고정 가중치 부호에 의한 복수 화상용 시각암호

김 미 라* 박 상 우** 박 지 환*

부경대학교 전자계산학과* 한국전자통신연구소**

Visual Cryptography for Multi Images Using Constant Weight Codes

Mi-Ra Kim* Sang-Woo Park** Ji-Hwan Park*

Dept. of Computer Science PuKyong National University*

Electronics and Telecommunications Research Institute**

요 약

M.Naor와 A.Shamir의 시각암호는 기존의 비밀 분산법에서 분산/복호시 연산량이 많은 것과는 달리 인간의 시각에 의해서 직접 복호될 수 있는 방식이다. 본 논문에서는 시각암호의 확장 방식인 복수 화상을 숨기는 방식에서 숨기려는 비밀화상의 수가 증가함에 따라 share 크기가 기하 급수적으로 커지는 문제점을 해결하기 위하여 고정 가중치 부호의 해밍 거리를 조정하여 share 크기를 줄이는 방법을 제안한다.

1. 서론

정보화 사회로 접어들면서 거의 모든 분야에서 컴퓨터를 이용한 정보 관리가 이루어진다. 관리되는 정보 중에는 여러 사람에게 공개되어도 무방한 정보가 있는가 하면 결코 공개되어서는 안될 중요한 정보도 있다. 만약, 개인이 중요한 정보를 관리하게 되면 정보의 누설 등의 문제가 발생하게 된다. 그래서 중요한 정보를 여러 개로 분산하여 임의의 개수 이상이 결합되면 비밀 정보의 접근이 허용되는 비밀 관리 체계인 (k, n)문턱치 방식이 A.Shamir에 의해 제안되었다[1]. 그러나, 이 방식은 비밀을 분산/복호하는 어떤 경우에도 연산량이 많기 때문에 고성능 컴퓨터를 사용해야 한다.

비밀 정보로서 화상이 이용되고 복잡한 암호 연산 없이도 복호될 수 있는 방식, 즉 비밀 분산법의 새로운 형태인 시각암호가 M.Naor와 A.Shamir에 의해 제안되었다[2]. 이 방식은 (k, n)비밀 분산법의 복호기 역할을 인간의 눈이 대신하는 것으로 분산된 비밀화상은 슬라이드와 같은 물리적으로 중첩 가능한 곳에 인쇄된다. 임의의 그룹 내 n명에게 배포된 슬라이드 중 어느 k명 이상의 슬라이드를 겹치는 경우에는 비밀화상이 복원 가능하지만, $k-1$명 이하의 슬라이드를 겹치는 경우에는 비밀화상이 복원 불가능한 방식이다.

시각암호의 확장 방식으로서 겹쳐진 슬라이드의 장수에 따라 다른 비밀화상이 복원되는 복수 화상을 숨기는 방식[3]과 겹쳐진 장수에 따라 비밀 정보가 서서히 복원되는 non-perfect 방식[4]이 T.Katoh와 H.Imai에 의해 제안되었다. 그러나, Katoh와 Imai의 복수 화상을 숨기는 방식에서는 숨기려는 비밀화상의 수가 증가할수록 share 크기가 기하 급수적으로 커지기 때문에 복원된 화상의 인식이 어려워지는 문제점이 있다.

본 논문에서는 고정 가중치 부호[5]를 이용하여 share 크기를 줄이는 방식을 제안한다. 2장에서는 M.Naor와 A.Shamir에 의해 제안된 방식을 소개하고, 3장에서는 T.Katoh와 H.Imai에 의해 제안된 복수 화상을 숨길 수 있는 방식과 non-perfect 방식의 구체적 구성법을 제시한다. 4장에서는 세 장의 슬라이드에 두 개의 비밀화상을 숨길 때 고정 가중치 부호의 해밍 거리를 조정하여 share 크기를 줄이는 방식을 제안하고, 그 성능을 컴퓨터 시뮬레이션을 통하여 평가한다. 마지막으로 5장에서는 결론 및 향후 연구과제를 도출한다.

2. Naor와 Shamir의 시각암호

2.1 기본 model

시각암호 또는 시각 비밀 분산 문제의 가장 간단한 형태는 비밀화상이 흑과 백의 화소 집합으로 구성되고, 각 화소들은 따로 따로 조작된다. 비밀화상의 각 화소는 n장의 슬라이드 상에 각각 m개의 부분 화소로 표현되며 이것을 share라 부르며, 화상의 크기는 m배 확대된다.

이 구조는 비밀화상의 각 화소가 $n \times m$ 부울 행렬 $S=[s_{ij}]$로 표현 가능하며, 이때 s_{ij}의 값은 i번째 share 중 j번째 부분화소가 흑인 경우에는 1을 백인 경우에는 0을 나타낸다. Share들을 정확하게 일치하도록 겹쳤을 때, 행렬 S의 행들의 부울 "or"로 표현되는 결합 share를 볼 수 있다. 결합 share의 grey 단계는 "or"연산을 한 m차 벡터 V의 해밍 가중치 $H(V)$와 비례한다. 이 grey 단계는 어떤 고정된 문턱치 $1 \le d \le m$와 상대적인 차 $\alpha > 0$에 대해서 $H(V) \ge d$이면 흑으로 $H(V) < d - \alpha m$이면 백으로 인식된다.

정의 1 (k, n)시각 비밀 분산법은 $n \times m$ 부울 행렬들의 두 집합 C_0, C_1으로 구성된다. 백화소를 분산하기 위해서 C_0의 행렬들 중 하나를 임의로 선택하고, 흑화소를 분산하기 위해서 C_1의 행렬들 중 하나를 임의로 선택한다. 선택된 행렬의 각 행은 한 개의 share에 대응하며 행의 각 요소가 1이면 흑을 0이면 백을 나타낸다. 다음 세 가지 조건을 만족하면 (k, n)시각 비밀 분산법의 해가 유효하게 된다.

1. C_0의 임의의 S에 대해서, n행들 중 임의의 k행의 "or" 연산을 한 m차 벡터 V의 해밍 가중치는 $H(V) < d - \alpha m$을 만족한다.
2. C_1의 임의의 S에 대해서, n행들 중 임의의 k행의 "or" 연산을 한 m차 벡터 V의 해밍 가중치는 $H(V) \ge d$를 만족한다.
3. $q < k$인 $\{1, 2, \cdots, n\}$의 임의의 부분 집합 $\{i_1, i_2, \cdots, i_q\}$에 대해서, $C_t (t \in \{0, 1\})$의 각 $n \times m$행렬을 행 $i_1, i_2, \cdots, i_q$로 제한함으로써 얻어진 $q \times m$행렬의 집합 $D_t (t \in \{0, 1\})$는 동일한 빈도를 갖는 동일한 행렬을 포함한다.

조건 1, 2는 share를 겹쳤을 때 복원되는 화상의 *contrast*를 나타내고, 조건 3은 k장 미만의 share를 겹쳤을 때 분산된 화소가 흑인지 백인지를 결정할 수 없는 *security*를 나타낸다.

시각 비밀 분산법에 사용되는 파라메타들은

- m : share를 구성하는 화소들의 수를 나타내며, 원화상과 분산된 화상과의 해상도 손실이 되므로 가능한 한 적어야 한다.
- α : 원화상의 백화소와 흑화소로부터 생성된 결합 share들간의 가중치의 상대적인 차로서 contrast의 손실을 나타내므로 가능한 한 커야 한다.
- r : 집합 C_0, C_1의 크기이며, $\log r$은 share들을 나타내기 위해 필요한 임의의 bit수로 화질에는 영향을 주지 않는다.

2.2 (k, k)시각 비밀 분산법

(k, k)시각 비밀 분산법을 구성하기 위해서 k개의 원소를 갖는 전체 집합 $W=\{e_1, e_2, \cdots, e_k\}$와 원소의 개수가 짝수인 부분집합 리스트 $\pi_1, \pi_2, \cdots, \pi_{2^{k-1}}$, 원소의 개수가 홀수인 부분집합 리스트 $\sigma_1, \sigma_2, \cdots, \sigma_{2^{k-1}}$을 고려하자.

$1 \le i \le k$와 $1 \le j \le 2^{k-1}$에 대해서 S_0와 S_1은 $e_i \in \pi_j$일 때 $S_0[i, j] = 1$, $e_i \in \sigma_j$일 때 $S_1[i, j] = 1$로 정의되는 $k \times 2^{k-1}$행렬이다. C_0와 C_1은 S_0와 S_1의 열들을 교환해서 만든 행렬들의 집합을 나타낸다.

$$C_0 = \{S_0\text{의 열들을 교환해서 만든 모든 행렬들}\}$$
$$C_1 = \{S_1\text{의 열들을 교환해서 만든 모든 행렬들}\}$$

그리고, (k, k)시각 비밀 분산법은 $m = 2^{k-1}$, $\alpha = \frac{1}{2^{k-1}}$, $r = 2^{k-1}!$을 갖는다.

3. Katoh와 Imai의 시각암호

3.1 (n, n)시각 비밀 분산법의 구성 방식

비밀화상의 각 화소를 n개의 share에 분산하는 경우를 생각한다. 행의 수가 n이고, i ($i = 0, 1, 2, \cdots, n$)개의 1을 갖는 모든 경우의 열 ($_nC_i$ 개)로 구성되는 행렬 $M_{n,i}$를 생각한다.

이때, 1의 수가 짝수인 열로 구성된 행렬을 선택해서 결합시킨 행렬 S_0가 (n, n)시각 비밀 분산법에서 백화소를 표현하는 share를 생성하고, 1의 수가 홀수인 열로 구성된 행렬을 선택해서 결합시킨 행렬 S_1이 (n, n)시각 비밀 분산법에서 흑화소를 표현하는 share를 생성한다.

예 : $n = 3$의 경우

$$M_{3,0} = \begin{pmatrix}0\\0\\0\end{pmatrix},\ M_{3,1} = \begin{pmatrix}100\\010\\001\end{pmatrix},\ M_{3,2} = \begin{pmatrix}011\\101\\110\end{pmatrix},\ M_{3,3} = \begin{pmatrix}1\\1\\1\end{pmatrix}$$

$$S_0 = \begin{pmatrix}0011\\0101\\0110\end{pmatrix},\ S_1 = \begin{pmatrix}1001\\0101\\0011\end{pmatrix}$$

이 행렬에 의해서 생성된 share를 한 개씩 겹쳤을 때 share에 포함된 흑부분 화소의 수는 표1과 같다. 실제로 행렬 S_0, S_1 각각의 열 교환으로 만들어진 모든 행렬 집합 C_0, C_1을 이용하여 (n, n)시각 비밀 분산법을 구성할 수 있다.

표1. 겹친 share의 개수와 흑부분 화소수

	1장	2장	3장		1장	2장	3장
$M_{3,0}$	0	0	0				
$M_{3,1}$	1	2	3	S_0(백)	2	3	3
$M_{3,2}$	2	3	3	S_1(흑)	2	3	4
$M_{3,3}$	1	1	1				

3.2 시각 비밀 분산법의 확장 방식의 일반화

$M_{n,i}$에서 생성된 각각의 share를 겹친 개수에 대응하는 흑부분 화소수의 천이를 열 벡터로 하는 행렬 T_n을 구한다. 단, 모든 요소가 0인 행렬($M_{n,0}$)은 흑부분 화소수의 천이에 영향을 미치지 않으므로 제거한다.

share 생성 행렬을 구성하기 위해서 필요한 $M_{n,i}$ ($i = 1, 2, \cdots, n$)의 개수 x_i를 요소로 하는 벡터를 $X_n = (x_1, x_2, \cdots, x_n)^T$, 구성된 행렬에 의해서 생성된 share를 j($j = 1, 2, \cdots, n$)개를 겹쳤을 때

share에 포함된 흑부분 화소수 y_i를 요소로 하는 벡터를 $Y_n=(y_1, y_2, \cdots, y_3)^T$라 한다. 이때, T_n, X_n, Y_n는 다음과 같은 관계식

$$Y_n = T_n X_n \quad (1)$$

$$X_n = T_n^{-1} Y_n \quad (2)$$

으로 된다.

흑 및 백화소로 이루어진 share를 생성하기 위해 필요한 흑부분 화소수의 천이의 상대적 오차, Y_n을 각각 $Y_{Wn}=(y_{w1}, y_{w2}, \cdots, y_{wn})^T$, $Y_{Bn}=(y_{b1}, y_{b2}, \cdots, y_{bn})^T$라 하면, 겹친 share의 개수에 따라 다음과 같은 관계가 성립한다.

	1장	2장	…	n장
Y_{Wn}	y_{w1}	y_{w2}	…	y_{wn}
Y_{Bn}	$y_{b1}=y_{w1}$	$y_{b2}=y_{w2}$	…	$y_{bn}=y_{wn}+1$

따라서, 흑 및 백화소의 share 생성 행렬을 구성하기 위해서 필요한 행렬 $M_{n,i}$의 개수를 각각 벡터 $X_{Wn}=(x_{w1}, x_{w2}, \cdots, x_{wn})^T$, $X_{Bn}=(x_{b1}, x_{b2}, \cdots, x_{bn})^T$라고 했을 때, 이것을 식(2)에 대입하여 share 크기 m을 최소로 만드는 X_{Bn}, X_{Wn}을 구한다.

벡터 X_{Bn}, X_{Wn}에 따라 share 생성 행렬을 구성했을 때 백화소의 share 생성 행렬의 열의 수가 흑화소의 share 생성 행렬의 열의 수보다 작은 경우가 발생한다. 이때는 행렬 $M_{n,0}$을 부족한 만큼 더해준다.

$M_{n,i}$에 의해서 생성된 share를 차례로 겹쳤을 때 share가 포함하는 흑부분 화소수의 천이를 일반적으로 나타내면, 행렬 $M_{n,i}$의 크기는 $n\times {}_nC_i$이고, 각 행은 $\frac{i}{n}\times {}_nC_i = {}_{n-1}C_{i-1}$개의 1과 ${}_nC_i - {}_{n-1}C_{i-1} = {}_{n-1}C_i$ 개의 0으로 구성된다. j 개의 share를 겹쳤을 때에도 그 부분화소가 백이 되도록 하기 위해서는 행렬 $M_{n,i}$의 j 개 행의 "or" 연산을 수행했을 때 그 요소가 0의 상태로 있어야 하며, 각 부분화소가 백일 확률 $P_W(n, i, j)$는

$$P_W(n, i, j) = \begin{cases} \frac{{}_{n-i}C_i}{{}_nC_j} & (n \geq i+j) \\ 0 & (n < i+j) \end{cases} \quad (3)$$

이고, $M_{n,i}$에 의해서 생성되는 share를 j 개 겹쳤을 때 흑부분 화소의 천이는 계수 $N_B(n, i, j)$에 의해서

$$N_B(n, i, j) = (1 - P_W(n, i, j)) \cdot {}_nC_i \quad (4)$$

로 표현된다. 따라서, 행렬 T_n은 일반적으로 아래와 같이 표현 가능하다. 단, $1 \leq i, j \leq n$이다.

$$T_n = [t_{ji}] \quad (5)$$

$$t_{ji} = (1 - P_W(n, i, j)) \cdot {}_nC_i \quad (6)$$

역행렬 T_n^{-1}은

$$T_n^{-1} = [x_{ij}] \quad (7)$$

$$x_{ij} = \begin{cases} (-1)^{(n-j)+(i-1)} {}_iC_{n-j}, & i \geq n-j \\ 0, & i < n-j \end{cases} \quad (8)$$

이다. 이때, 역행렬 T_n^{-1}의 각 요소의 크기인 열은 파스칼의 삼각형과 일치한다.

$M_{n,i}$에 대해서 행벡터의 해밍 가중치는 모두 동일하며 임의로 선택된 $j(=1,2,\cdots,n)$행의 "or" 연산으로 구해진 행벡터 또한 모두 동일한 해밍 가중치를 가지기 때문에 $M_{n,i}$에 의해서 구성된 1개의 share 생성 행렬에서 모든 행벡터의 해밍 가중치가 동일하므로 구별 불가능하다. 더욱이, 다른 share 생성 행렬 집합에서 선택된 행벡터도 각각 설정된 행수 미만의 행벡터의 "or"연산으로 구해진 행벡터의 해밍 가중치가 동일하기 때문에 어느 share 생성 행렬 집합에서 생성된 share인가를 구별 불가능하다. 따라서, 미리 설정된 개수 미만의 share로는 어떠한 정보도 얻을 수 없게 된다.

3.3 복수 화상을 숨기는 방식

지금까지의 방식은 한 개의 비밀화상을 n장의 슬라이드에 분산시켜, 정해진 장수 이상의 슬라이드를 겹치지 않는 한 비밀화상을 완전히 숨길 수 있는 방식이었고, Naor 등에 의해서 제안된 방식[1]에서도 이같은 방식의 구성법만 검토하고 있다.

예를 들면, $n=3$일 때 각각의 슬라이드는 아무 의미도 없지만, 이들 중 임의의 슬라이드 두 장을 겹치면 첫 번째 비밀화상이, 세 장의 슬라이드를 겹치면 두 번째 비밀화상이 나타나도록 구성하는 방법이다. 또한, 첫 번째 비밀화상에서는 두 번째 비밀화상의 추정이, 두 번째 비밀화상에서는 첫 번째 비밀화상의 추정이 모두 불가능하다.

가장 간단한 구성 예는 표2와 같은 흑부분 화소수의 천이를 만족하는 네 개의 share 생성 행렬을 구성할 수 있다. 단, 여기서는 벡터 $Y_{WW}=(y_1,y_2,y_3)$를 기준으로 표현한다. 3.2절과 같은 share 생성 행렬의 구성 과정을 적용하고 마지막으로 열의 수가 적은 경우에는 $M_{3,0}$를 더해준다.

표2. 겹친 share의 개수와 흑부분 화소수

	1장	2장	3장
Y_{WW}	y_{ww1}	y_{ww2}	y_{ww3}
Y_{BW}	y_{ww1}	$y_{ww2}+1$	y_{ww3}
Y_{WB}	y_{ww1}	y_{ww2}	$y_{ww3}+1$
Y_{BB}	y_{ww1}	$y_{ww2}+1$	$y_{ww3}+1$

$$\begin{cases} X_{WW}=(1,1,3) \\ X_{BW}=(0,3,0) \\ X_{WB}=(2,0,4) \\ X_{BB}=(1,2,1) \end{cases}$$

$$S_{00}=M_{3,0}M_{3,1}M_{3,2}M_{3,3}^3=\begin{pmatrix}0100011111\\0010101111\\0001110111\end{pmatrix}$$

$$S_{10}=M_{3,0}M_{3,2}^3=\begin{pmatrix}0110110110\\0101101101\\0011011011\end{pmatrix}$$

$$S_{01}=M_{3,1}^2M_{3,3}^4=\begin{pmatrix}1001001111\\0100101111\\0010011111\end{pmatrix}$$

$$S_{11}=M_{3,1}M_{3,2}^2M_{3,3}=\begin{pmatrix}1001101101\\0101011011\\0010110111\end{pmatrix}$$

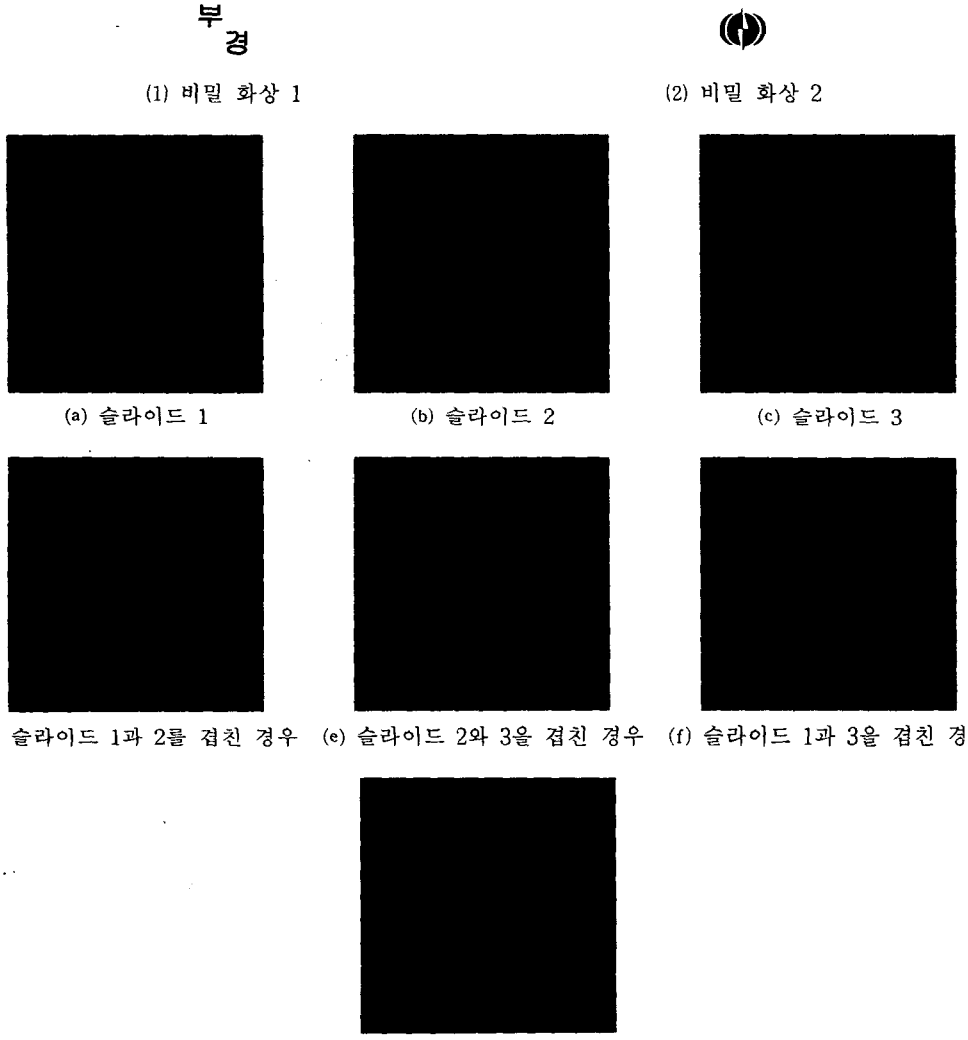
단, $M_{n,i}^p$는 행렬 $M_{n,i}$의 p개의 연결을 나타낸다.

벡터 Y_c $(c\in\{WW,BW,WB,BB\})$는 각각 식(2)에 대입해서 얻어지는 벡터 X_c를 기초로 구성되는 행렬 $S_{00},S_{10},S_{01},S_{11}$에 대응하는 벡터이다. 따라서, X_{WW}는 share를 두 개 또는 세 개를 겹쳤을 때 모두 백화소를 생성하는 행렬 S_{00}, X_{BW}는 share를 두 개 겹쳤을 때는 흑화소를, 세 개를 겹쳤을 때는

백화소를 생성하는 행렬 S_{10}, X_{WB}는 share를 두 개 겹쳤을 때는 백화소를 세 개 겹쳤을 때는 흑화소를 생성하는 행렬 S_{01}, X_{BB}는 share를 두 개 또는 세 개를 겹쳤을 때 모두 흑화소를 생성하는 행렬 S_{11}을 각각 구성하기 위해서 필요한 행렬 $M_{3,i}$ $(i=1,2,3)$의 개수를 요소로 갖는 벡터이다.

이 행렬에 의해서 생성된 share는 열 개의 부분화소로 구성되지만, 종횡비를 맞추기 위해서 각 share에 여섯 개의 부분화소를 군더더기로 추가해야 한다. 단, 추가하는 부분화소의 색은 모든 share에서 동일해야 하며 흑/백 제한은 없다.

그림1은 Katoh와 Imai에 의해 제안된 복수 화상용 시각암호의 시뮬레이션 결과이다. (1)과 (2)는 비밀화상1과 비밀화상2를 나타내며, (a), (b) 및 (c)는 각 슬라이드로서 아무런 정보도 얻을 수 없는 랜덤화상을 이룬다. 그러나, (d), (e)와 (f)는 각각 임의의 두 장을 겹쳤을 때 첫 번째 비밀화상이 되며, (g)는 세 개의 슬라이드를 모두 겹쳤을 때 두 번째 비밀화상이 나타남을 알 수 있다.



(1) 비밀 화상 1 (2) 비밀 화상 2

(a) 슬라이드 1 (b) 슬라이드 2 (c) 슬라이드 3

(d) 슬라이드 1과 2를 겹친 경우 (e) 슬라이드 2와 3을 겹친 경우 (f) 슬라이드 1과 3을 겹친 경우

(g) 슬라이드 1, 2, 3을 모두 겹친 경우

그림1. 3장의 슬라이드에 두 개의 비밀화상을 숨긴 경우(Katoh와 Imai 방식)

3.4 Non-perfect 방식

겹친 share의 개수의 증가에 따라 비밀화상이 서서히 복원되는 non-perfect 방식도 Katoh와 Imai에 의해 제안된 방식으로 실현 가능하다. 예를 들면, 네 장의 슬라이드에 "ABC" 세 문자를 숨기려고 할 때 한 장의 슬라이드로는 아무런 의미의 정보를 얻을 수 없고, 임의의 두 장을 겹치면 "A", 임의의 세 장을 겹치면 "AB", 네 장 모두 겹치면 "ABC"가 복원되는 방식이다.

행렬 $M_{4,i}$에서 생성된 각각의 share를 겹친 개수에 따른 흑부분 화소수의 천이를 열 벡터로 하는 행렬 T_4는

$$T_4 = \begin{pmatrix} 1 & 3 & 3 & 1 \\ 2 & 5 & 4 & 1 \\ 3 & 6 & 4 & 1 \\ 4 & 6 & 4 & 1 \end{pmatrix} \tag{9}$$

이고, T_4의 역행렬을 구해서 식(2)에 대입하면

$$X_4 = T_4^{-1} Y_4 = \begin{pmatrix} 0 & 0 & -1 & 1 \\ 0 & -1 & 2 & -1 \\ -1 & 3 & -3 & 1 \\ 4 & -6 & 4 & -1 \end{pmatrix} Y_4 \tag{10}$$

이 된다.

$Y_{14} = (a_1, a_2, a_3, a_4)$를 기준으로 한 겹친 share의 개수에 따른 흑부분 화소수의 천이는 표3과 같다.

표3. 겹친 share의 개수에 따른 흑부분 화소수의 천이

	1장	2장	3장	4장
Y_{14}	a1	a2	a3	a4
Y_{24}	a1	a2+1	a3+1	a4+1
Y_{34}	a1	a2	a3+1	a4+1
Y_{44}	a1	a22	a3	a4+1

share 생성 행렬을 구성하기 위해서 필요한 행렬 $M_{4,i}$의 개수를 각각 벡터 $X_{14} = (x_{14_1}, x_{14_2}, x_{14_3}, x_{14_4})^T$, $X_{24} = (x_{24_1}, x_{24_2}, x_{24_3}, x_{24_4})^T$, $X_{34} = (x_{34_1}, x_{34_2}, x_{34_3}, x_{34_4})^T$, $X_{44} = (x_{44_1}, x_{44_2}, x_{44_3}, x_{44_4})^T$라고 했을 때, 식(10)에 대입하면

$$\begin{cases} x_{14_1} = & & & (-1)\cdot a_3 + & 1\cdot a_4 \\ x_{14_2} = & & (-1)\cdot a_2 + & 2\cdot a_3 + & (-1)\cdot a_4 \\ x_{14_3} = & (-1)\cdot a_1 + & 3\cdot a_2 + & (-3)\cdot a_3 + & 1\cdot a_4 \\ x_{14_4} = & 4\cdot a_1 + & (-6)\cdot a_2 + & 4\cdot a_3 + & (-1)\cdot a_4 \end{cases}$$

$$\begin{cases} x_{24_1} = & & & (-1)\cdot a_3 + & 1\cdot a_4 \\ x_{24_2} = & & (-1)\cdot a_2 + & 2\cdot a_3 + & (-1)\cdot a_4 \\ x_{24_3} = & (-1)\cdot a_1 + & 3\cdot a_2 + & (-3)\cdot a_3 + & 1\cdot a_4 + 1 \\ x_{24_4} = & 4\cdot a_1 + & (-6)\cdot a_2 + & 4\cdot a_3 + & (-1)\cdot a_4 - 3 \end{cases}$$

$$\begin{cases} x_{34_1} = & & & (-1)\cdot a_3 + & 1\cdot a_4 \\ x_{34_2} = & & (-1)\cdot a_2 + & 2\cdot a_3 + & (-1)\cdot a_4 + 1 \\ x_{34_3} = & (-1)\cdot a_1 + & 3\cdot a_2 + & (-3)\cdot a_3 + & 1\cdot a_4 - 2 \\ x_{34_4} = & 4\cdot a_1 + & (-6)\cdot a_2 + & 4\cdot a_3 + & (-1)\cdot a_4 + 3 \end{cases}$$

$$\begin{cases} x_{44_1} = & & & (-1)\cdot a_3 + & 1\cdot a_4 + 1 \\ x_{44_2} = & & (-1)\cdot a_2 + & 2\cdot a_3 + & (-1)\cdot a_4 - 1 \\ x_{44_3} = & (-1)\cdot a_1 + & 3\cdot a_2 + & (-3)\cdot a_3 + & 1\cdot a_4 + 1 \\ x_{44_4} = & 4\cdot a_1 + & (-6)\cdot a_2 + & 4\cdot a_3 + & (-1)\cdot a_4 - 1 \end{cases}$$

되고, 다음과 같은 4×18 크기의 행렬이 구성된다.

$$\begin{cases} X_{14} = (0,1,2,3) \\ X_{24} = (0,1,3,0) \\ X_{34} = (0,2,0,6) \\ X_{44} = (1,0,3,2) \end{cases},$$

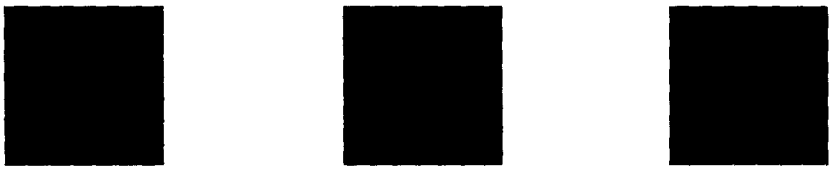
$$S_{14} = M_{4,2}M_{4,3}^2M_{4,4}^3 = \begin{Bmatrix} 111000011101111110 \\ 100011101110111110 \\ 010101110111011110 \\ 001110111011101110 \end{Bmatrix}$$

$$S_{24} = M_{4,2}M_{4,3}^3 = \begin{Bmatrix} 111000011101110111 \\ 100011101110111011 \\ 010101110111011101 \\ 001110111011101110 \end{Bmatrix}$$

$$S_{34} = M_{4,2}^2M_{4,4}^6 = \begin{Bmatrix} 111000111000111111 \\ 100011100011111111 \\ 010101010101111111 \\ 001110001110111111 \end{Bmatrix}$$

$$S_{44} = M_{4,1}M_{4,3}^3M_{4,4}^2 = \begin{Bmatrix} 100001110111011111 \\ 010010111011101111 \\ 001011011101110111 \\ 000111101110111011 \end{Bmatrix}$$

(a) 슬라이드 1 (b) 슬라이드 2 (c) 슬라이드 3 (d) 슬라이드 4



(e) 슬라이드 두 장을 겹친 경우 (f)슬라이드 3장을 겹친 경우 (g)슬라이드를 모두 겹친 경우

그림2. "ABC"를 non-perfect 방식으로 복원하는 과정

그림2에 non-perfect 방식으로 구현한 결과를 나타낸다. 즉, 한 장의 슬라이드만으로 나타내어지는 (a), (b), (c) 및 (d)는 랜덤한 화상이며, (e), (f) 및 (g)는 각각 두 장, 세 장 및 네 장을 겹쳐 감에 따라 "A", "AB" 및 "ABC"의 비밀화상이 차례로 복원됨을 알 수 있다.

4. 제안 방식

4.1 복수 화상을 숨기는 방식의 문제점

복수 화상을 숨기는 Katoh와 Imai방식에서 이론적으로는 많은 비밀화상을 숨길 수 있지만, 실제로 시각에 의해 인식 가능한 흑/백의 휘도비 α가 $\frac{1}{25}$ 에서 $\frac{1}{36}$ 정도로 제한되기 때문에 비밀화상의 수에

따른 share 크기의 변화(표4)에서 알 수 있듯이 기껏해야 두 개 또는 세 개만 숨길 수 있다. 따라서, share의 크기 m을 가능한 한 적게 해야 할 필요가 있다.

표4. 숨기려는 비밀 화상 수에 따른 share 크기의 변화

슬라이드 수	비밀화상 수	share 크기
3	2	10
4	3	36
5	4	116
6	5	358

4.2 고정 가중치 부호를 이용한 share 크기의 축소

3장의 슬라이드에 두 개의 비밀화상을 숨기는 방식에서 4.1절에서 제시된 문제점을 해결하기 위해서 고정 가중치 부호를 이용하여 share 크기를 줄이는 방법을 제안한다. 우선 구성 방식에 사용되는 파라메타들을

- m : 부호어의 길이 ($\sum_{i=0}^{2} {}_3C_i$)
- w : 한 부호어의 해밍 가중치

  $M_{3,0}, M_{3,1}, M_{3,2}, M_{3,3}$를 연결했을 때 생성되는 행렬 M_3의 한 행의 해밍 가중치

$$M_3 = M_{3,0}M_{3,1}M_{3,2}M_{3,3} = \begin{pmatrix} 01001101 \\ 00101011 \\ 00010111 \end{pmatrix}, \quad w = 4$$

- d : 두 부호어 간의 해밍 거리

로 정의한다.

두 비밀화상의 화소값 조합 WW, WB, BW, BB에 대응하는 share 생성 행렬을 구성하기 위해서 겹친 share의 개수에 따른 흑부분 화소수의 천이를 표5에 나타내었다. 표5에서 기준이 되는 화소값 조합 WW의 흑부분 화소수는 행렬 M_3의 "or" 연산을 수행한 두 행의 해밍 가중치와 세 행 모두 "or" 연산했을 때 해밍 가중치에 각각 대응한다. 나머지 화소값 조합에 대한 흑부분 화소수는 겹친 share의 개수에 따라 백화소(W)일 때 화소값 조합 WW의 흑부분 화소수와 같고, 흑화소(B)일 때 화소값 조합 WW의 흑부분 화소수에 1을 더한 값이다.

표5. 겹친 share에 따른 흑부분 화소수의 천이

두 비밀 화상의 화소값 조합	두 개의 share를 겹쳤을 때 흑부분 화소수	3개의 share를 겹쳤을 때 흑부분 화소수
WW	6	7
WB	6	8
BW	7(7, 7, 5)	7
BB	7(7, 7, 5)	8

$m = 8, w = 4$일 때 두 행에 "or" 연산을 한 해밍 가중치가 6이 되기 위해서 $d = 4$이어야 한다. $d = 4$인 부호어를 뽑아내면 두 행에 대한 해밍 가중치는 모두 6이 되고, 3행에 대한 해밍 가중치는 5, 7 또는 8이 된다. 따라서, 표5의 화소값 조합 WW, WB의 해밍 가중치를 만족시키는 행렬을 쉽게 구할 수 있다.

화소값 조합 BW, BB를 위한 행렬은 두 행에 대한 해밍 가중치가 7이 되어야 한다. 그러나, d가 4인 경우 두 행에 대한 해밍 가중치 7이 생성되지 않으며, 6인 경우 두 행을 뽑는 모든 경우에 대한

해밍 가중치가 7로 생성되지 않으며, 세 행에 대한 해밍 가중치, 7 또는 8도 생성되지 않으므로 $d=2$ 인 경우도 함께 생각한다. 화소값 조합 BW, BB에 대해서 세 행의 해밍 가중치가 각각 7과 8이면서 임의의 두 행의 해밍 거리가 2이고 나머지 두 경우의 해밍 거리가 6인 부호어로 행렬을 구성하면 다음과 같다.

$$S_{WW} = \begin{pmatrix}11110000\\11001100\\10101010\end{pmatrix}, \quad S_{WB} = \begin{pmatrix}11110000\\10101100\\10100011\end{pmatrix}$$

$$S_{BW} = \begin{pmatrix}11110000\\10001110\\01111000\end{pmatrix}, \quad S_{BB} = \begin{pmatrix}11110000\\10001110\\11100001\end{pmatrix}$$

행렬 S_{BW}, S_{BB}에서 세 행에 "or" 연산을 한 해밍 가중치는 7과 8로 표5와 동일하지만, 두 행에 대한 "or" 연산의 해밍 가중치는 괄호 속에 있는 값으로 나타난다.

두 행에 대한 해밍 가중치가 모두 7로 표5의 해밍 가중치를 만족시키지 않지만, 제안된 구성 방식으로 슬라이드를 구성하면 두 개의 슬라이드를 겹치는 세 가지 경우 모두에 대해서 첫 번째 비밀화상이 복원된다. 왜냐하면, 두 행에 대한 해밍 가중치가 5인 부분은 주위의 해밍 가중치 6보다 적기 때문에 반전된 형태로 복원된다. 또한, 모든 share 생성 행렬의 각 행이 동일한 해밍 가중치를 가지므로 안전성은 보장된다.

4.3 시뮬레이션 결과

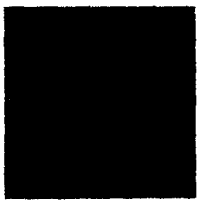
종횡비를 왜곡시키지 않기 위해 4.2절에서 제안된 share 생성 행렬에 1만 갖는 행렬 $M_{3,3}$을 추가해서 시뮬레이션 하였다. 시뮬레이션을 위해서 사용된 원화상은 그림1의 (1)과 (2)이다. (1)은 첫 번째 비밀화상이고, (2)는 두 번째 비밀화상으로 크기는 각각 120×120화소를 이룬다.

그림3에서 (a), (b), (c)는 각각 제안 방식으로 구성된 네 개의 행렬을 사용하여 원화상이 3×3배 확대된 그림이다. 첫 번째 비밀화상인 "부경"을 나타내고 있는 (d), (e), (f)는 각각 (a)와 (b), (b)와 (c), (a)와 (c)를 각각 겹쳤을 때의 결과이며, (g)는 (a), (b), (c) 모두 겹친 결과로 두 번째 비밀화상을 나타내는 그림이다.

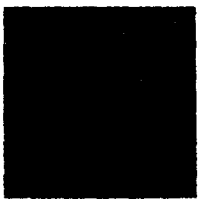
Katoh와 Imai 방식의 결과인 그림1과 제안 방식의 그림3에서 복원된 첫 번째 화상과 두 번째 화상을 시각적으로 비교해 보면, 그림1에 비해 그림3이 훨씬 더 선명하게 복원됨을 알 수 있다. 이것은 그림3에서 사용된 share 생성 행렬의 크기가 그림1에서 사용했던 share 생성 행렬의 크기보다 더 적어지기 때문이다. 즉, 복원된 화상에서 선명도를 나타내는 흑/백화소 사이의 휘도비 α가 $\frac{1}{9}$과 $\frac{1}{16}$로 그림3에 비해 그림1이 더 크기 때문이다.

마지막으로 그림3에서 (a), (b), (c)는 모두 독립적으로는 아무런 정보도 포함하지 않는 것처럼 보이며, 실제로 이들만으로는 두 비밀화상에 대한 어떤 것도 추정할 수 없다.

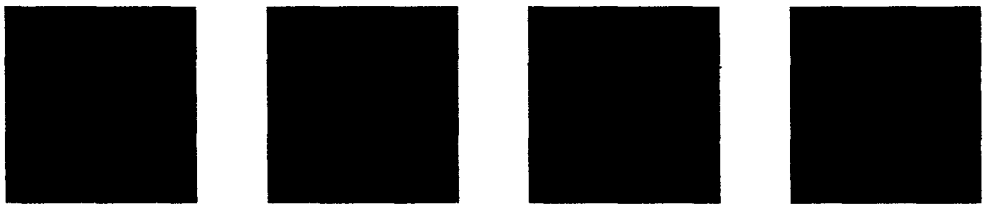
(a) 슬라이드 1



(b) 슬라이드 2



(c) 슬라이드 3



(d) 슬라이드 1과 2를 겹친 경우 (e) 슬라이드 2와 3을 겹친 경우 (f) 슬라이드 1과 3을 겹친 경우 (g) 슬라이드 1, 2, 3을 모두 겹친 경우

그림3. 제안 방식으로 구성된 슬라이드

5. 결론

시각암호에 대한 고찰과 non-perfect 방식의 구체적인 구성법을 제시했고, Katoh와 Imai 방식에서는 비밀화상의 수에 따라 share 크기가 커지기 때문에 본 논문에서는 세 장의 슬라이드에 두 개의 비밀화상을 숨길 때 고정 가중치 부호의 해밍 거리를 조정하여 share의 크기를 줄이는 방법을 제안하였다.

제안 방식에 의해 복원되는 첫 번째 화상이 세 장의 슬라이드 중 어떤 두 장을 뽑느냐에 따라 화상이 반전되기는 하지만, 안전성이 보장되고 Katoh와 Imai가 제안한 방식 보다 share 크기를 줄이면서 두 개의 비밀화상을 완전히 복원할 수 있었다. 향후, 제안 방식의 일반화에 대한 연구가 필요하며, 제안 방식보다 share 크기를 더 줄일 수 있는 방식에 대한 연구가 계속되어야 할 것이다.

참 고 문 헌

[1] A.Shamir, "How to share secret," Commun. of the ACM, 22, pp.612-613, 1979.

[2] M.Naor and A.Shamir, "Visual Cryptography," Advances in Cryptology-EUROCRYPT'94, Perugia, Italy, pp.1-12, May 1994.

[3] T.Katoh and H.Imai, "On Extended and Applications of Visual Secret Sharing," ISEC95-19, pp.41-47, September 1995.(in Japanese)

[4] T.Katoh and H.Imai, "An Extended Construction Method of Visual Secret Sharing Scheme," IEICE Trans., vol.J79-A no.8, pp.1344-1351, August 1996.

[5] R.E.Kibler, "Some new constant weight codes," IEEE Trans. Inform. Theory, vol.IT-26, pp.364-365, May 1980.