The Journal of GIS Association of Korea, Vol. 17, No. 1, pp. 51-66, April 2009

벡터 맵 데이터의 정확성과 위상을 고려한 디지털 워터마킹

김 정 엽*・박 수 홍**

Digital Watermarking of 2D Vector Map Data for the Accuracy and Topology of the Data

Junh-Yeop Kim*, Soo-Hong Park**

요 약

정보통신 기술의 발달과 더불어 여러 데이터를 디지털화함에 따라 데이터 소유자의 권리를 보호하고자 하는 저작권 보호에 대한 관심이 많아졌다. 디지털 워터마킹은 저작권 보호를 위한 강력한 방법 중에 하나이다. GIS에서 많이 사용되는 벡터 맵 데이터에 대한 저작권 보호를 위해 본 연구에서는 새로운 디지털 워터마킹 기법을 제안하고자 한다. 제안하는 방법은 CRC의 원리를 이용하여 워터마크를 삽입하고 검출하는 방법으로 실험 결과 여러 공격에서도 삽입한 워터마크를 검출하여 데이터 소유권을 보호할 수 있음을 알 수 있었다. 따라서 제안하는 방법은 벡터 맵 데이터의 소유권을 보호하기 위한 방법으로 활용 가능할 것으로 기대된다.

주요어 : 디지털 워터마킹, 벡터 데이터, CRC, 일방 함수

ABSTRACT : There have been concerned about the copyright as numerous data are digitalized because of the growth of performance of the computer and Internet. Digital watermarking is one of strong methods to protect copyright. We proposed a novel digital watermarking for vector map data. Although vector map data are used widely in GIS, there is little interest in copyright. The proposed method is to embed and extract watermarks using CRC principle. The experimental results show that this method can protect the copyright of the vector map by extracting embedded watermarks. Therefore, the proposed method can be utilized as the technique to protect vector map data.

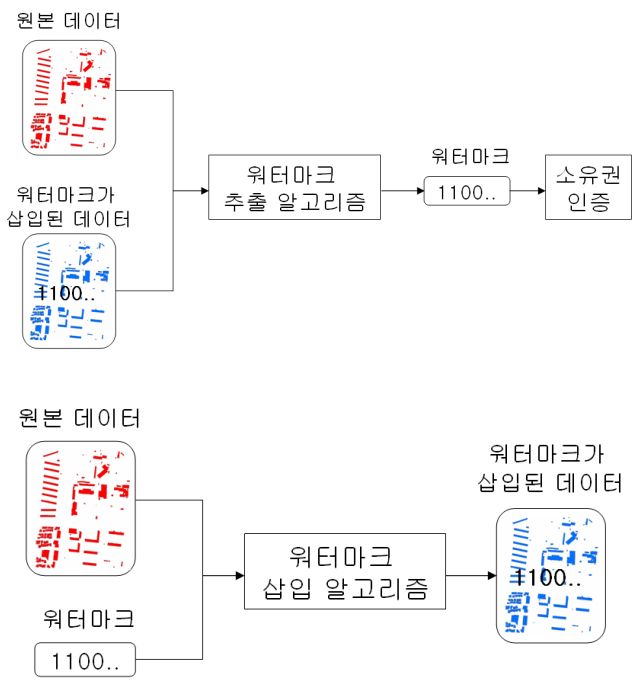
Keywords : Digital watermarking, vector data, CRC, one-way function

*인하대학교 박사 졸업
**인하대학교 지리정보공학과 부교수

1. 서 론

디지털의 시대인 오늘날에는 무수한 데이터가 생성이 되고, 복제가 되며, 배포가 되고 있다. 이러한 데이터들은 우리의 생활에 직·간접적으로 영향을 끼치고 있으며, 다양한 목적으로 활용되고 있다. 하지만, 언제 어디서나 필요한 데이터나 정보를 획득하고 다룰 수 있다는 것은 항상 불법 복제라는 위험도 같이 존재하게 된다(윤영주, 2004). 일반적으로 디지털 데이터의 제작은 많은 시간과 비용이 들지만 복제에는 짧은 시간에 비용도 거의 들지가 않는다. 따라서 이러한 행위는 사회의 커다란 문제점으로 부각되고 있다.

디지털 데이터의 보호를 위해 여러 가지 방법들이 있으나 그 중에서 디지털 워터마킹에 관한 관심이 최근에 많아지고 있다. 디지털 워터마킹이란 디지털 데이터에 소유권을 입증할 수 있는 부가적인 정보를 삽입하는 것을 말한다. 이러한 워터마킹 기법은 데이터가 불법 복제가 된 이후에도 소유권을 입증할 수 있기 때문에, 소유권 보호를 위한 방법으로 관심을 받고 있다(Yuanyuan Li and Luping Xu, 2003). [그림 1]은 워터마크를 추출할 때 원본 데이터가 필요한 non-blind 방식의 디지털 워터마킹 방법을 보여주고 있다. Non-blind 방식은 blind 방식과 달리, 삽입한 워터마크를 추출할 때 원본 데이터가 필요 없는 방법으로 워터마크 추출 과정이 보다 효율적이라고 할 수 있다. 반면에 non-blind 방식은 원본 데이터를 이용하기 때문에 다양한 공격에서 워터마크를 검출하기가 쉬운 장점을 지니고 있다.



[그림 1] 워터마크 삽입과 추출 과정

디지털 워터마킹은 기준에 따라 여러 분류로 나눌 수 있으나, 가장 대표적으로 주파수 영역에서 워터마크를 삽입하는 방식과 공간 영역에 워터마크를 삽입하는 방식으로 나눌 수 있다. 주파수 영역의 워터마킹은 여러 공격에 강인하나 왜곡 현상을 통제하는 것이 어려우며, 공간 영역의 워터마킹은 반대로 왜곡 현상을 통제하기 쉬운 장점을 지니고 있다. 디지털 워터마킹은 워터마크를 삽입한 이후에 워터마크를 검출할 수 있는 능력인 입력 효과성(Embedding Effectiveness), 워터마크를 삽입한 이후에 원본 데이터 질의 훼손 여부를 보는 충실도(Fidelity), 데이터에 다양한 처리를 한 후에도 워터마크를 검출할 수 있는 능력인 강인성(Robustness), 워터마크를 삽입하지 않은 데이터에서 워터마크가 검출되는 긍정적 오류율(False Positive Rate)의 특징을 가지고 있다(김현승, 2005).

디지털 워터마킹의 시작은 멀티미디어 파일의 소유권 보호를 목적으로 1990년대 중반 이후부터 관심을 받아왔으나, 벡터 데이터에 관한 연구는 현재까지 많이 부족한 상태이다. 벡터 데이터는 네비게이션이나 웹 맵 서비스 등 GIS의 많은 분야에서 사용되고 있다. 하지만, 벡터 데이터의 생산과 유지에는 많은 시간과 비용이 소요되며, 이러한 이유로 일반적인 다른 데이터에 비해 더 높은 가치를 지니고 있다(Itaru Kitamura et al, 2001; Michael Voigt and Christoph Busch, 2003; Michael Voigt et, 2004; Lopez Carlos, 2002). 따라서 벡터 데이터를 무료로 사용하기에는 무리가 있으며, 소유권 보호와 보안을 위한 벡터 맵 데이터 디지털 워터마킹이 필요하다(Ryutarou Ohbuchi et al, 2003; Ryutarou Ohbuchi et al, 2002).

벡터 맵 데이터를 위한 워터마킹을 구현하기 위해서는 벡터 맵 데이터의 특징을 고려해야 한다. 특히, 벡터 데이터의 유효성을 훼손시키지 않기 위해 워터마크를 삽입한 이후에도 위상관계의 변화가 없어야 한다. 본 연구는 이러한 점을 고려한 새로운 벡터 맵 디지털 워터마킹 방법을 제안하는 것을 연구의 목적으로 한다.

본 연구의 방법은 워터마크 추출시 효율성을 고려하여 blind 방식의 워터마킹 방법으로 하였으며, 데이터의 전송시 에러 여부를 살펴보는 방식인 CRC(Cyclic Redundancy Check) 방식을 응용하여 워터마크를 삽입하였다. 실험은 기존의 연구들과 마찬가지로, 워터마크를 삽입한 데이터에 여러 공격을 가한 후에 워터마크 검출 여부를 판단하였다. 이러한 결과로 워터마킹의 강인성을 입증하고, 벡터 맵의 소유권을 보호할 수 있는 방법이 될 수 있는지의 가능성을 살펴보았다.

2. 관련 연구

지금까지의 디지털 워터마킹은 이미지, 오디오, 비디오와 같이 멀티미디어 데이터에서 많이 적용되어 왔다. 이에 반해 벡터 데이터에 대한 워터마킹 연구는 적은 편이다. 기존의 벡터 데이터 워터마킹 연구를 살펴보면 크게 주파수 영역에서의 방법과 공간 영역에서의 방법으로 구분 지을 수 있으며, 주파수 영역에서의 방법은 주로 DFT(Discrete Fourier Transform), DWT(Discrete Wavelet Transform), DCT(Discrete Cosine Transform)의 변환 방법을 이용하였다(Jungyeop and Soohong, 2007).

Vassilios 외(2000)와 Itaru 외(2001)는 좌표

를 complex value(z(n) = x(n) + iy(n))로 바꾸고 DFT 변환을 통해 워터마크를 삽입하도록 하였다. Vassilios 외(2000)는 blind 방식이며, Itaru 외(2001)는 non-blind 방식으로 방법상에서는 큰 차이가 없었다. 두 방법 모두 특정한 공격에서만 삽입한 워터마크가 검출이 되었으며, GIS에서 자주 사용되는 단순화나 잘라내기 등의 공격에는 약함을 보여주었다. 그리고 워터마크를 삽입하면 원본 데이터와 많은 차이를 보여주었다. Yuanyuan과 Luping (2003)은 원본이 필요 없는 blind 방식의 워터마킹 알고리즘을 제안하였다. 이 연구도 마찬가지로 좌표를 complex value로 바꾸고 워터마킹을 하였다. 이전의 방법과 유사하나, 이 방법은 그 값을 웨이블릿(wavelet) 변환을 하여 DWT의 계수를 구하고, 계수에 워터마크를 삽입하도록 하였다. 워터마크 검출은 삽입할 때의 계산 값과 추출할 때의 계산 값을 서로 비교함으로써 NC(Normalized correlation)을 구하고, 그 값을 특정 임계치(threshold)와 비교하여 워터마크가 삽입이 됐는지 여부를 판단하였다. 실험 결과 여러 공격에 대한 워터마크 추출은 가능하였으나, 워터마크를 삽입하였을 때, 원본 데이터와 차이점이 발생하는 문제점을 보였다. Michael 외(2004)는 원본이 필요 없는 blind 방식의 워터마킹 기법으로 DCT 방법을 이용해 워터마크를 삽입하였다. 가까이 있는 좌표들은 유사한 특성을 갖는다는 점을 이용해 계수의 상관관계를 비교하여 워터마크를 삽입하는 방법을 소개하였다. 이 방법 또한 기존의 워터마킹 방법과 마찬가지로 워터마크를 삽입하였을 때, 원본 데이터와 워터마크가 삽입된 데이터와의 차이점이 많이 나타났으며, 단순화나 좌표의 추가/삭제 같은 공격에는 워터마크 검출이 어려웠다.

이와는 달리 공간 영역에서의 워터마킹 연구 또한 있어 왔다. 공간 영역에서는 주파수 영역보다 여러 공격에 대해 강인하지 못하다고 알려져 있으나, 이는 알고리즘의 개선으로 충분히 극복할 수 있으며, 공간 영역에서의 방법은 워터마크를 삽입함으로써 발생하는 왜곡현상을 통제하기가 쉽다는 장점이 있다.

Ryutarou 외(2002)는 특정한 값의 개수보다 크면서 각 영역(rectangular) 안의 좌표의 개수가 유사하게 영역을 나누고 구역 내의 좌표에 PSNR(Pseudo Random Number Sequence)를 이용하여 워터마크를 삽입하도록 하였다. 제안한 방법은 원본이 필요한 non-blind 방식이며, 다양한 공격에서 워터마크 추출의 결과가 좋았다. 하지만 이 방법은 워터마크 삽입 후에 데이터의 변형이 일어났으며, 위상관계 또한 훼손되는 단점이 있었다. Hwanil 외(2001)는 우선, 이미지를 같은 크기의 블록으로 나누고, 블록 안에 있는 좌표들을 새로운 좌표로 바꾸는 방법을 제안하였다. 이 방법은 원본이 필요 없는 blind 방식의 방법이며, 실험결과 회전이나 원점 이동 같은 공격에는 약하고, 워터마크를 삽입한 후, 원본 데이터와 비교하였을 때 많은 왜곡 현상이 나타났다. 따라서 벡터 맵 데이터에 사용하기에는 부족함을 보여주었다.

Michael과 Christoph(2003)는 워터마크 삽입을 위해 소유자만이 알고 있는 크기로 원본 데이터를 그리드로 분할하고, 워터마크 키에 의해 구성요소를 선택한다. 이렇게 선택된 구성요소는 패치로 구분되고, 두 패치 간의 통계적인 값을 비교하여 워터마크를 삽입하는 방법을 제안하였다. 이 방법은 워

터마크를 검출할 때 통계적인 방법을 사용하기 때문에 좌표의 개수에 많이 좌우되는 한계점을 지니고 있다. Chengyong 외(2006)는 더글라스-푸케 알고리즘에 의한 포인트 집합을 추출하고, cover data를 만들었다. Cover data는 두 개의 집합으로 나누고, 하나의 집합은 변화를 주지 않고, 다른 하나의 집합에는 거리의 값에 변화를 준다. 두 집합의 통계치가 비슷하게 분할을 하고, 그 데이터를 다시 좌표로 바꾼다. 이 방법은 워터마크를 삽입하였을 때, 데이터의 변화가 적다는 장점이 있지만, 더글라스-푸케가 아닌 다른 방법의 단순화 알고리즘에는 약하게 되며, 잘라내기 공격에도 약하다는 단점이 있다.

이와 같이 기존 연구들을 살펴보면 두 가지 문제점을 지니고 있다. 첫 번째는 GIS에서 사용하게 될 다양한 조작에 대해 워터마크 검출이 이루어지지 않는다는 것이다. 따라서 벡터 맵 데이터로 활용을 하기에는 부족함이 있다. 두 번째는 워터마크를 삽입하게 되면, 원본 데이터와 비교할 때 왜곡 현상이 나타났다. 즉, 워터마크를 삽입하면 위상관계가 달라지거나 객체의 모양이 틀어지는 경우가 발생하였다. 벡터 맵 데이터는 위치 정확도뿐만 아니라 위상관계도 매우 중요한 정보이기 때문에 기존의 방법을 벡터 맵 데이터를 위한 방법으로 사용하기에는 무리가 있다. [표 1]은 기존 워터마킹 연구들의 방법을 분류하고, 실험에서 수행하였던 공격의 종류와 그에 따른 워터마크 검출 여

<표 1> 기존연구의 특징

논문	도메인	원본 사용 여부	검출시 전처리	공격 종류						
				회전공격 원점이동	일반화	좌표 추가	데이터 재배치	노이즈 첨가	잘라 내기	위상 관계 고려 여부
Yuanyuan과 Luping (2003)	DWT	x	x	o	x	x	x	o	x	x
Vassilios et al (2000)	DFT	x	x	o	o	o	o	o	x	x
Itaru et al (2001)	DFT	o	x	o	o	o	o	o	o	x
Michael et al (2004)	DCT	x	x	-	-	-	-	-	-	x
Ryuatarou et al (2002)	spatial	o	o	o	o	o	o	o	o	x
Hwanil et al (2001)	spatial	x	x	x	-	-	o	o	x	x
Michael과 Christoph (2003)	spatial	x	x	o	o	o	o	o	o	x
Chengyong et al (2006)	spatial	x	x	x	o	o	o	x	x	x

부를 보여주고 있다. 대부분의 연구가 모든 공격에서 강인함을 보여주지 못하고 있으며, 특히, 워터마크 삽입시 벡터 맵 데이터에서 중요한 부분인 위상 관계를 고려하지 않고 있음을 알 수 있다.

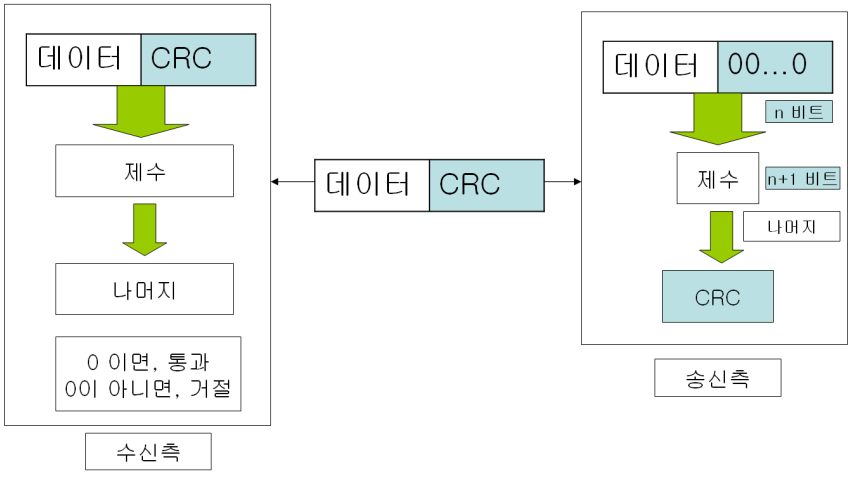
3. 벡터 맵 디지털 워터마킹의 구현

3.1 CRC

CRC 오류 체크 방법은 데이터의 오류 검사를 하는데 있어서 대표적인 에러 검출 방법 중의 하나로 워터마크 생성을 위한 좋은 적용 방법이 될 수 있다. CRC는 기본적으로 데이터 통신에 있어서 전송되어온 데이터 내에 에러가 있는지 확인하는 방법 중의 하나이며, 2진 나눗셈을 기반으로 한다(김한규 외, 2004).

[그림 2]는 CRC의 간단한 원리를 설명하는 것으로 우선, 데이터의 끝에 n 비트가 삽입이 된다. 두 번째로 2진 나눗셈이라 불리는 과정을 이용하여 n+1 비트의 제수로 나누며, 이 나눗셈으로부터 얻어지는 나머지가 CRC가 된다. 마지막으로 n 비트를 나눗셈에서 CRC로 대체한다. 이렇게 덧붙여진 데이터는 수신측에서 같은 제수로 나누어 나머지가 0인지 아닌지를 판단하여 오류 여부를 판단하는 것이다.

본 연구에서는 이와 같은 원리를 이용하여 좌표를 2진 데이터로 바꾼 후에 소수점 이하의 일정한 크기의 비트수를 데이터로 하고, 사용자가 주어진 값을 제수로 하여 나온 CRC를 워터마크로 간주하였다. 계산된 워터마크는 좌표에 반영을 하여 워터마크가 삽입된 데이터를 생성하도록 하였다. 이렇게 CRC 방법을 이용함으로써 워터마크 생성을 보다 간단하게 할 수 있다. 하지만 워터마크 생성이 간단하더라도, 일방 함수의 특성상 어느 위치에 생성이 되며, 워터마크의 크기를 아는 것은 어렵다. 따라서 삽입한 워터마크를 훼손하려는 공격에는 강인할 수가 있다.



[그림 2] CRC 발생기와 검사기

3.2 교차테스트

원본 데이터에 워터마크를 삽입하면 크던 작던 좌표의 변화는 일어나게 된다. 이상적인 방법은 워터마크를 삽입해도 좌표의 변화가 없는 경우이지만, 현실적으로는 불가능하다. 따라서 벡터 맵 데이터의 허용오차 범위 내에서 좌표의 변화까지를 인정할 수 있다. 하지만, 좌표의 이동으로 인해 위상관계가 변한다면 이는 벡터 맵 데이터의 유효성을 훼손시키는 것이 된다. 따라서 이러한 것을 방지하기 위하여 원본 데이터와 워터마크가 삽입된 데이터의 교차 테스트를 통해 위상관계의 훼손 여부를 살펴볼 필요가 있다. 만약 워터마크 삽입 후에 위상관계가 변하게 되면 그 좌표에는 워터마크를 삽입하지 않도록 해야 한다. 이러한 교차테스트는 위상관계를 유지시켜줄 뿐만 아니라, 워터마크 삽입 후 벡터 맵 데이터에서 객체들의 모양 유지에도 필요한 과정이 된다. 연구에서는 이러한 교차테스트를 통해 벡터 맵 데이터의 유효성을 유지시켰으며, 이는 기존의 벡터 데이터를 위한 연구들과의 가장 큰 차이점이 된다.

이와 같은 교차 테스트는 원본 데이터에서 두 선분이 교차하는 경우, 교차하지 않는 경우, 한 선분의 끝점이 다른 선분에 포함되는 경우를 각각 테스트하며 리턴 값을 가지도록 하였다. 그리고 워터마크를 삽입한 데이터에서 교차테스트를 하여 리턴 값이 달라지면, 위상관계가 훼손된 것으로 간주하여, 그 선분을 표현하는 좌표에는 워터마크를 삽입하지 않도록 하였다.

기본적으로 연구에서 제안하는 방법들은 워터마크 삽입으로 인해 위상관계나 데이터의 정확성을 떨어뜨리지 않도록 하였다. 즉, 소수점 이하에서 워터마크를 삽입하도록 하였기 때문에, 벡터 맵 데이터의 허용오차에 들어오도록 하였다. 하지만 리턴 값이 달라지는 경우에 대하여 고려해야 할 사항이 있다. 워터마크 삽입이 매우 작은 부분에서 이루어져서 사실상 눈으로 구분이 안가거나, 벡터 맵 데이터의 허용오차 내에서의 변화로 인해 교차 테스트 결과 리턴 값이 달라지는 경우가 있다.

3.3 제안한 워터마킹 알고리즘

제안한 방법은 blind 방식으로 워터마크를 검출할 때는 원본 데이터가 필요 없으나, 다만 전처리 과정에서만 원본 데이터를 활용하게 된다. 본 연구의 방법은 기존의 방법들과 달리 3.1절에서 설명한 CRC라는 개념을 워터마크에 접목시켰으며, 위상관계의 유지를 위해 교차테스트를 하였다. 이러한 접근법을 통해 보다 강인한 워터마킹 방법을 구현함과 동시에 충실도 또한 만족시키도록 하였다. 그리고 제안하는 방법의 특징은 랜덤 테이블을 통하여 워터마크를 삽입하고, 검출할 수 있으며, 거래 추적이 가능한 방법이 될 수 있도록 하였다. 본 연구의 워터마킹 알고리즘은 우선 워터마크를 삽입하기 위하여 랜덤 테이블을 생성한다. 랜덤 테이블은 향후에 워터마크 키로 활용될 수 있으며, 데이터마다 다른 랜덤 테이블을 적용할 수도 있다. 예를 들어, 표 1과 같이 소유자가 13이라는 값으로 랜덤 변수 범위를 지정하면, 난수는 0부터 13까지 발생할 수 있다. 이때, 0부터 9의 숫자는 0부터 13의 범위를 갖는 난수로 대체하게 된다. 새로운 난수로 대

<표 2> 랜덤 테이블의 예

실제값	0	1	2	3	4	5	6	7	8	9
변환값	5	6	11	7	0	2	12	5	8	3

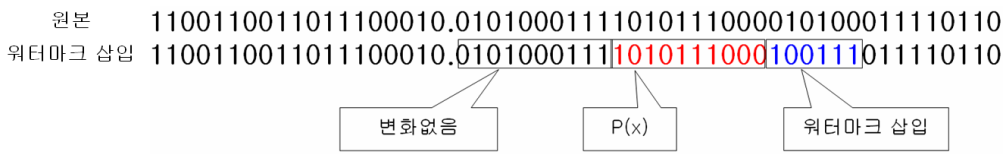
체할 때는 실제 좌표의 정수 부분만을 이용하여 변환한다. 즉, 실제 좌표가 236821.75라면, 117128115.75라는 값으로 바꿀 수 있다. 이렇게 바뀐 새로운 값은 워터마크 삽입을 위한 계산에서 사용되어 지는 것이며, 실제 좌표를 변경하여 저장하는 것이 아니다. 따라서 다시 원래 좌표 값으로 되돌릴 필요는 없으며, 워터마크 추출을 위해 데이터의 소유자는 13이라는 난수 발생 범위와 랜덤 테이블을 가지고 있으면 된다.

이렇게 랜덤 테이블을 통해 원본 데이터의 좌표들을 모두 변환하다. 그리고 사용자의 고유값을 입력받아 그 값을 이용하여 워터마크가 삽입될 위치를 결정하게 된다. 이렇게 사용자의 고유 값을 이용하면 향후에 워터마크를 검출하여 저작권을 증명하려 할 때, 어떤 사용자의 실수로 인해 불법적인 배포가 되었는지 증명할 수 있는 수단이 된다. 이는 거래추적에서 이용할 수 있는 근거가 될 수 있다. 예를 들어, 사용자 이름의 이니셜을 사용자의 고유값으로 사용할 수 있다. 만약, 홍길동이라는 사용자가 있다면, 홍길동의 이니셜은 HKD가 된다. 이 이니셜 HKD를 아스키코드로 변환시키면 H: 1001000(71), K: 1001011(74), D: 1000100(67) 값으로 바뀐다. 각각의 값은 모두 더해져서 212라는 사용자의 고유값으로 바꿀 수 있다. 이 고유 값은 랜덤 테이블과 함께 워터마크 삽입 과정과 검출 과정에서 반드시 필요한 값이 된다.

이렇게 랜덤 테이블을 이용한 새로운 좌표와 사용자 고유값을 구하면 고유값과 새로 바뀐 좌표의 차이를 2048(2^11)로 나눈다. 실험에서 사용하려는 shp 파일은 IEEE754의 부동소수점 표현 방식으로 저장이 되어 있다. 워터마크 삽입은 저장되는 방식을 따라 비트를 이용하여 이루어진다. 2048로 나눈 나머지는 CRC에서 G(x)가 되며 이를 이진수로 바꾸고, 그 이진수의 자릿수만큼 좌표의 소수점 이하에서 자리 이동을 한다. 이동을 하고 나 후의 다시 그 자릿수만큼을 십진수로 변환하여 그 수를 P(x)로 한다. 이렇게 P(x)와 G(x)를 구하게 되면, 변형된 CRC 방법으로 연산을 하여 최종적으로 워터마크를 삽입하게 된다. 변형된 CRC 방법은 다음과 같다. 이와 같이 CRC 방법을 변형시킨 이유는 일반적인 CRC 방법은 비트 연산을 통해 원래 데이터에 나머지 비트를 추가하는 방식이며, 나머지를 구하는 과정에서 앞에 비트가 0이 나오는 경우 처리해주는 과정이 따로 필요하다. 본 연구는 에러 검출이 목적이 아니므로 절차를 줄이기 위한 방법으로 식 (1)과 같이 변형을 하여 이용하였다.

$$G(x) - (P(x)\%G(x)) = \text{워터마크} \qquad \text{식 (1)}$$

이러한 워터마크 삽입 과정의 간단한 예를 들면 다음과 같다. 사용자 고유값이 212이고, 바뀐 좌표의 정수가 117128115일 경우, 두 값의 차이는 117127903이 되며, $117127903 \equiv 735 \bmod 2048$ 이 된다. 735(G(x))는 이진수로 $1011011111_{(2)}$이며, 자릿수는 10자리이다. 따라서 IEEE754 방식으로 표현된 좌표에서



[그림 3] 워터마크 삽입 예(G(x)가 735일 경우)

소수점을 표현하는 자리를 10자리 이동한 다음 그 뒤에 표현되어 있는 10자리의 비트인 $1010111000_{(2)}$(696)을 P(x)로 하고, $1011011111_{(2)}$(735)을 G(x)로 하여 변형된 CRC 방법으로 연산을 하면 워터마크는 $100111_{(2)}$이 생성된다. 이렇게 생성된 워터마크는 P(x)뒤에 삽입하는 것으로 워터마크 삽입 과정은 끝난다.

3.4 워터마크 검출

워터마크 추출은 기본적으로 삽입과정과 유사하다. 제안하는 방법은 워터마크 추출시에는 원본 데이터가 필요하지 않은 blind 방식의 디지털 워터마킹 방법이다. 하지만 워터마크 추출 전에 geometrical 변형에 대해서는 전처리 과정으로 원 상태로 돌려주는 과정이 필요하다. 벡터 데이터는 이미지 데이터보다 많은 장점을 가지고 있지만, 그 중의 하나는 geometrical 변형(회전, 원점 이동 등)이 있고 난 후에 원 상태로 돌리는 과정이 상대적으로 쉬우며, 데이터의 변화가 없다는 것이다. 이러한 장점을 이용하여 워터마크 검출에 앞서 geometrical 변형을 해준다.

이와 같이 공격당한 데이터에 대해 전처리 과정을 거친 후 원본과 같은 상태로 만들어지면, 워터마크 검출 과정이 시작된다. 워터마크를 삽입할 때 생성하였던 랜덤 테이블을 이용하여 공격당한 데이터의 좌표를 변형시킨다. 그리고 워터마크를 삽입할 때와 동일한 과정으로 고유 값과 변형된 좌표와의 차이를 통해 G(x)와 P(x)를 통해 워터마크를 계산하고, 이 값이 워터마크가 삽입될 위치에 있는 값과 일치하는지를 비교한다. 이와 같이 직접적인 비교를 통해 워터마크를 검출할 수 있으며, 이를 통해 궁극적으로 저작권을 인증하고 보호할 수 있다.

직접적인 비교를 통해 워터마크가 삽입이 되었는지 비교를 할 수도 있으며, 얼마나 많은 워터마크가 추출이 되는지 검출율을 통해 비교할 수도 있다. 검출율은 CR(Correspondence Ratio)을 통해 계산이 된다. 워터마크 삽입은 모든 좌표에 삽입을 하였으므로, 검출시에 사용한 데이터의 좌표 개수를 분모로 하고, 워터마크가 검출된 좌표의 개수를 분자로 하여 계산을 할 수 있다.

$$CR = \frac{\text{검출된 좌표의 개수}}{\text{데이터의 좌표의 개수}} \qquad \text{식 (2)}$$

4. 실험 및 분석

4.1 벡터 맵 데이터의 공격

성공적인 공격이란 워터마크가 삽입된 데이터를 그대로 사용할 수 있으면서 삽입된 워터마크는 제거되는 경우를 의미한다. 벡터 데이터는 일반적인 멀티미디어 데이터와 구

조가 다르기 때문에 이러한 공격들의 방법과 특징이 다르다. 벡터 데이터에 대한 공격들을 분류해 보면 크게 다음과 같이 분류할 수 있다(Niu, 2006). Geometrical 공격은 회전이나 원점 이동과 같은 공격들을 의미한다. 벡터 맵 데이터의 경우는 이러한 공격에 대해서 정보를 잃어버리는 것이 아니며 원래의 값으로 돌리는 것 또한 상대적으로 매우 쉽다. 따라서 이러한 공격이 벡터 맵 데이터에서는 매우 위협적인 공격이라고 할 수 없다. Vertex 공격은 좌표의 추가나 삭제가 일어나는 공격을 의미한다. 이러한 공격은 특히 벡터 맵 데이터의 일반화나 잘라내기에서 많이 일어나는 공격이다. 또한 맵 데이터의 갱신을 통해 특정 객체가 생성이 되거나 삭제되는 경우에도 vertex 공격으로 볼 수 있다. 이러한 공격들은 벡터 맵 데이터를 활용하는데 있어서 매우 빈번하게 일어나는 것이므로, 벡터 맵 데이터를 위한 디지털 워터마킹은 이러한 공격에 매우 강인해야 한다. 노이즈 공격은 벡터 맵 데이터에 노이즈를 첨가하여 삽입한 워터마크를 파괴하기 위한 공격이다. 하지만 벡터 맵 데이터는 좌표의 정확도를 가지고 현실을 표현하는 것이기 때문에 이러한 노이즈 첨가는 벡터 맵 데이터에 심각한 왜곡을 초래할 수 있다. 따라서 실질적으로 이러한 노이즈 공격은 빈번하게 일어나지 않으며 좋은 공격이라고 할 수 없다.

실험에서는 좌표의 이동과 회전, 단순화, 잘라내기, 좌표 추가에 대한 공격을 하였다. 노이즈 첨가는 이미지 데이터에서는 자주 있을 수 있는 공격이지만, 벡터 데이터에서는 좌표의 위치가 매우 중요한 정보로 작용을 하기 때문에, 실질적으로 벡터 데이터에서 노이즈 첨가라는 공격이 거의 일어나지 않는다(Niu, 2006). 따라서 본 연구에서는 노이즈 첨가 공격을 하지 않았다.

4.2 실험 결과

워터마크를 생성하기 위해서는 먼저 좌표를 다른 값으로 변경을 해주는 작업이 필요하며, 변경 작업은 랜덤 테이블을 통해 이루어진다. 랜덤 테이블을 이용하여 실제 좌표가 새로운 변환 값으로 변경되는 과정은 5.1에서 설명한 것과 같이 진행을 하였으며, 실험에서는 랜덤 테이블의 범위를 19로 하였다. 따라서 원본 데이터의 모든 좌표의 정수 자리는 0에서 19의 범위를 갖는 새로운 값으로 대체된다. 이렇게 변경된 새로운 정수 값은 사용자가 입력한 고유값을 이용하여 나머지 연산을 하게 된다. 실험에서는 고유값을 SIK라는 값을 이용하였다. S는 82, I는 72, K는 74라는 아스키코드 값을 가지며, 228이라는 값을 고유값으로 하게 된다. 변경된 좌표에서 이 고유값을 빼고 2의 11승(2048)로 나눈 나머지가 G(x)가 된다. 이 G(x)는 좌표마다 중복되는 경우도 있을 수 있으나 기본적으로 다른 값을 가지게 된다. P(x)는 G(x)의 비트 자릿수만큼을 원 좌표의 소수점 이하에서 이동을 한 후, 뒤이어 그 자릿수만큼의 비트를 P(x)로 한다.

워터마크 삽입을 위해 랜덤 테이블과 P(x), G(x)를 구하게 되면 변형된 CRC 방법을 통해 워터마크 삽입이 이루어진다. 워터마크 삽입은 데이터의 모든 좌표에서 이루어진다. 모든 좌표에 워터마크를 삽입한 후에는 교차 테스트를 하여 원본 데이터와 워터마크가 삽입된 데이터의 위상관계가 변하는지의 여부를 판단하여 워터마크를 삽입하였을 때

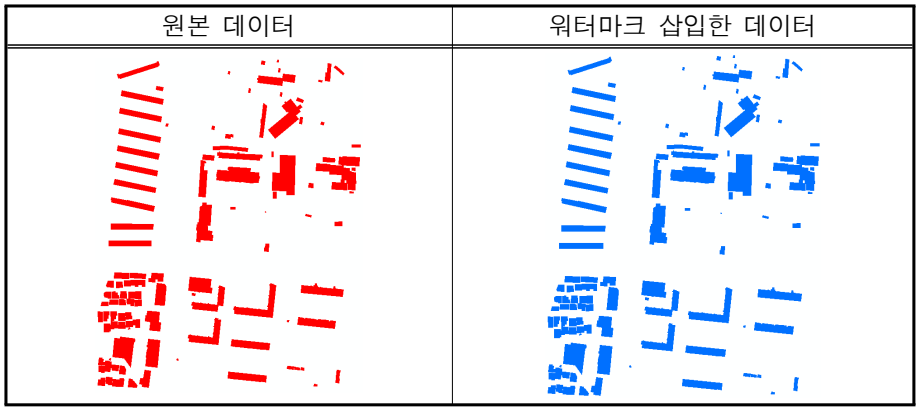
위상관계의 변화를 야기시키는 좌표에는 워터마크를 삽입하지 않도록 하였다.

실험을 통한 결과 워터마크를 삽입한 이후에도 위상관계가 변하는 경우는 발생하지 않았다. 이는 제안하는 알고리즘이 벡터 맵 데이터의 유효성을 유지시켜준다는 것을 의미한다. 제안한 방법에서는 또한 워터마크 삽입 후에 원본 데이터와의 왜곡 정도를 알아보기 위해 RMSE(Root Mean Squared Error)와 통계치를 계산하였다. 실험을 통한 결과를 살펴보면, 폴리곤 데이터의 RMSE 값은 0.0000으로 원본 데이터와 거의 차이가 없었다.

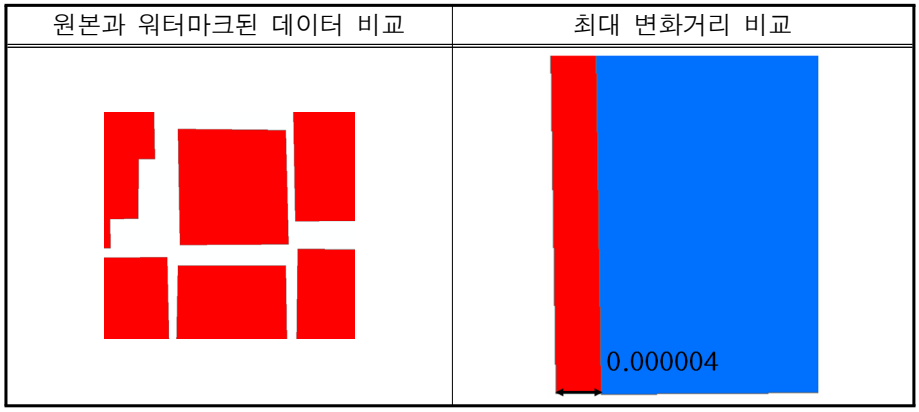
<표 3> 워터마크 삽입 이후의 변화량(폴리곤)

RMSE	0.0000
평균 좌표 변화거리	0.0000
최대 좌표 변화거리	0.0000
최소 좌표 변화거리	0.0000

그리고 워터마크 삽입 이후에 좌표의 평균 이동거리와 최대 이동거리, 최소 이동거리가 모두 0.0000으로 나타났다(표 3). 좌표의 변화거리는 허용 오차안에 들어오기 때문에 벡터 맵 데이터의 정확도에서는 문제가 되지 않는다.

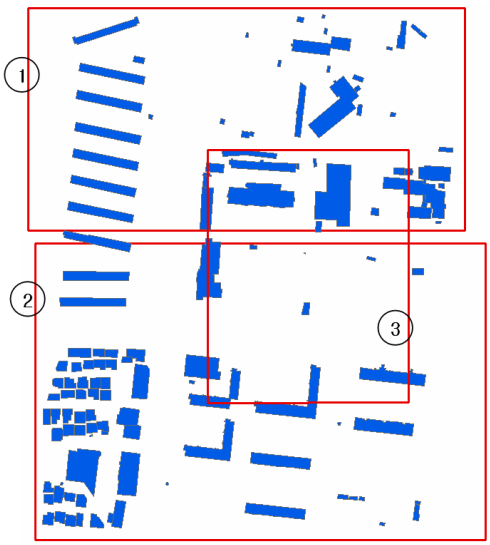


[그림 4] 원본 데이터와 워터마크를 삽입한 데이터(폴리곤)



[그림 5] 워터마크 삽입 후 최대 변화거리 확대(폴리곤)

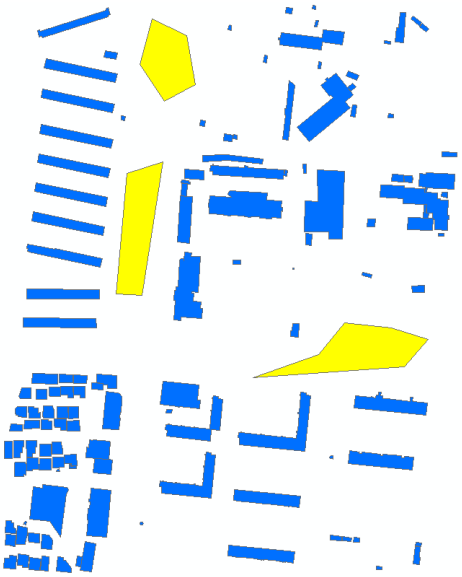
본 연구의 실험에서는 geometrical 공격으로 회전과 원점 이동 공격을 하였으며, vertex 공격으로는 잘라내기(그림 6)와 일반화(그림 7), 객체 추가(그림 8)의 공격을 하였다.



[그림 6] 실험에서 사용한 3가지 잘라내기

워터마크 검출을 하기 전에 vertex 공격에서는 전처리 과정이 필요 없지만, geometrical 공격에 대해서는 원본 데이터와 비교하여 원 상태로 맞춰준다. 이러한 전처리 과정이 끝나면 워터마크 검출을 시작하게 된다. 검출 과정은 우선 데이터의 소유자가 워터마크를 삽입할 때 사용하였던 랜덤 테이블을 필요로 한다. 즉, 공격당한 데이터에서 랜덤 테이블을 통해 워터마크 삽입 과정과 마찬가지로 좌표를 특정 값으로 변화시킨다. 그리고 워터마크 삽입 과정에서 사용하였던 사용자의 고유값을 이용하여 각 좌표마다 삽입할 워터마크와 삽입할 위치를 계산하고, 실제로 공격당한 데이터에서 그 위치에 계산된 워터마크가 있는지를 검출한다. 표 4는 원본 데이터와 워터마크가 삽입된 데이터의 좌표를 보여주고 있으며, 워터마크 검출은 삽입된 워터마크와 계산된 워터마크의 일치 여부를 통하여 이루어진다.

[그림 7] 일반화 공격을 한 데이터



[그림 8] 객체 추가 공격을 한 데이터

<표 4> 삽입한 워터마크(데이터 일부)

	원본 x 좌표
원본	01000001000010011111010001011010001111010111000010100011110101111
워터마크	01000001000010011111010001011010001111010111000010110100101001111
원본	01000001000010011111001110010000000000000000000000000000000000000
워터마크	01000001000010011111001110010000000000000000000110101100000000000000
원본	01000001000010011111010001010110110011001100110011001100110011001101
워터마크	01000001000010011111010001010110110011001100110110011000110011001101
	원본 y 좌표
원본	01000001000110110111100110101101101011100001010001111010111000001
워터마크	01000001000110110111100110101101101011100001010001111100110100001
원본	01000001000110110111100100100110011100001010001111010111000010100
워터마크	01000001000110110111100100100110011100001010001111011010110010100
원본	01000001000110110111100010011010111000010100011110101110000101000
워터마크	01000001000110110111100010011010111000010100011110101000101101000

<표 5> 공격당한 데이터의 워터마크 검출율

좌표	원점 이동	회전	일반화	객체 추가	잘라내기(1)
x좌표	1123/1123=100.0	1079/1123=96.1	734/734=100.0	1123/1144=98.2	389/389=100.0
y좌표	1123/1123=100.0	1090/1123=97.1	734/734=100.0	1123/1144=98.2	389/389=100.0

삽입한 워터마크는 이와 같이 직접 추출하여 확인이 가능하며, 또한 CR 값을 통해 얼마나 워터마크가 검출이 되었는지도 확인할 수 있다. 워터마크가 삽입이 된 데이터는 사용자가 그대로 사용을 할 수 있으나, 사용목적에 따라 사용자에 의해 변형이 가해질 수 있다. 따라서 실험에서는 이를 고려하여 회전, 원점 이동, 일반화, 객체 추가, 잘라내기에 대한 데이터의 조작을 거친 후에 워터마크 검출율을 계산하였으며, 각 공격에 대한 검출율은 다음과 같다(표 5, 6).

<표 6> 3가지 잘라내기에 대한 검출율

잘라내기	검출율(CR)	
1번(상단)	x	389/389=100.0
	y	389/389=100.0
2번(하단)	x	734/734=100.0
	y	734/734=100.0
3번(중앙)	x	321/321=100.0
	y	321/321=100.0

4.3 결과 분석

디지털 워터마킹에서는 워터마크를 삽입하지 않은 데이터에서 마치 워터마크를 삽입한 것처럼 워터마크가 검출이 될 수 있다. 이는 데이터 사용에 있어서 정당한 권리가 있는데도 불법적인 사용자로 인식을 하거나, 혹은 합법적인 방법으로 데이터를 구입하여도 그 데이터를 사용하지 못하게 될 수도 있다. 따라서 워터마킹 알고리즘은 이러한 긍정적 오류율에 관해서도 고려를 해야 한다. 이러한 긍정적 오류율을 알아보기 위하여, 인덱스 번호가 376082325인 다른 수치지도에 워터마크를 삽입하지 않고, 이 데이터를 가지고 워터마크가 추출이 되는지 여부를 확인해 보았다. 실험 결과 6227개의 좌표 중에서 x 좌표는 288개가 검출이 되어, 288/6227 = 4.62%가 나왔으며, y 좌표는 112개가 검출이 되어 112/6227 = 1.80%가 나왔다. 이 긍정적 오류율은 실질적인 워터마크 검출율에 비해 매우 작은 숫자로 나타나며 이는 곧 긍정적 오류율이 제안하는 방법에서는 큰 문제가 되지 않음을 나타낸다.

제안하는 방법은 워터마크 삽입시 일방함수와 함께 랜덤 테이블을 활용하여 워터마크를 삽입하도록 하였다. 워터마크 삽입은 전체 맵 데이터에서 이루어지도록 하였으며, 이전 방법과 마찬가지로 워터마크 삽입 이후 RMSE와 통계치를 계산하여 원본 데이터와의 차이를 계산하였다. 실험 결과 RMSE는 매우 작은 값이 나왔으며 이는 맵 데이터의 허용오차를 벗어나지 않는 값이었다. 그리고 워터마크 삽입 전·후를 살펴보아 위상관계 훼손 여부를 확인하였다. 제안한 방법으로는 워터마크를 삽입하여도 위상관계가 훼손되는 경우가 없었다.

제안한 방법에서 워터마크 검출은 워터마크 삽입과정과 유사하게 진행하며, 삽입한 워터마크가 공격당한 데이터에서 검출이 되는지를 살펴보았다. 직접 삽입한 워터마크를 추출하는 것과 동시에 CR 값을 통해 검출율을 살펴보았다. 검출 결과 CR 값은 매우 높은 수치로 나타났으며, 이러한 결과는 본 연구의 방법이 데이터 소유자의 소유권을 보호할 수 있음을 보여준다. 또한 CR 값뿐만 아니라 실질적으로 좌표에 삽입된 워터마크를 확인할 수 있어, 소유권 보호를 위한 충분한 근거를 제시할 수 있다.

제안한 방법은 워터마크 검출 과정에 있어서 원본 데이터를 사용하지 않고 워터마크 삽입시 사용했던 랜덤 테이블을 사용하기 때문에, 워터마크 검출시 좀 더 효율적이라고 할 수 있다. 특히, 랜덤 테이블을 이용하여 원래 좌표의 값을 새로운 값으로 변화시켜 사용하는 것은 제안한 방법의 보안성을 강화시켜 줄 수 있다. 또한 삽입한 워터마크의 비트수도 기본적으로 이전 방법보다 크기 때문에 좀 더 많은 양의 워터마크를 삽입할 수 있다. 이는 결국 긍정적 오류율을 줄여주는 효과로 나타났다.

<표 7> 긍정적 오류율

구분	폴리곤 데이터
x 좌표의 긍정적 오류율	288/6227 = 4.62%
y 좌표의 긍정적 오류율	112/6227 = 1.80%

5. 결 론

디지털 워터마킹은 기본적으로 워터마크를 삽입하는 것이기 때문에 원본 데이터에

손상이 갈 수 있다. 이것은 벡터 맵 데이터의 경우, 객체의 모양이 변한다거나 객체들간의 위상관계가 변하는 것으로 나타날 수 있다. 기존의 벡터 데이터에 관한 워터마킹을 살펴보면 소유권 보호를 위한 목적은 달성하고 있으나 이러한 점에 대해서는 간과하고 있는 경우가 많았다. 즉, 워터마크를 삽입한 후에는 원본 데이터와 달리 객체들간의 위상관계가 변하거나 모양이 변하는 경우가 나타났었다. 제안한 방법에서는 벡터 맵의 유효성을 유지하기 위하여 모양의 유지와 위상관계의 유지를 고려하여 워터마크를 삽입하였으며, 실험 결과 워터마크를 삽입한 후에도 벡터 맵 데이터의 변화는 거의 없었으며, 좌표의 변화 또한 수치지도의 위치 허용오차 이내에 있음을 알 수 있었다. 실험에서는 워터마크를 삽입한 후에 원점 이동과 회전, 잘라내기, 일반화, 객체 추가 공격을 하여 워터마크의 검출 능력을 살펴보았다. 검출 결과는 CR 값을 통해 삽입한 워터마크를 얼마나 검출할 수 있는지 수치로 알아볼 수 있었으며, 실제 삽입한 워터마크를 추출할 수 있도록 하여 보다 정확하게 데이터 소유권을 보호할 수 있음을 보여주었다.

하지만 본 연구의 방법에서 보완해야 할 점이 있다. 제안하는 방법은 벡터 맵 데이터의 정확성과 위상관계를 유지하기 위해 벡터 맵 데이터를 구성하고 있는 각 좌표에서 워터마크를 소수점 이하의 먼 자리에 삽입하기 때문에 경우에 따라서는 워터마크가 훼손될 가능성을 지니고 있다. 즉, 눈으로 보이는 좌표의 변화는 거의 없으나 아주 약한 공격을 할 경우에는 워터마크 검출이 매우 어려워질 수 있다.

이는 워터마크의 특징들인 충실도와 강인성이 서로 상반되는 성격을 지니고 있다는 것으로 설명을 할 수 있다. 원본 데이터와의 유사성을 유지시키기 위해 충실도를 중요시하면 강인성이 약해지고, 반대로 강인성을 중요시하면 충실도가 약해진다. 따라서 향후에는 벡터 맵 데이터의 활용에 있어서 크게 문제가 되지 않는 범위에서 충실도를 떨어뜨리지만, 여러 공격에도 워터마크가 살아남을 수 있는 강인성을 높일 수 있도록 하는 연구가 필요할 것이다.

또한 두 방법은 모두 전처리 과정이 필요한 방법으로서 모든 공격에 대해 필요한 과정은 아니나 geometrical 공격에 대해서는 전처리 과정을 요구하고 있다. 이는 벡터 맵 데이터의 장점을 활용하고자 하는 것이지만 워터마크 검출 과정에 있어서는 비효율적인 과정이라고 볼 수 있다. 따라서 이러한 전처리 과정이 필요 없이 워터마크 검출이 가능한 방법에 관한 연구가 필요하다.

제안하는 방법들이 지니고 있는 이러한 한계점들은 벡터 맵 데이터의 특성을 충분히 고려하기 위해 나타나는 문제점이라고 할 수 있다. 따라서 좀 더 강인한 워터마킹을 위해서는 벡터 맵 데이터의 장점들을 저하시킬 필요가 있으나 이는 향후에 벡터 맵 데이터의 활용 분야에 따라 분석이 필요한 부분이다. 결국 이러한 연구의 한계점을 극복해야 궁극적으로 벡터 맵 데이터의 특성을 완벽하게 반영하면서 보다 강인한 디지털 워터마킹을 구현할 수 있을 것이다.

감사의 글

본 연구는 국토해양부 첨단도시기술개발

사업 - 지능형국토정보기술혁신 사업과제의 연구비지원(과제번호 07국토정보 B01-01)에 의해 수행되었습니다.

참고문헌

김한규, 박동선, 이재광, 2004, 데이터 통신과 네트워킹, 교보문고.

김현승, 2005, 디지털 워터마킹, 도서출판 그린.

윤영주, 2004, “한글 문서 영상을 위한 텍스트 워터마킹”, 이화여자대학교 석사학위논문

Hwanil, Kang., Kabil, Kim., and Jonguk, Choi., 2001, A vector watermarking using the generalized square mask, Proceedings of International Conference on Information Technology: Coding and Computing, pp.234-236.

Itaru, Kitamura., Satoshi, Kanai., and Takeshi, Kishinam., 2001, Copyright Protection of Vector map using Digital Watermarking Method based on Discrete Fourier Transform, Proceedings of the IEEE 2001 International Symposium on Geoscience and Remote Sensing Symposium, Vol.3, pp.1191-1193.

Jungyeip, Kim., and Soohong, Park., 2007, Vector Map Data Watermarking Method using Binary Notation, The Journal of Geographic Information System Association of Korea, Vol.15, No.4, pp.385-395

López, Carlos., 2002, Wetermarking of digital geospatial datasets: a review of technical, legal and copyright issues, International Journal of Geographical Information Science, Vol. 16 No. 6, pp.589-607.

Michael, Voigt.·Christoph, Busch., 2003, Feature-based Watermarking of 2D Vector Data, Proceedings of the SPIE-Security and Watermarking of Multimedia Content, Vol. 5020, pp.359-366.

Michael, Voigt., Bian, Yang., and Christoph, Busch., 2004, Reversible Watermarking of 2D-Vector Data, Proceedings of the 2004 Multimedia and Security Workshop on Multimedia and security, pp.160-165.

Ryutarou, Ohbuchi., Hiroo, Ueda., and Shuh, Endoh., 2002, Robust Watermarking of Vector Digital Maps, Proceedings of IEEE International Conference on Multimedia and Expo, Vol. 1, pp.577-580.

Ryutarou, Ohbuchi., Hiroo, Ueda., and Shuh, Endoh., 2003, Watermarking 2D Vector Maps in the Mesh-Spectral Domain, Proceedings of International Conference on Shape Modeling and Applications, pp.216- 225.

Vassilios, Solachidis., Nikos, Nikolaidis., and Ioannis, Pitas., 2000, Fourier descriptors watermarking of vector graphics images, Proceedings of the International Conference on Image Processing, Vol. 3, pp.9-12.

Xiamu, Niu., 2006, A Survey of Digital Vector Map Watermarking, International Journal of Innovative Computing, Information and Control, Vol. 2, No. 6, pp.1301-1316.

Yuanyuan, Li., and Luping, Xu., 2003, A Blind Watermarking of Vector Graphics Images, Proceedings of the Fifth International Conference on Computational Intelligence and Multimedia Applications, pp.424-429.

접수일 (2009년 1월 22일)
최종수정일 (2009년 4월 16일)
게재확정일 (2009년 4월 21일)