

# 특이치 분해와 Fuzzy C-Mean(FCM) 군집화를 이용한 벡터양자화에 기반한 워터마킹 방법

## An Watermarking Method based on Singular Vector Decomposition and Vector Quantization using Fuzzy C-Mean Clustering

이병희<sup>1</sup>, 장우석<sup>2</sup>, 강환일<sup>3</sup>

<sup>1</sup> 경기도 용인시 명지대학교 정보공학과  
E-mail: lbh0329@naver.com

<sup>2</sup> 경기도 용인시 명지대학교 정보공학과  
E-mail: hideto7@mju.ac.kr

<sup>3</sup> 경기도 용인시 명지대학교 정보공학과  
E-mail: hwan@mju.ac.kr

### 요 약

본 논문은 원본이미지와 은닉이미지의 좋은 압축률과 만족할만한 이미지의 질, 그리고 외부 공격에 강인한 이미지은닉의 한 방법으로 특이치 분해와 퍼지 군집화를 이용한 벡터양자화를 이용한 워터마킹 방법을 소개하였다. 실험에서는 은닉된 이미지의 비가시성과 외부공격에 대한 강인성을 증명하였다.

**Key Words** : Singular value decomposition, Vector quantization, Fuzzy C-Mean clustering

### 1. 서 론

인터넷의 보급, 통신 기술의 발달과 다양한 매체의 개발로 인해 우리는 어디서나 쉽게 디지털화된 미디어를 접할 수 있다. 이러한 미디어 상에서는 누구나 방대한 디지털화된 데이터를 쉽게 복제하고 조작할 수 있게 됨에 따라 미디어의 지적 소유권 문제가 이슈화 되고 있다. 따라서 다양한 저작권 보호에 대한 방법이 연구되고 있는데 그 중 하나가 워터마킹 기술이다. 하지만 디지털화된 미디어의 데이터는 방대한 반면 대부분의 워터마킹 방법은 삽입되는 워터마크의 데이터 크기가 제한적이고 또한 삽입되는 데이터의 압축 방법에만 초점을 두고 있다.

본 논문에서는 원본이미지와 삽입된 이미지의 데이터 압축율과 만족할 만한 질의 워터마크 된 이미지를 얻기 위한 방법 중 특이치 분해와 FCM을 이용한 벡터 양자화를 적용한 이미지 은닉 방법을 소개하고 외부 공격에 대한 강인성을 실험을 통해 검증하였다.

### 2. 특이치 분해와 퍼지 클러스터링을 이용한 벡터 양자화

#### 2.1 특이치 분해

특이치 분해는 규모가 크고 다루기 힘든 행렬의 특이행렬을 얻어내는 방법 중의 하나로 유용하게 사용되는 행렬식으로 얻어진 특이행렬은 다양한 공격으로 인하여 데이터의 변형이 일어나더라도 특이 행렬의 특성은 크게 변하지 않는다는 특징이 있다. 특이치 분해에서 만일  $n \times n$  크기 행렬의 계수가  $r$ 인 이미지 행렬  $A$ 가 있다고 한다면, 행렬  $A$ 는 식(1)과 같이 나타낼 수 있다.[1] 여기서 행렬  $U$ 와  $V$ 는 열벡터  $u_i$  와  $v_i$ 를 갖는  $n \times n$  크기의 직교행렬이고  $D$ 는 식 (2)를 만족하는  $\sigma_i$ 을 대각 성분으로 갖는  $n \times n$  크기의 대각 행렬이다.

$$A = UDV^T$$

$$= [u_1, u_2, \dots, u_N] \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_N \end{bmatrix} \times [v_1, v_2, \dots, v_N]^T$$

$$= \sum_{i=1}^r \sigma_i u_i v_i^T, \quad (1)$$

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > \sigma_{r+1} = \dots = \sigma_N = 0 \quad (2)$$

식(2)에서 행렬  $A$ 의 계수  $r$ 는 0이 아닌 특

이치의 수를 나타내며 0이 아닌 특이치의 값은 행렬  $A$ 보다 낮은 계수를 갖는 행렬과의 근접도를 나타낸다. 또한 식(1)로부터 특이치 분해 요소들은  $Av_i = \sigma_i u_i$  와  $u_i^T A = \sigma_i v_i^T$  관계를 만족한다. 일반적으로 행렬  $A$ 는 여러 작은 값의 특이치를 가지므로 보다 낮은 계수의 행렬과 근접할 수 있다. 만일  $k \leq r$ 을 만족하는  $k$ 가 있다면 행렬  $A$ 에 근접한 행렬  $\hat{A} = \sum_{i=1}^k \sigma_i u_i v_i^T$  을 나타낼 수 있으며 에러 행렬은  $E = A - \hat{A}$ 로 나타낼 수 있다.

### 2.2 퍼지 C-Mean 클러스터링

FCM(Fuzzy C-Mean) 클러스터링 방법은 하나의 클러스터에 속해져 있는 각각의 데이터들을 클러스터에 대한 데이터의 소속정도에 따라 분류하는 데이터 분류 알고리즘으로 Bezdek에 의해 1973년도에 제안되었다.[2] FCM 클러스터링은  $n$ 개의 벡터의 집합을  $c$ 개의 퍼지그룹으로 분할하고 소속정도 측정 비용함수가 최소가 되는 것과 같은 각각의 그룹 안에서 클러스터의 중심을 찾는다.

FCM 클러스터링 방법은 0과 1사이의 소속정도에 의해서 나타난 소속감 정도를 가지고 주어진 데이터 점이 몇 개의 그룹에 속할 수 있다는 퍼지분할을 사용한다. 즉, 퍼지 분할을 적용하기 위해서 소속 함수  $U$ 는 0과 1사이 값을 가지는 요소들을 가지며 데이터 집합에 대한 소속감 정도의 합은 식(3)과 같이 항상 1이다.

$$\sum_{i=1}^c u_{ik} = 1, \quad \forall k = 1, \dots, n$$

$$0 < \sum_{k=1}^n u_{ik} < n$$
(3)

FCM 클러스터링에 대한 목적함수는 식(4)과 같은 형태를 가지고 있다.

$$J(u_{ik}, v_i) = \sum_{i=1}^c \sum_{k=1}^n u_{ik}^m (d_{ik})^2$$

$$d_{ik} = d(x_k - v_i) = \left[ \sum_{j=1}^L (x_{kj} - v_{ij})^2 \right]^{1/2}$$

$$v_i = \{v_{i1}, v_{i2}, \dots, v_{ij}, \dots, v_{iL}\}$$
(4)

여기서,  $u_{ik}$  는 0과 1사이의 수적인 값으로  $i$  번째 클러스터에 속해져 있는  $x_k$ 의  $k$ 번째 데이터의 소속정도를 나타낸다.  $v_i$ 는  $i$ 번째 클러스터의 중심 벡터이다.  $j(j = 1, 2, \dots, L)$ 는 특성 공간상의 변수이다.  $m$ 은 소속 함수의 퍼지성의 정도에 대한 영향을 나타내는 지수의 가중치이다. 이 값은  $m \in [1, \infty)$  와 같은 범위를 가지

고 있으며, 분류 공정에서 퍼지성의 양을 제어하는 파라미터이다. 보통  $m$ 을 2로 설정한다.

FCM 클러스터링 알고리즘은 다음과 같다.  
과정 1. (초기화 단계) 클러스터 개수  $c(2 < c < n)$ 을 정하고 지수의 가중치  $m(1 < m < \infty)$ 을 선택한다. 초기 소속 함수  $U^{(0)}$ 를 초기화한다. 알고리즘 반복횟수를  $r(r = 0, 1, 2, \dots)$ 로 표시한다.  
과정 2. 다음 식(5)을 이용해 퍼지 클러스터의 중심  $\{v_i | i = 1, 2, \dots, c\}$ 을 구한다.

$$v_{ij} = \frac{\sum_{k=1}^n (u_{ik})^m x_{kj}}{\sum_{k=1}^n (u_{ik})^m}$$
(5)

과정 3. 다음과 같이 새로운 소속 함수  $U^{(r+1)}$ 을 계산한다.

$$u_{ik}^{(r+1)} = \frac{1}{\sum_{j=1}^c \left( \frac{d_{ik}^r}{d_{jk}^r} \right)^{2/m-1}} \quad \text{for } I_k = 0$$

또는

$$u_{ik}^{(r+1)} = 0 \quad \text{for all classes } i,$$

여기서,  $i \in \tilde{I}_k$ ,  $I_k = \{i | 2 \leq c < n; d_{ik}^{(r)} = 0\}$ 이고  $\tilde{I}_k = 1, 2, \dots, c$ 이다. 그리고  $\sum_{i \in I_k} u_{ik}^{(r+1)} = 1$  이다.

과정 4. 식(6)을 계산해서 만일  $\Delta > \epsilon$ 이면  $r = r + 1$ 로 정하고 단계2로 가서 다시 알고리즘 반복 수행, 그렇지 않으면 종료한다. ( $\epsilon$  : 임계값)

$$\Delta = \|U^{(r+1)} - U^{(r)}\| = \max_{i,k} |u_{ik}^{(r+1)} - u_{ik}^{(r)}|$$
(6)

### 2.3 벡터 양자화

벡터 양자화는 낮은 비트율의 이미지 압축을 위한 중요한 기술이고  $l$ -차원 유클리디안 공간  $R^l$ 에서  $R^l$ 의 한정된 부분공간인  $C$ 로 사상 함수  $Q$ 로 정의될 수 있다.[3]

$$Q: R^l \rightarrow C,$$

여기서  $C = \{c_i | i = 1, 2, \dots, N\}$ 는 크기가  $N$ 인 코드북(codebook)이고  $c_i = (c_{i1}, c_{i2}, \dots, c_{il})$ 는  $l$ -차원의  $C$ 공간에서 각 가능한 사상된 코드워드(codeword)들이다.

벡터 양자화 과정은 다음의 세 과정으로 나뉜다: 코드북 생성, 부호화, 복호화.

본 논문에서는 FCM 클러스터링 방법을 사용하여 각 클러스터의 중심 값으로 코드북을 생성하였다. 부호화 과정에서 부호기는 먼저 이미지를 여러 블록으로 나누고 영상의 뒤틀림을 최소화하기 위해 주어진 벡터  $x = (x_1,$

$x_2, \dots, x_l$ )와 식(7)의 유클리디안 자승거리가 최소가 되는 코드워드의 인덱스를 취한다.

$$d^2(x, c_i) = \min_j \sum_{t=1}^l (x_t - c_{jt})^2 \quad (7)$$

인덱스를 나타내기 위해 사용되는 비트들은 항상 벡터  $x$ 를 나타내는데 필요한 것보다 적기 때문에 부호화된 이미지는 압축되어진다. 복호화 과정에서 복호기는 부호기와 같은 코드북을 사용해 단지 복호화 된 코드워드  $c_i$ 을 찾아 입력벡터  $x$ 에 근접된 값으로 재구성한다. 이와 같은 낮은 복잡성 때문에 벡터 양자화는 다른 압축 기술보다 좋은 계산적인 이점을 갖고 있다.

### 3. 특이치 분해와 벡터 양자화에 의한 이미지 압축

비록 특이치 분해에 의해 모든 특이치와 벡터들을 구할 수 있다 하더라도 어느 정도의 계산적인 복잡성을 갖는다. 부분적인 특이치 분해를 위해 더 적은 계산적 복잡성을 갖는 반복적 추산과정이 Yang 와 Lu에 의해 제안(1995)되었다.[4]

$$\begin{aligned} R_1 &= A \\ \hat{\sigma}_i &= u_i^T R_i v_i \quad i=1, 2, \dots, k, \\ R_{i+1} &= R_i - \hat{\sigma}_i u_i v_i \quad i=1, 2, \dots, k-1. \end{aligned} \quad (8)$$

식(8)에서  $R_i$ 는 처음  $(i-1)$  특이치 분해 요소들 ( $\hat{\sigma}_j, u_j$  그리고  $v_j$ )을 원래 행렬에서 빼준 나머지 행렬이라 부른다.  $v_i, u_i$ , 그리고  $\hat{\sigma}_i$ 의 계산 방법은 과정 3-5에서 소개한다.

모든 특이치 분해 요소들의 압축률을 높이기 위해 Yang 과 Lu(1995)는 특이치 분해를 이용한 압축에 벡터 양자화를 적용하는 알고리즘을 제안하였고, 그 과정은 다음과 같다.

과정 1 (코드북 생성) 주어진  $2^{n_1} \times 2^{n_2}$  크기의 각각의 화 습 이미지를 먼저  $2^{n_1-m} \times 2^{n_2-m}$  개의 블록(크기  $2^m \times 2^m$ )으로 나눈 후 각 블록을 특이치 분해 알고리즘을 이용해 특이벡터  $u_i, v_i$ 를 얻는다. 위의 과정으로 얻어진 특이벡터들을 코드북 디자인 방법을 이용해 두 개의 코드북  $C_U(), C_V()$ 를 얻는다.

과정 2 (입력이미지 분할) 주어진  $2^{n_1} \times 2^{n_2}$  크기의 입력이미지를 먼저  $2^{n_1-m} \times 2^{n_2-m}$ 개의 블록(크기  $2^m \times 2^m$ )으로 나누고 다음의 과정 3-5를  $k$ 번 수행한다.  $j=1$ 로 정한다.

과정 3 ( $v_j$  부호화) 행렬의 노름 계산방법을 이용하여 코드북  $C_V$ 의 모든 코드워드를 이용하여 식(9)를 만족하는 인덱스  $p$ 를 구한다.

$$\max_{\{1 \leq p \leq 2^j\}} \|R_j C_V(p)\|_2 \leq \|R_j\|_2 \|C_V(p)\|_2 = \|A\|_2 = \sigma_j \quad (9)$$

식(9)에서  $R_j$ 는 블록  $B_i$ 에서 초기  $(j-1)$  특이치 분해 요소들을 뺀 나머지 행렬이다. 얻어진 인덱스  $p$ 는  $v_j$ 에 가장 가까운 특이벡터를 가리키며  $p^j$ 로 표시한다.

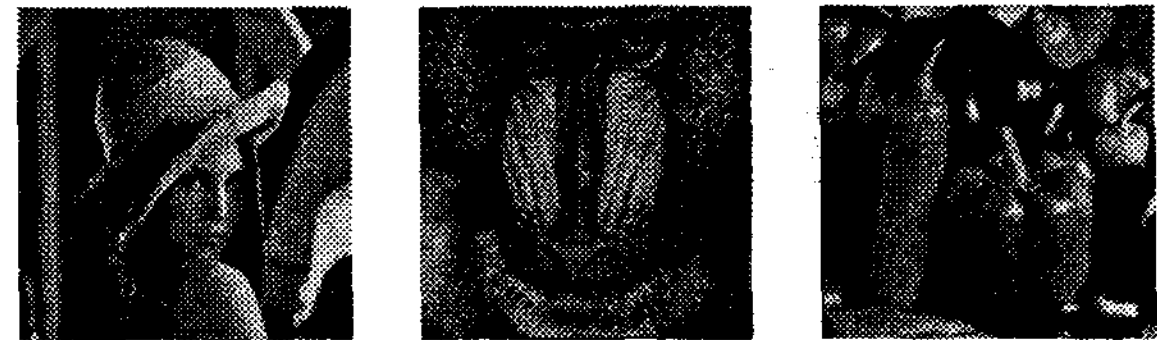
과정 4 ( $u_j$  부호화) 과정 3에서 얻어진  $v_j$ 에 가장 가까운 특이벡터  $C_V(p^j)$ 로부터 코드북  $C_U$ 에서의 코드워드  $C_U(q)$ 는 식(10)을 이용하여 계산되어진다.

$$\begin{aligned} \max_{\{1 \leq q \leq 2^j\}} C_U^T(q) R_j C_V(p^j) &\leq \|C_U^T(q)\|_2 \|R_j C_V(p^j)\|_2 \\ &= \|R_j C_V(p^j)\|_2 \leq \sigma_j, \end{aligned} \quad (10)$$

과정 3에서와 같이 얻어진 인덱스  $q$ 는  $u_j$ 에 가장 가까운 특이벡터를 가리키며  $q^j$ 로 표시한다.

과정 5 (가장 근접한  $\sigma_j$  계산) 식(8)에 의해 부호화된  $v_j$ 와  $u_j$ 를 이용하여 특이치  $\sigma_j (=u_j^T R_j v_j)$ 는 계산된다. 만일  $j < k$  라면  $j=j+1$  연산 후 과정 3부터 다시 반복한다.

각 블록은 과정 3-5를  $k$  기본 특이치 분해 요소를 계산하기 위해  $k$ 번 반복하게 되는데 이것을 부분 특이치 분해 과정이라 한다.



(a) Lena (b) Baboon (c) Pepper  
그림 1. 실험에 사용될 이미지 (512×512, Gray)

### 4. 워터마크 삽입 방법

만일  $2^{n_1} \times 2^{n_2}$  크기의 원본 이미지를  $CI$ 라 하고 은닉될 이미지를  $EI$ 라 한다면 각각의 이미지를  $2^{n_1-m} \times 2^{n_2-m}$ 개의 블록으로 나누면 다음과 같이 표현된다.

$$\begin{aligned} CI &= \{CI_i, 1 \leq i \leq 2^{n_1-m} \times 2^{n_2-m}\} \\ EI &= \{EI_i, 1 \leq i \leq 2^{n_1-m} \times 2^{n_2-m}\} \end{aligned}$$

소개되는 이미지 은닉 과정은 다음과 같다:

과정 1 (부분 특이치 분해 과정)  $CI$ 와  $EI$ 의 각 블록을 특이치 분해 방법을 이용하여 처음  $k$ 기저 특이치 분해 요소  $u_j, v_j, \hat{\sigma}_j$ 와 인덱스  $p^j, q^j$ 를 구한다.

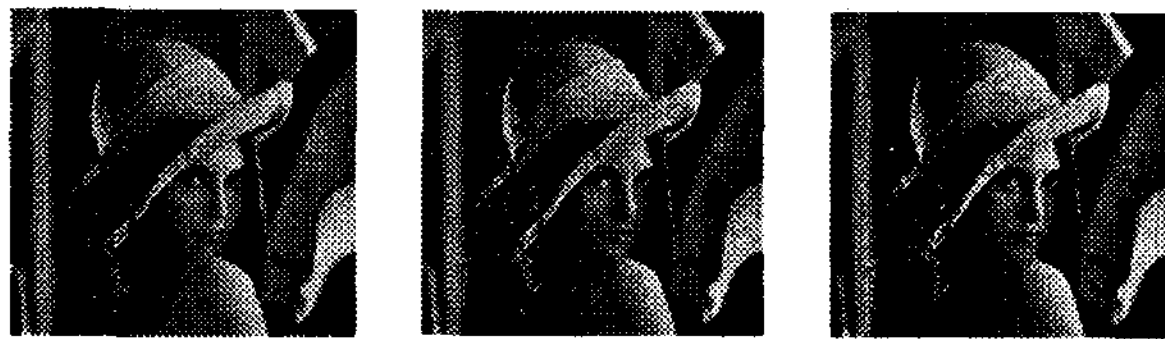
과정 2 (데이터 삽입) 열 우선 방법에 의해 원본이미지  $CI$ 의 블록  $CI_j$ 에서  $k+1 \leq j \leq 2k$

위치의  $k$  특이치 분해 요소를 은닉 이미지  $CI$ 의 블록  $EI_k$ 의 처음  $k$  특이치 분해 요소로 대체하다. 그리고 나머지  $2k+1 \leq j \leq 2^m$  위치의  $u_j, v_j, \hat{\sigma}_j$ 을 0으로 놓는다.

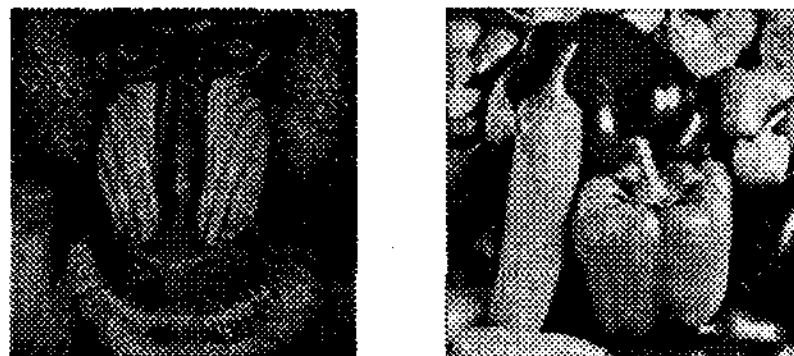
과정 3 (양자화) 은닉 데이터가 삽입된 블록  $CI_k$ 의 새로 생성된 특이치의 대각 순서는 내림차순이 아닐 수 있기 때문에  $k$ 개의 삽입된 특이치는 양자화 과정에 의해 양자화 된다. 양자화 된 특이치는 주 대각 성분이 내림차순으로 위치할 것이다.

과정 4 (특이치의 부호화)  $CI_k$ 의 초기  $k$  특이치 분해 구성 요소 들은 원래 값보다 크기 때문에 각각 가장 가까운 정수로 부호화 된다. 하지만 다음  $k$ 개의 양자화 된 특이치는 원래 값보다 작기 때문에 유지한다. 부호화된  $CI_k$ 를 워터마크 된 이미지라 부른다.

위에 소개된 이미지 은닉 과정은 쉽게 이미지를 은닉할 수 있고 또한 은닉된 이미지를 역 과정을 통해 쉽게 구해낼 수 있다.



(a) 원본이미지 (b) Baboon삽입 (c) Pepper삽입  
그림 2. 원본이미지와 워터마크 된 이미지



(a) (b)  
그림 3 추출된 이미지: (a) Baboon, (b) Pepper

표 1. 은닉이미지 삽입 전, 삽입 후 PSNR

삽입 이미지	코드북 크기	
	64	128
Lena	32.162	33.34
Baboon	30.1645	31.1645
Pepper	29.874	30.148

## 5. 실험 및 결과

실험에서는 먼저 소개한 이미지 은닉 방법을 시험해 보고 외부 자극에 관한 실험을 하였다. 먼저 이미지 은닉 실험에서는 원본이미지로 512×512 크기의 Lena 이미지 (그림 1(a))를 사용하였고 학습이미지로 512×512크기의 Lena,

Baboon (그림 1(b)), Peper (그림1(c))를 사용하였다. 각 이미지의 블록은 8×8 크기로 나누어 실험하였고 k=3개의 특이치 분해 요소를 계산하였다. 코드북은 64, 128 크기로 실험하였으며 코드북 디자인은 FCM 클러스터링을 사용하여 Lena, Baboon, Pepper 등의 학습이미지를 기반으로 디자인 하였다. 소개된 방법의 효과를 측정하기 위해 최대신호 대 잡음비 (PSNR: Peak Signal to Noise Ratio)을 사용하였다.

표 1과 그림 2는 원본 이미지만 양자화 했을 때 나온 PSNR 값과 각 이미지를 삽입한 후에 나온 PSNR로 각각의 은닉이미지를 삽입하였을 때 이미지 질을 나타낸다. 삽입전과 삽입후의 PSNR 변화는 아주 적다는 것을 알 수 있다.

표 2. 추출된 은닉이미지의 PSNR

삽입 이미지	코드북 크기	
	64	128
Baboon	26.481	27.950
Pepper	28.016	28.973

표 2 와 그림 3은 추출된 이미지의 질을 나타낸다. 표를 통해 추출된 이미지의 질이 만족하다는 것을 알 수 있다.

표 3. 9가지 외부 공격에 대한 PSNR

공격형태	Median filtering	Gaussian Blurring	Sharpening
PSNR	29.6744	24.92	28.43

공격형태	Brightness Adj.	Contrast	Scaling
PSNR	15.32	33.5125	19.6752

공격형태	Noise add.	Histogram equalization	Rotation
PSNR	22.12	27.4708	26.9939

표 3은 64크기의 코드북을 이용해 원본 이미지 Lena에 Baboon이미지를 삽입 후 9가지의 외부 공격에 대한 강인성을 실험한 후의 PSNR 값이다.[5] 그림 4는 워터마크된 이미지에 9가지 외부공격을 시행한 그림이다. 공격에서 메디안 필터링(9x9), 가우시안 블러링(r=5), 히스토그램 평활화, 밝기 조정(50%), 가우시안 노이즈 삽입(v=400), 샤프닝, 회전(40°), 크기조정(1-> 1/4->1), 명암조정(50%) 등의 외부공격 전과 후의 값 변화는 있으나 모든 이미지는 확실히 구분 가능했다.





그림 4. 외부공격: (a) 메디안필터링, (b)가우시안블러링, (c)히스토그램평활화, (d)밝기조정(50%), (e)가우시안노이즈삽입(v=400), (f)샤프닝, (g)회전(40°) (h)크기조정(1-> 1/4->1), (i)명암조정(50%)

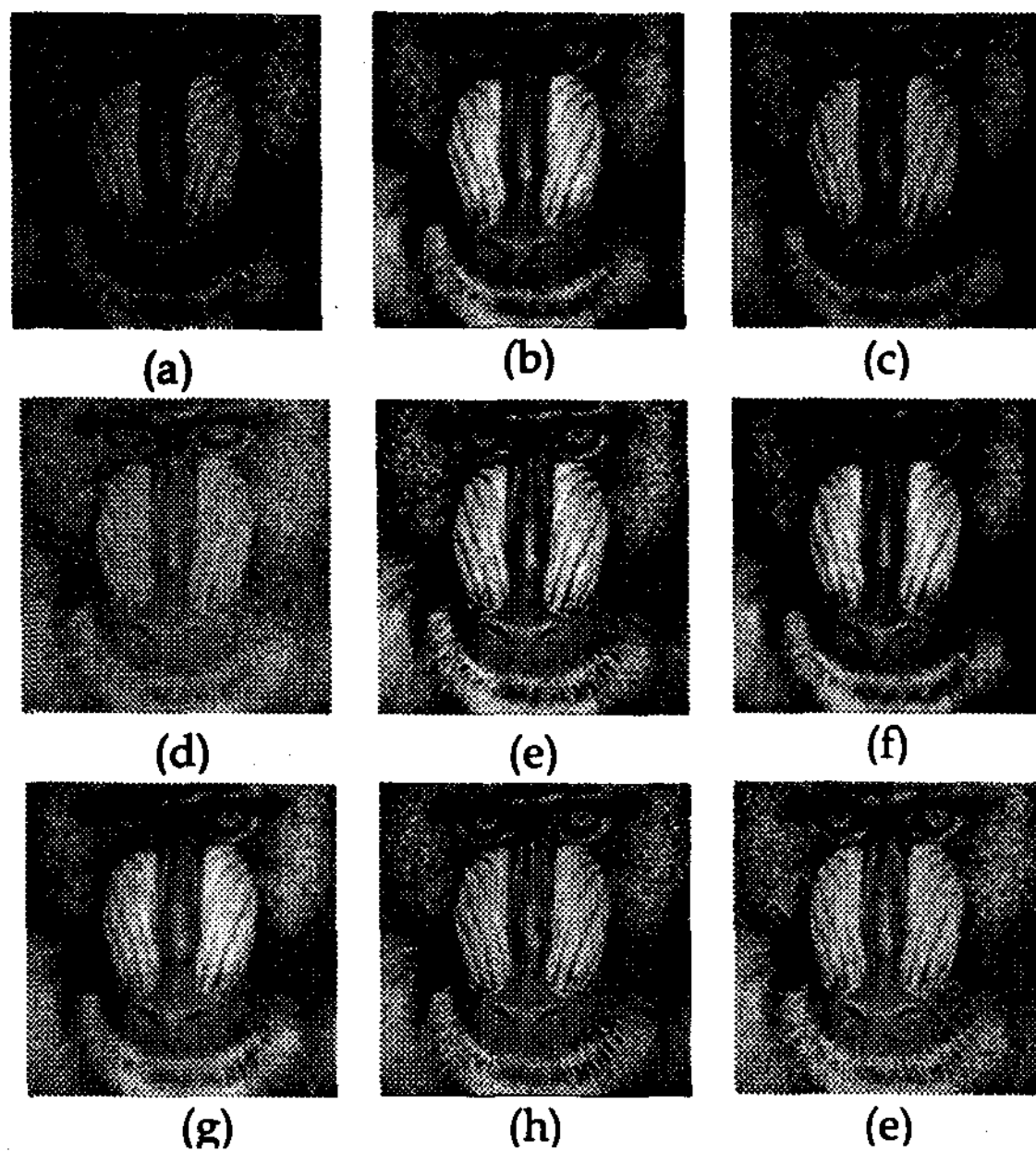


그림 5. 추출이미지: (a) 메디안필터링, (b)가우시안블러링, (c)히스토그램평활화, (d)밝기조정(50%), (e)가우시안노이즈삽입(v=400), (f)샤프닝, (g)회전(40°) (h)크기조정(1-> 1/4->1), (i)명암조정(50%)

## 6. 결 론

본 논문에서는 이미지 데이터를 은닉하기 위한 방법의 하나로 특이치 분해와 벡터 양자화를 소개하고 은닉된 후 이미지의 질의 변화를

실험을 통해 알아보았다. 소개한 방법에서 워터마킹 전과 후의 이미지 질의 변화는 크지 않고 시각적으로 확인할 수 없을 정도로 이미지를 은닉할 수 있었다. 외부공격에 대한 실험에서는 실제 실험 시 워터마크 추출 후 측정된 PSNR 값이 밝기 조정 등에서 현저히 떨어짐을 볼 수 있었다. 하지만 외부 공격 후 추출한 워터마크는 질은 떨어지지만 확연히 구분할 수 있어 다양한 저작권 보호에 사용될 수 있을 것이다.

## 참 고 문 헌

- [1] Andrew, R., H.C., Patterson, C.L., "Singular value decomposition(SVD) image coding", IEEE Trans. Comm. 24(4), pp. 425-432, 1976.
- [2] Bezdek, J C; Ehrlich, R; Full, W, "FCM: The fuzzy c-means clustering algorithm", COMP. GEOSCI. Vol. 10, no. 2-3, pp. 191-203. 1984
- [3] Gray R., "Vector quantization", IEEE ASSP Magazine, vol. 1 pp. 4-29, 1985
- [4] J.F. Yang, C.L. Lu, "Combined Techniques of Singular Value Decomposition and Vector Quantization for image Coding", IEEE Trans. Image processing, vol. 4, no. 8, Aug. 1995.
- [5] J.M. Shieh, D.C. Lou, M.C. Chang, "A semi blind digital watermarking scheme based on singular value decomposition", Elsevier B.V. Computer Standards & Interfaces, vol. 28 pp. 428-440, 2006.