{M2}> <AS_{M4}> <AS_{M6}>

그림 5 여러 가지 형태로 비밀 메시지를 은닉한 스테고 데이타 비교

그림 5는 원본에 비밀 메시지를 함수 Af_{Mi}에 적용한 스테고 데이타들의 실제 파형을 나타낸 것이다.

그림 5를 살펴보면 원본에 비밀 메시지를 삽입하는데 있어 AS_{M1}부터 AS_{M6}까지의 파장에는 별 차이가 없다. 청각적인 측면을 조사하기 위해 그림 6은 이러한 데이타들을 100명의 학생들에게 들려준 결과를 그래프로 표현한 것이다. 좀 더 정밀한 테스트를 위하여 헤드셋을 이용하였으며, 대다수의 학생들이 원본인 원본과 스테고 데이타과의 차이를 인식하지 못했다.

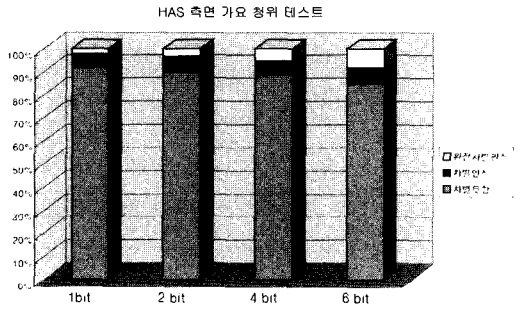


그림 6 웨이브 파일 청취 결과

그러나 은닉하려는 메시지의 크기를 늘리기 위해서 삽입되는 비트수를 무조건 늘리게 되면 원본과 함수 Af_{Mi}에 의해 생성된 스테고 데이타와의 실제 코드값 차이가 커지게 된다. 그러면 원본이 손상되므로 청취자는 원본에 이상이 있다고 생각할 것이며, 은닉된 정보를 찾는 공격자들은 쉽게 은닉된 비밀 메시지를 찾아낼 수 있다.

또한 원본과 Af_{Mi}에 의해 생성된 스테고 데이타들간의 어떠한 변화가 발생되었는지를 통계적으로 분석하기 위해 16비트 PCM 방식의 웨이브 파일을 원본으로 사용하였다[5]. 통계분석을 위해 원본과 각각의 스테고 데이타 모두를 16비트 단위로 구분하여 10진수로 변환한 분석용 데이타 3000개를 추출하여 SAS 패키지를 이용하여 상관분석(correlation analysis)을 수행하였다. 그림 7은 16비트 PCM 방식의 원본과 각각의 스테고 데이타를 분석한 결과이다.

분석한 결과를 살펴보면 AS_{M1}, AS_{M2}, AS_{M4}, AS_{M6}과 원본의 상관계수가 1에 가까우므로 각각의 스테고 데이타는 원본의 특성을 거의 유지하고 있다고 할 수 있다. 그러나 6비트를 삽입한 스테고 데이타만이 0.99988로서 원본에 특성이 어느 정도 누락되는 부분이 있음을 알 수 있다. 그러므로 6비트를 삽입하는 방법이 효과적이지 않으므로 원본과 유사한 특성을 나타내며 웨이브 파일에 16비트 당 삽입 가능한 비트 수는 2비트부터 4비트 사이라는 결과를 얻었다[5].

변수	N	평균	표준편차	합
원본	3000	94.03567	1637	282107
AS_{M1}	3000	94.04667	1637	282140
AS_{M2}	3000	94.08533	1637	282256
AS_{M4}	3000	94.00633	1637	282019
AS_{M6}	3000	94.69833	1638	281095

단순통계

변수	최소값	최대값
원본	-9382	11835
AS_{M1}	-9381	11834
AS_{M2}	-9382	11835
AS_{M4}	-9390	11826
AS_{M6}	9391	11791

피어슨 상관계수 N=3000

	AS_{M1}	AS_{M2}	AS_{M4}	AS_{M6}
원본	1.0000	0.00000	0.99999	0.99988
	<0.0001	<0.0001	<0.0001	<0.0001

그림 7 원본과 스테고 데이타와의 상관분석 결과

3.2 Capacity 향상을 위한 정보은닉 방법

하위비트 엔코딩은 공격자가 시각적으로 쉽게 은닉된 비밀 메시지를 인지할 수 있다. 이러한 문제점을 개선하기 위해 최하위 비트를 기준으로 비밀 메시지를 은닉하는 것이 아니라 비밀 메시지가 은닉되는 위치를 변경하여 비밀 메시지를 삽입하는 은닉함수가 필요하다.

실제로 상용화된 오디오 스테가노그라피의 경우 i 비트마다 1비트씩 비밀 메시지를 삽입하기 때문에 비밀 메시지가 삽입된 위치가 필터링 될 확률은 식 (1)과 같다.

$$P(bj) = \frac{1}{i} \quad (i=16,\ j\in\{1,2,3 \cdots 16\}) \qquad (1)$$

그림 8과 같이 사인곡선 형태를 이루도록 비밀 메시지 비트를 삽입하는 경우 사인곡선의 한 주기에 따라 i 비트마다 삽입된 비밀 메시지의 위치를 알아낼 확률을 구하면 식 (1)과 같다. 즉 i 비트마다 비밀 메시지의 1 비트를 은닉하였으므로, $P(bj) = \frac{1}{i}$이 된다. 필터링 된 bj 위치를 이용하여 bj+1의 위치를 알아낼 수 없고 P(bj)와 마찬가지로 $P(bj+1) = \frac{1}{i}$이 되므로 서로 독립사상이다. 따라서 비밀 메시지가 삽입된 위치를 알아낼 확률은 식 (2)와 같으며, 사인곡선의 한 주기에 은닉되는 비밀 메시지의 비트수가 필터링 될 확률은 16의 지수승 만큼 작아지게 된다.

$$P(b_1, \cdots, b_k) = \frac{1}{i^k} \qquad (2)$$

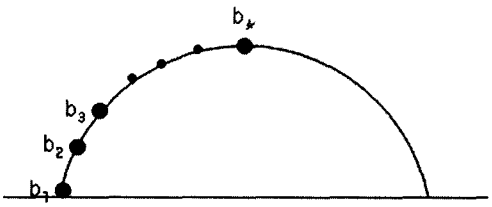


그림 8 사인곡선의 형태로 삽입된 비밀메시지 위치

따라서 상용화된 오디오 스테가노그라피와는 달리 각각의 비트가 은닉되는 위치가 달라지므로 은닉된 비밀메시지를 필터링이 어려워진다. 그러므로 은닉함수를 적용한 시스템이 상용화된 오디오 스테가노그라피보다는 쉽게 필터링 되지 않는다는 것을 알 수 있다.

다음은 사인곡선 형태의 누적 함수인 ASf_{Mi}을 적용한 ASS_{M4} 그리고 누적 함수인 Af_{Mi}를 적용한 AS_{M2}, AS_{M4}와 원본과의 상관관계를 분석한 결과이다. 분석한 결과를 살펴보면 모두 원본과 높은 상관관계가 나타남을 알 수 있다.

F-검정에서 상관관계가 있다고 나타난 AS_{M2}, AS_{M4}와 ASS_{M4}를 이용하여 원본과 각각의 스테고 데이타 사이에 10진수로 변환한 값에 차이가 있는 지를 검정하기 위하여 분산분석의 일원배치법을 적용하였다. 표 6에서 분석결과를 살펴보면 F 비값이 1.4E-05이므로 F 기각치 값인 2.605645보다 작아서 각각의 스테고 데이타들 간에 차이가 없는 것을 확인할 수 있다. 즉, 각각의 스테고 데이타들이 원본과 같은 특성을 가지고 있다는 결과를 얻었다.

표 3 원본과 AS_{M2}와의 F-검정

	원본	AS_{M2}
평균	94.03567	94.08533
분산	2680295	2680205
관측수	3000	3000
자유도	2999	2999
F 비	1.000033	
P(F<=f) 단측 검정	0.499635	
F 기각치: 단측 검정	1.061923	

표 4 원본과 AS_{M4}와의 F-검정

	원본	AS_{M4}
평균	94.03567	94.00633
분산	2680295	2680163
관측수	3000	3000
자유도	2999	2999
F 비	1.000049	
P(F<=f) 단측 검정	0.499464	
F 기각치: 단측 검정	1.061923	

변수	N	평균	표준편차	합
원본	3000	94.03567	1637	282107
AS_{M2}	3000	94.08533	1637	282256
AS_{M4}	3000	94.00633	1637	282019
ASS_{M4}	3000	94.23600	1637	282768

단순통계

변수	최소값	최대값
Origin	-9382	11835
AS_{M2}	-9382	11835
AS_{M4}	-9390	11826
ASS_{M4}	-9384	11828

피어슨 상관계수 N=3000

	AS_{M2}	AS_{M4}	ASS_{M4}
원본	1.0000	1.0000	1.0000
	<0.0001	<0.0001	<0.0001

그림 9 원본과 다양한 스테고 데이타와의 상관분석

표 5 원본과 ASS_{M4}와의 F-검정

	원본	ASS_{M4}
평균	94.03567	94.256
분산	2680295	2679531
관측수	3000	3000
자유도	2999	2999
F 비	1.000285	
P(F<=f) 단측 검정	0.496889	
F 기각치: 단측 검정	1.061923	

표 6 분산분석 일원배치법 적용 결과

인자의 수준	관측수	합	평균	분산
원본	3000	282107	94.03567	2680295
AS_{M2}	3000	282256	94.08533	2680205
AS_{M4}	3000	282019	94.00633	2680163
ASS_{M4}	3000	282768	94.256	2679531

분산 분석						
변동의 요인	제곱합	자유도	제곱 평균	F 비	P-값	F 기각치
처리	112.1817	3	37.39389	1.4E-05	1	2.605645
잔차	3.21E+10	11996	2680049			
계	3.21E+10	11999				

통계적으로 분석한 결과에 의하면 비밀 메시지의 필터링을 어렵게 하며, 원본이 은닉할 수 있는 메시지의 크기를 높일 수 있는 가장 효과적인 삽입 방법이 ASf_{Mi} 임을 알 수 있다. 다음은 원본에 비밀 메시지를 은닉시키는 ASf_{Mi} 의 알고리즘이다.

ASS_{Mi}가 S_{Mi}에 비해 숨길 메시지의 크기가 크다는 것을 증명하기 위해 capacity 함수 Cf(W, F)와 f_{SW}를 표 7과 같이 정의하였다.

<정리 1> 원본 Wc에 대해서 함수 f_{MI}를 적용하면 $Cf(W_c, f_{MI}) = k_{MI} \Rightarrow k_{MI} = f_{SW}(w_c)/16$과 같다.

[증명] 원본 Wc에 대해서 함수를 적용하면 Nwc를 얻게 된다. 따라서 하위비트에 저장하는 방법을 사용하여 16비트 PCM 방식의 웨이브 파일에 비밀 메시지를 삽입하는 것은 데이타 청크 부분에서 16비트 마다 1비트씩 저장하는 것이므로 $Cf(W_c, f_{Mi}) = k_{Mi}$는 Nwc를 16으로 나눈 $f_{SW} = (w_c)/16$ 값의 크기의 메시지를 저장한다.

<정리 2> 원본 Wc에 대해서 함수 ASf_{Mj}를 적용했을 때

$Cf(w_c, f_j) = k_{Mj}$ & $f_j = ASf_{Mj}$이고 $Cf(w_c, f_{MI}) = k_{MI}$ 이면 $k_{Mj} \geq k_{MI}$이다.

표 7 Capacity 함수와 Size 함수

함수이름	함수매핑	함수표현	설명
Capacity	$Cf : (W,F) \rightarrow N$	$Cf(w_c, f_j) = k_{Mj}$	k_{Mj}은 자연수, $wc \in W$는 원본의 집합, $f_j \in F = \{ f_{Mi}, Af_{Mi}, Sf_{Mi}, ASf_{Mi} \}$
Size	$f_{SW} : W \rightarrow N$	$f_{SW}(w_c) = N_w$	Nw는 Header 부분을 제외한 데이타 청크의 크기

ASf_{Mi} 의 알고리즘

```
Procedure Message_Embeddingn();
begin
   Cover-data read;
   for( p=1; p<=i ; p++)
      begin
         for( j=0; j<p; j++)
           begin
              Read Insertion bit of Message data;
              Loaded Message data overwrite  into j bits  position of cover-data;
           end;
       end;
   for( p=i-1 p<1 ; p--)
      begin
         for( j=0; j<p ; j++)
           begin
              Read Insertion bit of Message data;
              Loaded Message data overwrite  into j bits  position of cover-data;
         end;
       end;
end;
```

[증명] 정리 1에 의해서 $Cf(W_c, f_{M1}) = k_{M1}$이고, k_{M1} = Nwc/ 16이다.

i = 1일 때, $Cf(W_c, ASf_{M1}) = k_{M1}$이며 16비트에 1비트씩 삽입하면서 사인곡선 형식을 나타내는 것이므로 결국 $Cf(W_c, ASf_{M1}) = k_{M1}$와 같게 된다. 따라서 $k_{M1} = k_{M1}$이 된다. i = 2일 때, 정리 1에 의해서 $Cf(W_c, ASf_{M2}) = k_{M2}$이고 16비트마다 1비트 또는 2비트씩 삽입하므로 평균 1.5비트가 된다. k_{M2} =Nwc / 16* 1.5이고 k_{M1} =Nwc / 16이므로 $k_{M2} \geq k_{M1}$가 된다.

따라서 j ≥3일 때, k_{Mj} = Nwc / (16* (Σi) / 2) ≥ k_{M1}가 된다.

그러나 비밀 메시지가 하위비트를 기준으로 삽입되기 때문에 사인곡선의 형태로 비밀 메시지를 삽입하더라도 마지막 비트에 숨겨진 비밀 메시지는 공격자에 의해 필터링 될 수 있다. 이러한 문제점을 개선하기 위해 실제 DME(Dynamic Message Embedding) 모듈에서는 웨이브 파일의 16비트 단위를 10진수로 변환한 값을 기준으로 원본의 특성을 유지할 수 있는 범위 내에서 임계치(δ)를 지정하여 임계치 위치를 기준으로 비밀 메시지를 삽입한다.

하위비트를 기준으로 적용한 은닉함수 ASf_{Mi}를 적용한 ASS_{M4}와 16비트 단위를 10진수로 변환한 값이 임계치 이상이 되는 지점을 기준으로 비밀 메시지를 삽입하는 δASS_{M1}과 δASS_{M4}가 원본과 얼마나 유사한지를 분석하였다. 분석결과 이러한 함수들에 의해 생성된 스테고 데이타들이 원본과 유사하였다.

4. StegoWaveK 모델 설계 및 구현

3장에서 설명한 사인 누적 함수를 이용하여 임의의

표 8 원본과 δASS_{M1}와의 F-검정

	원본	δASS_{M1}
평균	37.4543333	37.288333
분산	10009986.3	10007245
관측수	3000	3000
자유도	2999	2999
F 비	1.00027390	
P(F<=f) 단측 검정	0.49700857	
F 기각치: 단측 검정	1.06192277	

표 9 원본과 δASS_{M4}와의 F-검정

	원본	δASS_{M4}
평균	37.4543333	37.5026667
분산	10009986.3	11110835.1
관측수	3000	3000
자유도	2999	2999
F 비	0.90092115	
P(F<=f) 단측 검정	0.49907389	
F 기각치: 단측 검정	0.94168806	

표 10 원본과 ASS_{M4}와의 F-검정

	원본	ASS_{M4}
평균	37.4543333	39.156333
분산	10009986.3	10812130
관측수	3000	3000
자유도	2999	2999
F 비	0.92581076	
P(F<=f) 단측 검정	0.49766103	
F 기각치: 단측 검정	0.94168806	

변수	N	평균	표준편차	합
원본	3000	37.454333	57.77347749	112363
ASS_{M4}	3000	39.156333	60.04369327	117469
δASS_{M1}	3000	37.288333	57.76556679	111865
δASS_{M4}	3000	37.502667	60.86745135	112508

단순통계

변수	최소값	최대값
Origin	-9934	9486
ASS_{M4}	-9382	9502
δASS_{M1}	9930	9488
δASS_{M4}	9930	9494

피어슨 상관계수 N-3000

	ASS_{M4}	δASS_{M1}	δASS_{M4}
원본	1.0000	1.0000	1.0000
	<0.0001	<0.0001	<0.0001

그림 10 원본와 다양한 스테고 데이타의 상관분석

위치에 메시지를 은닉하는 StegoWaveK 모델을 설계 구현하였다. 제안하는 오디오 스테가노그라피 모델인 StegoWaveK는 능동적으로 메시지를 은닉하는 DME 모듈을 설계하고 구현하였다. DME는 원본와 비밀 메시지의 특성을 분석하여 원본의 특성을 벗어나지 않는 범위내에서 원본에 16비트당 은닉되는 비트수와 은닉되는 위치를 분석하여 가장 알맞은 은닉함수를 선택하여 비밀 메시지를 은닉할 수 있도록 구현되었다.

그림 11은 압축, 암호화, 그리고 마지막으로 원본에 비밀 메시지를 삽입하는 단계로 구성되는 StegoWaveK 오디오 스테가노그라피 모델의 흐름도이다. 암호화는 사용자가 원하는 방법을 선택할 수 있다. 이러한 Stego-WaveK 모델에서 한글문서, 워드문서, 플래시 파일, 모든 그림 파일, 설계도면, 웨이브 파일 등을 비밀 메시지로이용하여 은닉한 후 정보손실 없이 추출하여 사용하였다. 상용화된 시스템의 경우도 같은 결과가 나왔다.

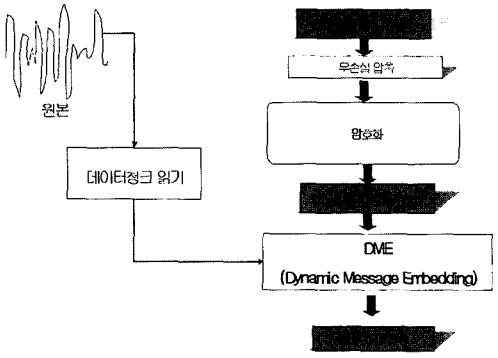


그림 11 StegoWaveK 모델의 흐름도

4.1 DME(Dynamic Message Embedding) 모듈

원본의 특성을 유지하는 범위 내에서 비밀 메시지를 하위 비트가 아닌 특정한 임계치에 삽입하도록 구현하는 모듈이다. 임계 치는 원본과 비밀 메시지의 특성을 분석하여 처리하도록 되어 있다. 또한 비밀 메시지를 삽입하는 알고리즘도 1비트 은닉하는 알고리즘부터 사인 곡선 형태로 숨길 정보의 양을 크게 할 수 있는 알고리즘 중 가장 적절한 것을 선택하여 은닉이 수행되도록 구현되어 있다. 그림 12는 비밀 메시지를 원본에 삽입하기 전에 수행하는 전처리 과정이다.

전처리 과정에서는 웨이브파일인 원본의 음량을 분석한 후 구간별로 웨이브파일의 음량 분포를 조사한다. 그 다음 비밀 메시지와의 비율을 참고해서 비밀 메시지가 숨겨질 수 있는 임계 치를 결정한다. 비밀 메시지가 은닉될 위치가 결정되면 메시지 은닉 알고리즘이 결정된다. 메시지 정보는 키값, 선택된 알고리즘, 메시지 크기, 저장이 시작된 위치 등에 대한 정보를 원본 파일에 저장하며, 메시지 추출시에도 이러한 정보들이 메시지의 정확한 추출에 필요한 기준이 된다. 그림 13과 그림 14는 실제 비밀 메시지를 은닉하고 추출하는 절차도이다.

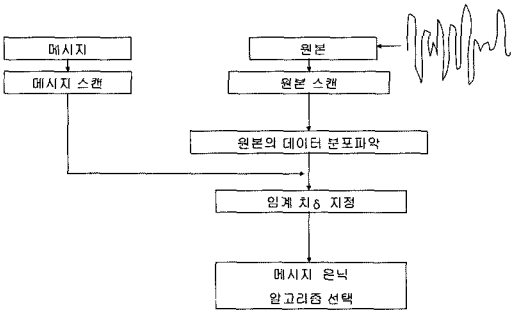


그림 12 전처리 과정

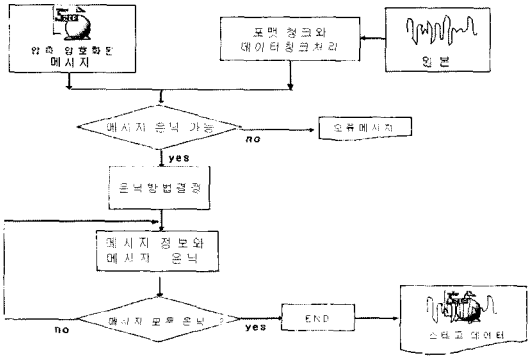


그림 13 메시지 은닉 흐름도

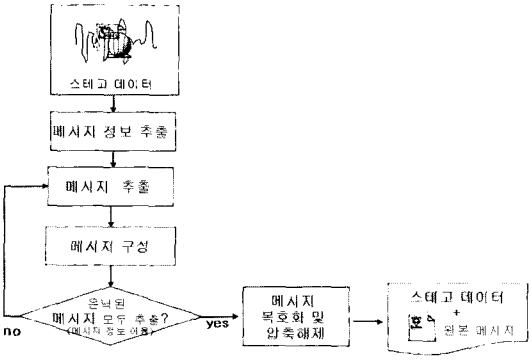


그림 14 메시지 추출 흐름도

4.2 CUR(Chunk Unit Read) 모듈

기본적으로 웨이브 파일의 헤더부분인 포맷 청크(Fmt Chunk) 크기는 18바이트이다. 이중 마지막 2바이트는 확장 정보의 크기값을 표현하지만 확장 정보를 포함하지 않는 경우라면 2바이트를 제외한 16바이트를 포맷 청크로 설계한다. 실제로 포맷 청크 크기를 16바이트로 설계해서 사용하는 경우가 더 많다. 상용화된 오디오 스테가노그라피에서는 웨이브 파일의 포맷 청크가 16바이트라는 전제하에 지정된 위치 즉 37번째 바이트부터 4바이트 값에 데이타 청크 구분자가 있어야만 정상적인 웨이브 형식으로 인지하도록 되어 있다. 그렇기 때문에 포맷 청크의 크기가 16바이트로 지정되어 있는 웨이브 파일만 웨이브 파일로 인식하고 18바이트로 되어 있는 웨이브 파일은 인식하지 못하는 문제점을 가지고 있다. 특히 MP3로 되어 있는 음악파일들은 웨이브 파일 형식으로 변환하면 18바이트나 14바이트의 포맷 청크 크기를 가지는 경우가 많아 이 방법을 사용할 수 없는 경우가 많다. 이러한 문제점을 해결하기 위해서 웨이브 파일의 청크에 따라 처리해 주는 CUR(Chunk Unit Read) 모듈을 설계했다. CUR 모듈에서는 단순히 고정된 위치 값을 읽어서 데이타의 특성을 판단하는 것이 아니라 웨이브 파일을 청크에 따라서 처리한다. 따라서 기본 포맷인 18바이트 포맷 청크로 된 웨이브 파일뿐만 아니라, 16바이트, 14바이트 포맷 청크로 된 웨이브 파일도 원본으로 사용할 수 있다. 그림 15는 선택된 오디오인 원본의 포맷 청크 에 따라 처리하는 CUR 모듈의 순서도이다.

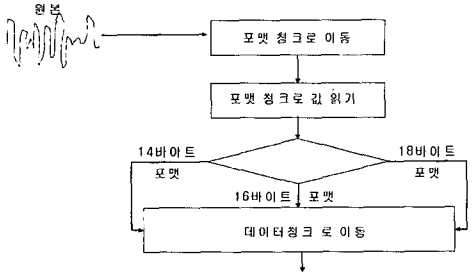


그림 15 데이타 청크

4.3 StegoWaveK 구현

4.1과 4.2에서 설명한 모듈들을 포함하여 VC++.Net으로 구현하였으며 사용자 측면에서는 크게 4단계로 이루어진다. 윈도우즈 98 이상의 환경에서는 버전과 종류에 관계없이 실행된다.

각 단계마다 찾아보기를 두어 사용자가 쉽게 필요한 데이타를 선택할 수 있으며, 윈도우즈를 사용하는 사용자라면 누구나 쉽게 자신의 정보를 은닉할 수 있다. 정보를 은닉에 사용된 원본과 스테고 데이타의 크기가 같고 특성도 유사하므로 청각적으로는 식별하기가 어렵다. 또한 은닉된 정보를 삭제하여도 필요할 때마다 스테고 데이타로 부터 정보를 추출하여 사용할 수 있다.

1 단계	원본 선택
2 단계	숨길 메시지 선택
3 단계	스테고 데이타 파일이름 결정
4 단계	원하는 암호화선택 후 실행

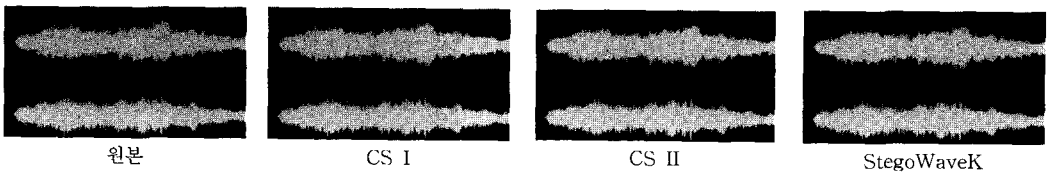
그림 16 StegoWaveK 사용자 인터페이스 Step 1

5. StegoWaveK 성능 평가

이 장에서는 VC++.Net으로 구현한 StegoWaveK와 실제 상용화되어 사용하고 있는 Invisible Secret 2002 (이후부터 CS I으로 표시)와 Steganos Security Suite 4(이후부터 CS II으로 표시)와 비교했다. [5]에 의하면 스테가노분석은 시각적, 청각적, 구조적, 그리고 통계적 방법을 이용하여 분석한다. 따라서 본 논문에서도 인간의 HVS(Human Visible System)측면, HAS (Human Auditory System) 측면, 통계적(Statical Analysis)측면, 그리고 오디오 측정(Audio Measurement) 측면으로 비교 분석하였다. 특히 HAS 부분은 다소 주관적일 수 있기 때문에 좀 더 객관적인 분석을 위해 스테고 데이타와 원본을 비교 분석하는 오디오 측정의 분석들을 추가하였다. 비교분석용 데이타는 유키쿠라모토의 피아노곡을 이용하였다. 다른 장르의 음악을 이용한 실험에서도 유사한 결과가 나왔다.

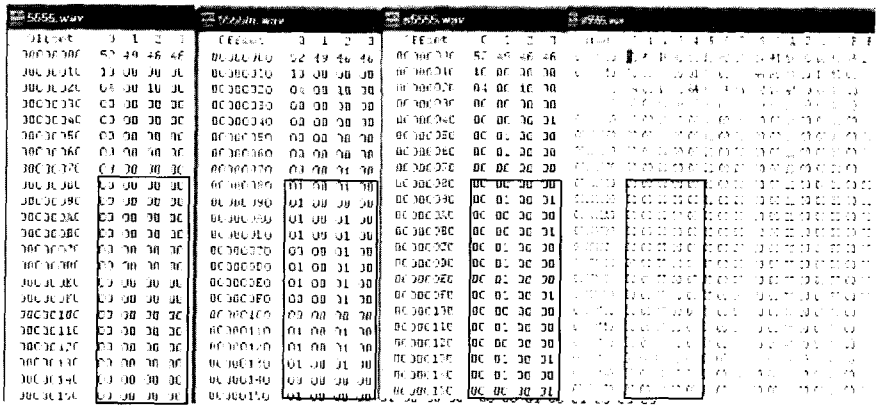
5.1 HVS(Human Visible System) 측면

상용화된 시스템 CS I, CS II과 본 논문에서 제안한 StegoWaveK 방법으로 만들어진 스테고 데이타를 오디오 편집툴인 쿨에디터를 이용하여 파형을 분석해보면 인간의 HVS 특성 때문에 시각적으로는 구분하기가 힘들다. 그림 17은 쿨에디터에서 스테고 데이타들의 파장을 캡처한 것이다.



원본 CS I CS II StegoWaveK

그림 17 원본과 오디오 스테가노그라피를 적용한 웨이브파일의 파장 비교



원본 CS I CS II StegoWaveK

그림 18 헥사 편집기를 이용한 코드보기

그림 17에서 제시된 바와 같이 시각적으로 Wave 파장을 보고 정보의 은닉 상태를 구분하기란 매우 힘들다. 제안한 StegoWaveK에서는 스테고 데이타로부터 메세지를 쉽게 얻어내지 못하게 하기 위해서 은닉함수 ASf_M를 적용하여 16비트 당 같은 수의 비트를 바꾸지 않고 그 값들을 사인 곡선 형태가 되도록 임계치 이상의 영역에서 메시지를 원본에 삽입하였다.

그림 18은 기존 시스템의 문제점으로 제시되었던 시각적 공격의 다른 방법인 헥사 편집기로 파일을 열어서 캡쳐한 것이다. 제안한 방법만이 대부분의 웨이브 파일의 앞부분이 0으로 되어 있는 것을 그대로 유지시켜 준다.

5.2 HAS(Human Auditory System) 측면

HAS(Human Auditory System) 측면에서 분석하기 위해 파일의 크기가 서로 다른 13개의 메시지와 4개의 웨이브 파일을 원본으로 선정하여 기존 시스템과 제안한 시스템을 비교 분석하였다. 하위비트 엔코딩을 사용하는 CS I과 CS II에 은닉된 스테고 데이타와 StegorWaveK 시스템을 통해 메시지가 은닉된 스테고 데이타를 100명의 학생들에게 들려주었다. 이는 다소 주관적인 판단에 의한 분석이기는 하나 대다수의 실험대상 학생들이 원본과 스테고 데이타 음악파일의 차이를 인식하지 못했다. 그림 19는 실험결과를 나타낸 그림이다.

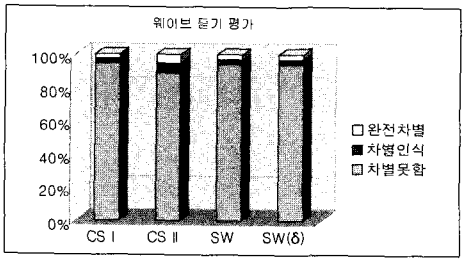


그림 19 시스템별 웨이브 청취결과

5.3 통계적 측면

다음은 원본과 CS I에서 생성된 스테고 데이타 그리고 StegoWaveK에 의해 비밀 메시지가 삽입된 두개의 스테고 데이타의 특징 값을 추출하여 원본과의 상관관계를 분석하였으며, 데이타가 모두 원본과 유사한 특성을 가지고 있음을 알 수 있다.

5.4 오디오측정 측면

오디오 측정 및 분석방식에는 주파수 대응(frequency response), 획득 또는 손실, 조화로운 왜곡(harmonic distortion), 모듈레이션간 왜곡(intermodulation

변수	N	평균	표준편차	합
원본	9144	-0.00949	0.47221	-86.77300
CS I	9144	-0.00964	0.47221	-88.17800
StegoWaveK	9144	-0.00965	0.47218	-90.21500

단순통계

변수	최소값	최대값
원본	-1.37000	2.10600
CS I	-1.37000	2.10600
StegoWaveK	-1.37000	2.10600

피어슨 상관계수 N=9144

	CS I	StegoWaveK
원본	0.99952	0.99952
	<0.0001	<0.0001

그림 20 원본과 다양한 스테고 데이타의 상관분석

표 11 CS I과의 F-검정

	원본	CS I
평균	-0.00964	-0.00965
분산	0.223003	0.222976
관측수	9143	9143
자유도	9143	9142
F 비	1.00012	
P(F<=f) 단측 검정	0.497721	
F 기각치: 단측 검정	1.35007	

표 12 StegoWaveK와의 F-검정

	원본	StegoWaveK
평균	-0.00964	-0.00988
분산	0.223003	0.221045
관측수	9143	9143
자유도	9143	9142
F 비	1.008857	
P(F<=f) 단측 검정	0.336674	
F 기각치: 단측 검정	1.035007	

distortion), 노이즈 수준(noise level), 위상 대응, 그리고 전송대응(transient response) 등이 있다. 이러한 변수들은 신호 레벨이나 위상, 그리고 주파수들을 포함하고 있다. 예를 들어 신호대잡음비인 SNR(Signal to Noise Ratio)은 로그나 데시벨(dB) 또는비율의 다양한 형태로 표현되는 레벨 측정방법이다. [4, 22, 23, 24]에서는 SNR을 이용하여 비밀 메시지가 은닉된 데이타 즉 스테고 데이타의 질을 측정하였다. SNR은 상대적인 값들을 나타내는 비율을 표현한다[2,14].

그림 21은 원본 그리고 CS I, CS II, 그리고 StegoWaveK을 이용하여 생성된 스테고 데이타와의 신호대 노이즈와의 비율을 나타낸 그래프이다. 제안된 시스템을 이용하여 생성된 스테고 데이타의 SNR이 CS I과는 차이가 나지 않으나 CS II와는 다소 차이가 난다.

그림 22는 음성의 품질을 평가해주는 PESQ(Perceptual Evaluation of Speech Quality)를 이용하여 원본과 각각의 스테고 데이타를 분석하였다. 실제 PESQ와 같은 자동화 음성 측정 결과를 100% 신뢰하기는 힘

표 13 분산분석 일원배치법 적용 결과

인자의 수준	관측수	합	평균	분산
원본	9143	-88.177	-0.00964	0.223003
CS I	9143	-88.218	-0.00965	0.222976
StegoWaveK	9143	-90.315	-0.00988	0.221045

분산 분석

변동의 요인	제곱합	자유도	제곱 평균	F 비	P-값	F 기각치
처리	0.00037	3	0.000123	0.000555	0.999982	2.605148
잔차	8136.622	36568	0.222507			
계	8136.623	36571				

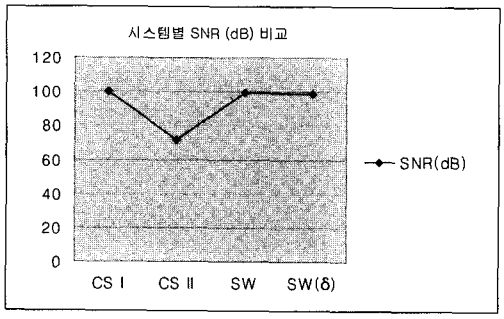


그림 21 원본과 각각의 스테고 데이타사이의 SNR 비교

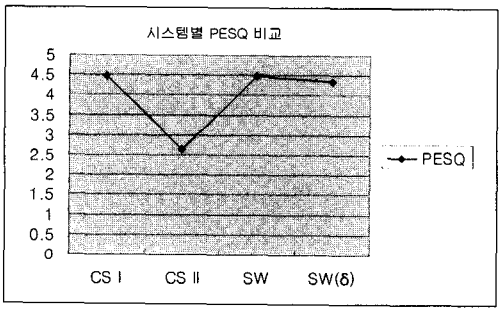


그림 22 원본과 각각의 스테고 데이타사이의 PESQ 비교

들지만 그 결과는 다양한 관련 테스트들의 기술에 사용될 만큼 충분한 신뢰도를 가지고 있다[25].

마지막으로 쿨에디터의 분석(Analyze) 메뉴의 웨이브 폼 통계(Wave form statistics)를 이용하여 원본과 각각의 스테고 데이타를 분석하였다.

6. 결 론

정보를 은닉하기 위해 응용되고 있는 오디오 스테가노그라피는 웨이브 파일을 원본으로 사용하며, 제안된 시스템에서 얻어진 스테고 데이타의 파일 크기도 같고 일반 청취자나 사용자는 음질의 차이가 나지 않아 스테고 데이타에 정보가 은닉되어 있다는 사실을 알 수 없다. 또한 간단한 오디오 편집 툴을 이용하여 분석한 파장은 인간의 HVS 시스템에서 직관적으로 분석할 수 없다. 뿐만 아니라 HAS 시스템에서도 분석이 불가능하기 때문에 비밀 정보 전달에 용이하게 응용될 수 있다.

상용화된 오디오 스테가노그라피는 원본으로 사용가능한 웨이브 파일의 종류에 제한이 있으며 원본의 하위 비트에 비밀 메시지를 1비트씩 삽입하기 때문에 쉽게 필터링 될 수 있다는 문제점과 1비트씩 삽입함으로 인해 비밀 메시지를 은닉하기 위해 사용되는 원본의 크기

표 14 원본과 각각의 스테고 데이타사이의 웨이브 폼 통계결과

	원본		CS I		CS II		StegoWaveK		StegoWaveK(δ)	
Right	Left	Right	Left	Right	Left	Right	Left		Right	
Min Sample Value:	-26050	-27052	-26050	-27052	26050	-27052	-26050	-27052	-26050	-27052
Max Sample Value:	25833	23125	25833	23125	25833	23125	25833	23125	25833	23125
Peak Amplitude(dB)	-1.99	-1.67	-1.99	-1.67	-1.99	-1.67	-1.99	-1.67	-1.99	-1.67
Possibly Clipped:	0	0	0	0	0	0	0	0	0	0
DC Offset:	-0.001	0	-0.001	0	0.001	0	-0.001	0	-0.001	0
Minimum RMS Power(dB)	**-97.9**	**-111.89**	**-92.8**	**-96.57**	**-68.58**	**-70.77**	**-97.9**	**-111.88**	**-97.86**	**-112.13**
Maximum RMS Power(dB)	-10.86	-10.61	-10.86	-10.61	-10.86	-10.61	-10.86	-10.61	-10.86	-10.61
Average RMS Power(dB)	-18.1	-18.84	-18.1	-18.84	-18.1	-18.84	-18.1	-18.84	-18.1	-18.84
Total RMS Power(dB)	-17.38	-18.13	-17.38	-18.13	-17.38	-18.13	-17.38	-18.13	-17.38	-18.13

가 커야 한다는 문제점을 내포하고 있다.

본 논문에서 제안한 StegoWaveK 모델은 기존의 상용화된 오디오 스테가노그라피가 가지고 있는 문제점을 개선하기 위해 제안된 한국형 모델이다.

첫째 웨이브 파일의 포맷 청크가 16바이트로 구성된 웨이브 파일에만 비밀 메시지를 은닉할 수 있다는 문제점을 개선하기 위해, 웨이브 파일을 청크 단위로 처리할 수 있도록 구현하였다.

둘째 원본이 숨길 수 있는 메시지의 크기를 향상시키며, 공격자에 의해 쉽게 필터링 될 수 있다는 문제점을 개선하기 위해 능동적으로 메시지를 은닉하는 DME 모듈을 설계하고 구현하여 적용하였다. DME 모듈은 원본과 비밀 메시지의 특성에 따라 비밀 메시지가 은닉되는 임계 치와 삽입 알고리즘을 선택할 수 있도록 구현되어 있다. 이는 제안한 모델이 비밀 키를 사용하는 스테가노그라피임에도 불구하고 공개 키 스테가노그라피와 같이 은닉된 정보의 유무를 구별하기가 힘든 효과를 나타낸다.

셋째 비밀 메시지를 은닉하기 전에 일반적으로 비밀 메시지의 특성을 가시적으로 확인할 수 없고 스테고 데이타와 원본으로부터 비밀 메시지를 얻었다고 해도 그 내용을 파악할 수 없도록 하기 위해 암호화를 수행한다.

넷째 윈도우 운영체제처럼 파일의 속성을 이용하여 단순한 파일 숨기기 형태로 가시적으로 보이지 않는 수준의 정보 은닉이 아니라 자주 사용하는 오디오 데이타에 정보를 은닉하고 은닉된 정보는 더 이상 컴퓨터에 존재하지 않더라도 원할 때마다 자신의 숨겨진 정보를 조금의 손실도 없이 추출해낼 수 있다. 따라서 주요한 설계도면이나 프로그램 파일, 극비 문서 등을 숨기는데 활용할 수 있다.

향후에는 임베디드 시스템으로의 전환과 관련된 연구가 필요하다. [26]에서 제시된 여러 가지 새로운 방법을 도입하여 변경하거나, TIFF 와 같이 성능이 좋은 무손실 압축 프로그램과도 연계하여 제안한 StegoWaveK 시스템의 성능을 향상시킬 수 있다. 좀 더 편리한 사용자 인터페이스도 추후 연구되어야할 영역이다. 이 시스템을 이용하여 멀티미디어 컨텐츠를 이용하는 사이버 교육에서 사용자에 대한 기본정보 및 평가시스템에서 활용할 수 있는 다양한 정보를 은닉하는 데 활용할 수 있다. 개인 정보를 다양한 수준으로 보안하는 다단계 개인정보 보호시스템으로도 활용가능하다.

참 고 문 헌

[1] J.Zollner, H.Federrath, H.Klimant, A.Pfitzmann, R.Piotraschke, A.Westfeld, G.Wicke, G.Wolf, "Modeling the security of steganographic systems," 2nd Workshop on Information Hiding, Portland, LNCS 1525, pp.345-355, Springer-Cerlag, April 1998.

[2] S.K. Pal, P.K. Saxena, S.K. Muttoo, "The Future of Audio Steganography," STEG'02, July 11-12, 2002.

[3] Neil F. Johnson, "Introduction to steganography Hidden Information," GMU 2001-Computer Crime Symposium, August 13-17, 2001, George Mason University, Fairfax, Virginia.

[4] Stefan Katzenbeisser, and Fabien A.P.Petitcolas "Information hiding techniques for steganography and digital watermarking," Artech House Publishers, 2000.

[5] Peter Wayner, "Disappearing cryptography Information Hiding : Steganography & Watermarking," second edition, chapter 17, Morgan Kauffman, 2002.

[6] http://www.cbcis.wustl.edu/~adpol/courses/cs502/project/report/node1htm

[7] http://www.neobytesolutions.com/invsecr/index.html

[8] http://www.steganos.com/./en/

[9] Fabien A.P. Petitcolas, Ross J. Anderson and Markys G.Kuhn, "Information Hiding-A Survey," Proceedings of the IEEE, special issue on protection of multimedia content, 87(7):1062-1078, July 1999.

[10] 김현곤, 원동호 외 12, "지적 재산권 보호를 위한 정보은닉 기술 및 표준화 연구", 한국 전산원, pp.19-41, 2000.

[11] Christian Cauchin, "An Information-Theoretic Model for Steganography," In Proceeding of 2nd Workshop on Information Hiding, Lecture Notes in Computer Science 1525, pp.306-318, Springer, 1998.

[12] J. Zollner, H. Federath, H. Klimant, and A. Pfitzmann, "Modeling the security of steganography systems," In Proceeding of 2nd Workshop on Information Hiding, Lecture Notes in Computer Science 1525, pp.344-354, Springer, 1998.

[13] Stefan Katzenbeisser and Fabien A.P. Petitcolas, "Defining Security in Steganographic Systems," SPIE Vol.4675, Security and Watermarking of Multimedia Contents IV, pp.260-268, 2002.

[14] Brian Chen and Gregory W. Wornell, "Quantization Index Modulation: A Class of Probably Good Methods for Digital Watermarking and Information Embedding," IEEE Transaction on Information Theory, Vol.47, NO.4, MAY 2001.

[15] RG van Schyndel, AZ Trikel, CF Osborne, "Digital Watermark," International Conference on Image Processing, Vol.2, pp.86-90, 1994.

[16] Djimitri Wiggert, "Codes for Error Control and Synchronization," Artech House Inc, 1988.
[17] BenZamin Arazi, "A Commonsense Approach to the Theory of Error Correcting Codes," MIT Press 1986.
[18] http://members.tripod.com/steganography/stego/software.html
[19] http://www.cs.uct.ac.za/courses/CS400W/NIS/papers99/dsellars/stego.html
[20] htttp://debut.cis.nctu.edu.tw/~yklee/Reaserch/Steganography/Qalter_Bender/IHW96.pdf
[21] http://www.rfdh.com/bas_com/2-10.htm
[22] Richard C. Cabot, Bruce Hofer, and Robert Met zler, "Chapter 13.1 Audio Measurement and Analy sis," 2002.
[23] Jerry Whitaker, Blair Benson, "Chapter 13 Stand ard Handbook of Audio and Radio Engineering," McGraw Hill Professional, 2002.
[24] J.D.Gordy and L.T.Bruton IEEE MWSCAS 2000, "Performance Evaluation of Digital Audio Water marking Algorithms."
[25] http://211.38.132.225/new_b24.htm
[26] Ira S. Moskowitz, Garth E. Longdon, and LiWu Chang, "A New paradigm Hidden Steganography," New Security Paradigms workshop 2000, Sep tember, 19th~21st, 2000, Cork Ireland.
[27] Ross J. Anderson, Fabieb A.P. Petitcolas, "On The Limits of Steganography," IEEE Journal of Selec ted Areas in Communication, 16(4):474-481, May 1998.
[28] GJ Simmoms, "The Prisoners' Problem and the Subliminal Channel," in Proceeding o CRYPTO '83, Plenum Press, pp.51 68, 1984.
[29] http://www ccrma.stanford.edu/CCRMA/Courses /422/projects/WaveFormat.
[30] RG van Schyndel, AZ Trikel, CF Osborne, "A Digital Watermark," International Conference on Image Processing, Vol.2, pp.86-90, 1994.

김 영 실

1989년 2월 고려대학교 정보통신대학 컴퓨터학과 졸업(이학사). 1991년 8월 고려대학교 대학원 정보통신대학 컴퓨터학과 졸업(이학석사). 1995년 3월 고려대학교 대학원 정보통신대학 컴퓨터학과 박사과정 수료. 1996년 3월~1998년 2월 용인송담대학 멀티미디어과 겸임교수. 1998년 3월~2001년 2월 대림대학 컴퓨터정보계열 전임강사. 2001년 3월~현재 대림대학 컴퓨터정보계열 조교수. 관심분야는 보안공학, 멀티미디어, 소프트웨어공학

김 영 미

1982년 동국대학교 통계학과(이학사) 1984년 동국대학교 대학원 통계학과(이학석사). 1988년 동국대학교 대학원 통계학과 박사과정 수료. 현재 (주)쎄스 암호화 연구실장. 관심분야는 암호화알고리즘, 정보은닉, 키분배알고리즘, 웹프로그램밍(asp, javascipt)

백 두 권

정보과학회논문지 : 컴퓨팅의 실제 제 9 권 제 2 호 참조