

유전자 알고리즘을 적용한 영상 전처리 필터 구현에 관한 연구

A Study on Image Preprocessing Filter Using Genetic Algorithm

동성수 (디지털전자정보과), 구지훈 (인하대학교 전기공학과)
Sung-Soo Dong (Dept. of Digital Electronics & Information), Ji-Hoon Koo
(Dept. of Electrical Eng. Inha Univ.)

Key Words : Genetic Algorithm(유전자 알고리즘), Image Preprocessing Filter(영상 전처리 필터).

ABSTRACT : In this paper, we present an image filter using genetic algorithm. The filter is robust to the characteristic variance of image and noise by evolving the parameter and combination of image preprocessors properly. Also, we have adopted adaptive mutation strategy, which uses different mutation rate for specific region of chromosome. The filter is implemented on FPGA board and controlled by host PC.

1. 서론

유전자 알고리즘(genetic algorithm)은 탐색공간이 크거나 분석적으로 해를 찾을 수 없는 문제에 해결책을 제시할 수 있는 능력으로 인해 여러 응용분야에서 주목받고 있다. 이 중 하나로써 시스템의 성능을 개선하거나, 또는 예기치 못한 환경에 스스로 적응해나가는 시스템을 구성하는데 많이 연구되고 있다. 이는 시스템에 다양한 환경에 대한 자율성을 부여하기 위한 방편이며, 설계시 고려되지 않은 환경에 적응해 나갈 수 있는 시스템을 구현하는 것을 목적으로 한다. 이 논문에서는 유전자 알고리즘을 이용하여 영상의 전처리 필터 역할을 하는 적응형 시스템을 구현하였다. 영상처리에 있어서 가장 기본이 되는 과정인 전처리는 여러 분야로 나누어 지게 되고, 방법 또한 다양하게 소개 되어 왔다. 하지만 전처리 과정은 응용분야에 대해서도 의존적이고, 영상에 속하여 있는 잡영에 대해서도 매우 의존적이다. 영상처리는 일반적으로 특징추출, 영상개선, 영상복구, 영상 재구성, 영상 분석, 영상인식, 영상 압축 등으로 나눌 수 있으며, 특히 영상개선은 전처리과정으로 대부분의 다른 작업 전에 선행되는 필수적이고 빈번히 사용되는 작업이다. 현재까지 연구되어온 전처리 방법은 매우 다양하지만, 적용분야, 잡영종류, 영상의 종류 등에 따라 그 방법도 상이하게 다르며, 때로는 이들 전처리 방법의 적용분야에 따라 사람의 개입이 필요한 경우도 있었다. 이번 연구는 전처리에 있어서 의존적인 특성을 진화 알고리즘을 이용하여 상황에 따른 수동적인 필터의 선택이나 파라미터 선택을 자동적으로 수행하는 것에 목적을 두었다.

이 논문에서는 유전자 알고리즘을 이용하여 잡영종류 및 영상종류가 계속해서 변화하는 환경 속에서 스스로 대처하여 적응할 수 있는 영상의 전처리 필터를 구현한다.

2. 유전자 알고리즘을 이용한 적응형 시스템 구현

유전자 알고리즘을 이용하여 시스템의 내부 파라미터, 또는 구성 내용을 주어진 상황에 맞게

적절히 변화시켜 나가는 것을 의미하며, 유전자 알고리즘을 이용한 시스템 구성 및 파라미터 변화는 적응형 의수 제어 컨트롤러, 영상압축, 신경망과 결합된 적응형 등화기(equalizer) 등과 같은 적용이 히구치(Higuchi)등에 의해 연구되어졌다.

2. 1 유전자 알고리즘

역사적으로 1963년 홀랜드(John Holland)에 의해 창안된 유전자 알고리즘은, 자연계의 생물 진화과정에 있어서 어떤 세대(generation)를 형성하는 개체(individual)들의 집합, 즉 개체군(population) 중에서 환경에 대한 적합도(fitness)가 높은 개체가 높은 확률로 살아남아 재생(reproduction)할 수 있게 되며, 이때 교배(crossover) 및 돌연변이(mutation)로서 다음 세대의 개체군을 형성한다는 생물 진화의 원리로부터 착안된 알고리즘으로서 확률적 탐색이나 학습 그리고 최적화를 위한 한 가지 기법이라고 간주할 수 있다. 포겔(Fogel)은 진화 방식의 모형 화를 시도하여 간단한 유산상태 시스템의 최적화를 수행하기도 하였다.

일반적인 유전자 알고리즘 연산자로는 선택(selection), 교배(crossover), 돌연변이(mutation)가 근간을 이루고 있으며, 적용 대상에 따라, 세부적인 방법이 조금씩 변화하여 왔다. 이 논문에서 사용된 유전자 알고리즘은 Fig. 1 과 같다.

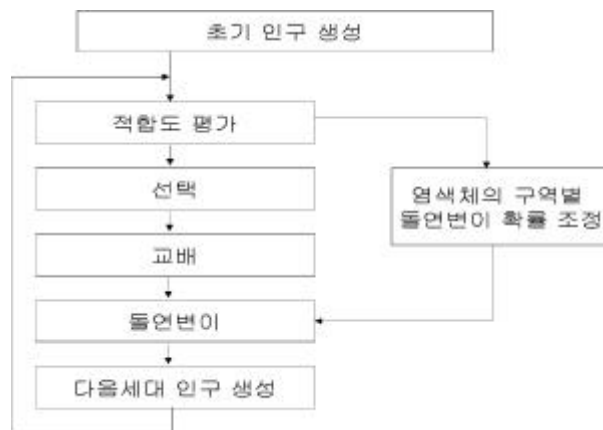


Fig. 1 Genetic algorithm

초기 인구(population)생성은 염색체(chromosome)의 길이가 1,200bit을 갖는 후보해(individual)를 불규칙하게 생성해 낸다. 적합도 평가는 해의 수렴 방향을 정의하는 것으로 각각의 후보해들이 얼마나 해에 접근하고 있는가의 판단 기준이다. 이 논문에서는 여러 종류의 잡영 종류에 대해서 원 영상과 유사하도록 잡영을 제거되는 방향으로 설정하였다.

선택(selection)은 이전 인구 중에서 적합도가 높은 후보해를 다음 세대 생성에 사용 하기 위해 수행하는 연산자로서, 여기서는 일반적인 토너먼트식 방법을 사용하였다. 교배(crossover)는 6개의 교배 점을 불규칙적으로 생성해서 교배확률을 적용하여 두 개의 염색체(chromosome)를 교배한다. 교배 확률은 일반적으로 0.7에서 0.9사이의 고정된 값을 사용하며, 값이 너무 작으면 이전 세대의 인구(population)중 우수한 해는 교배에 의한 파괴 없이 그대로 다음 세대에 전달되지만 해에 접근하는 시간이 오래 걸리며, 값이 너무 크면 그 반대의 효과가 있다. 돌연변이(mutation)는 각 비트(bit)에 대하여 해당 블럭의 돌연변이 확률을 적용하여 해당 비트(bit)의 염색체 내용을 0 에서 1, 1 에서 0 으로 전환한다.

일반적으로 돌연변이 확률은 후보해(individual)의 염색체(chromosome)의 각각의 비트(bit)에 대하여 균일한 돌연변이 확률을 결정하는 것으로, 확률이 크면 해의 접근 속도는 느려지지만 국소 해에서 탈출이 쉽고, 확률이 너무 작으면 해 접근 속도는 빠르지만 인구(population)가 모두 균일해져서 국소 해에서의 탈출이 어렵다. 보통 일반적인 경우는 고정된 작은 값이 사용된다. 여기서 사용될 염색체의 구역별 돌연변이 확률 조정은 이후 진행될 돌연변이의 확률을 조정해주는 것으로, 이 논문에서 사용된 돌연변이 확률은 8bit 단위의 블록으로 나누어서 각각에 다른 확률을 적용하는 기법을 사용하였다. 처음에는 모두 같은 값으로 시작하지만 특정영역의 돌연변이로 인해서 이전에 비해 적합도가 높아진다면, 이 부분의 돌연변이 확률을 높이고, 적합도가 떨어진다면 돌연변이 확률을 내리는 방법을 사용하였다. 이러한 방법을 사용하여 돌연변이에 의한 진화는 긍정적인 방향으로 가속시킨다는 이론이다. 또 진행 중 구간별 돌연변이 확률의 평균값을 계산하여 평균이 0.5~0.0001 사이의 값을 갖도록 조정하였다.

2. 2 적응형 시스템 구조

유전자 알고리즘을 이용하여 상위 단위의 기능블럭 조합문제와 하위단위의 기능블럭의 내부 파라미터들을 동시 진화가 가능한 구조를 구성하였으며, 영상처리를 목적으로 한 시스템이므로 각각의 기능블럭은 여러 가지 전처리 필터 중 하나로 정의하였다. Fig. 2 는 전체 시스템 구조를 나타내고 있다. 사용된 하드웨어는 Xilinx VertexE2000과 2Mbyte의 서로 독립적인 4개의 SRAM Bank, PCI 인터페이스를 갖고 있는 Celoxica의 RC1000-PP 보드를 사용하였다. 한번에 처리해야할 데이터의 양이 많다는 영상처리의 특성상 보드상의 SRAM을 영상 버퍼로서 사용하게 된다.

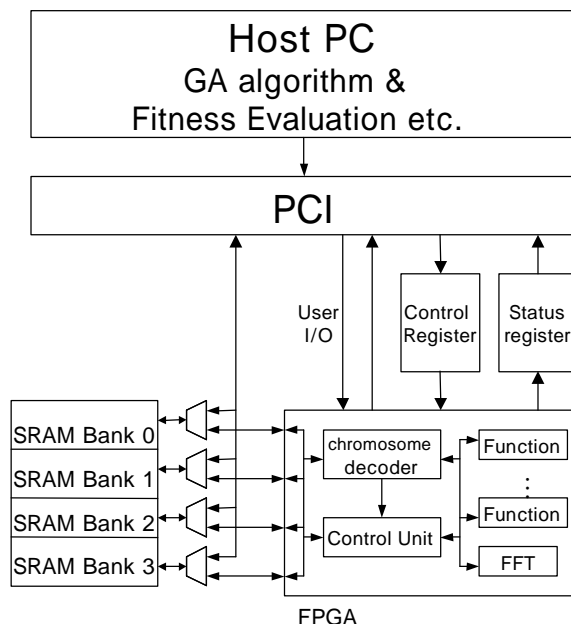


Fig. 2 System architecture

제어 레지스터(control register)와 상태 레지스터(status register)는 Host PC상의 프로그램과 FPGA와의 상호 통신(handshaking)을 위해 사용되며, 영상 데이터, 적합도 평가

결과 및 염색체 데이터는 보드상의 SRAM 경유, 32bit 버스(bus)를 이용해서 FPGA와 Host PC 사이를 이동한다.

유전자알고리즘 연산결과, 나온 염색체(chromosome)는 염색체 해독기(chromosome decoder)를 거쳐 유전형(genotype)에서 표현형(phenotype)으로 변하게 된다. 염색체의 비트 열(bit string)은 각각 기능블럭의 파라미터를 결정하고 기능블럭들간의 실행 순서 조합을 결정하게된다.

기능블럭은 모두 5개 사용하였으며, 각기 특성과 파라미터의 종류 및 개수, 그리고 파라미터마다 필요로 하는 값의 범위가 다르다. 해독기(decoder)를 통해서 염색체는 해당범위의 필요로 하는 파라미터 형태로 바뀌게 된다.

염색체 길이는 기능블럭들의 파라미터로 모두 1,168bit, 기능블럭들간의 조합 문제를 풀기 위해 20bit, 또 32bit 버스(bus) 폭에 맞는 길이를 만들기 위해 추가로 12bit, 모두 1,200bit(150byte)로 설정되었다.

$$\begin{aligned} ChromosomeLength &= 1,168(Parameter) + 20(combination) + 12(dummy) \\ &= 1,200bit \quad (150byte) \end{aligned}$$

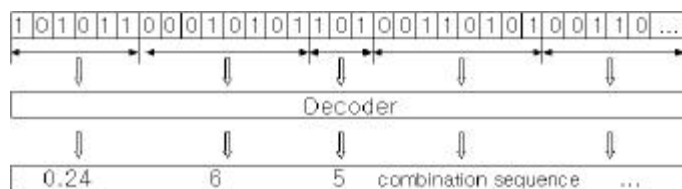


Fig. 3 Chromosome decoder

2. 3 기능블럭의 정의 및 파라미터

(1) 첫 번째 기능블럭, 준동형의 필터(Homomorphic filter) :

디지털 영상은 조명(illumination, $i(x,y)$)과 반사(reflectance, $r(x,y)$) 성분으로 구성되는 광학(optical) 영상이다. 이 필터의 목적은 이러한 광학 영상의 반사요소는 향상시키고, 조명요소는 감소시키기 위해서 사용된다.

영상 $f(x,y)$ 를 조명-반사 모델(illumination - reflectance model)로 표현하면 식(1)과 같이 표현된다.

$$f(x,y) = i(x,y)r(x,y) \tag{1}$$

푸리에(fourier) 변환에 의한 조명과 반사성분의 분리를 위해 $z(x,y) = \ln r(x,y)$ 라고 하고, 함수(function) $H(u,v)$ 을 사용하면, 식(2)으로 표현되며, 결과 영상은 식(3)과 같이 나타난다.

$$\begin{aligned} z(x,y) &= \ln i(x,y) + \ln r(x,y) \\ S(u,v) &= H(u,v)Z(u,v) \end{aligned}$$

$$= H(u, v)I(u, v) + H(u, v)R(u, v) \quad (2)$$

$$\begin{aligned} g(x, y) &= \exp[s(x, y)] \\ &= \exp[i'(x, y)] \exp[r'(x, y)] \\ &= i_0'(x, y) r_0'(x, y) \end{aligned} \quad (3)$$

식(2)에서 볼 수 있듯이 $H(u, v)$ 을 적절하게 설계함으로써 조명(illumination)성분을 감소시킬 수 있다. $H(u, v)$ 의 내부 파라미터로써 염색체(chromosome)의 2byte를 할당하였다.

(2) 두 번째 기능블럭, 다중 대역 필터(Multi-band filter) :

주파수 영역에 나타나는 잡영을 제거하기 위해서 사용된다. 잡영의 형태가 영상의 주파수 영역 별로 달라진다고 가정했을때, 잡영을 제거하기 위해서는 가변적인 밴드의 갯수와 밴드폭을 잡영의 형태에 따라 가변적으로 나타내 주어야 한다.

$$f(x, y) \rightarrow \text{FFT} \rightarrow H(u, v) \rightarrow \text{IFFT} \rightarrow g(x)$$

$H(u, v)$ 의 형태는 다음과 같다.

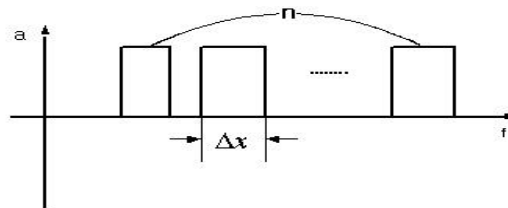


Fig. 4 Multiband-filter

각 밴드의 시작점과 끝점을 기술하기 위해 각각 2byte, 즉 밴드 하나 당 4byte를 할당하였고, 밴드의 최대 개수를 35개 이하로 정의하였다. 모두 140byte가 할당된다.

(3) 세 번째 기능블럭, 메디안 필터(Median filter) :

해당 마스크 창에 속하는 픽셀들의 중간 값을 선택하는 비선형 필터로서 임펄스 잡영 (impulse noise)의 제거에 효과적이다. 여기서 사용될 메디안 필터(median filter)의 마스크 종류는 Fig. 5 와 같다. 사용되는 파라미터는 마스크 종류를 결정하기 위해 1byte가 할당되었다.

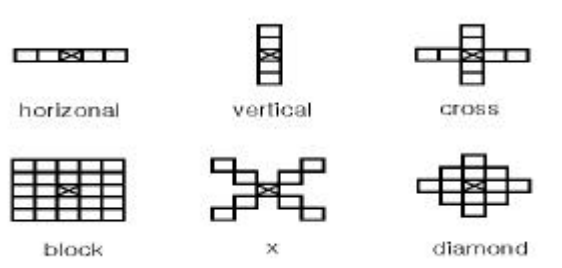


Fig. 5 Median filter mask

(4) 네 번째 기능블럭, 지역 증대 필터(Local enhancement filter) :
 영상에 대해서 전체 및 부분적인 정보의 통계적 방법으로 영상을 개선하는 방법이다.

$$G(x, y) = A(x, y)[f(x, y) - m(x, y)] + m(x, y) \quad (4)$$

where $A(x, y) = kM/\sigma(x, y)$

$m(x, y)$: 잡영

$\sigma(x, y)$: 마스크 영역의 표준 편차

사용된 파라미터는 마스크의 가로, 세로방향 크기를 결정하기 위해 각각 1byte, k 값을 위해 1byte. 모두 3byte가 사용되었다.

(5) 다섯 번째 기능블럭, 히스토그램 균등 필터(Histogram equalization filter) :
 영상의 회색 정도(gray level)의 분포가 집중되어 있는 부분을 평준화를 이용하여 확률적 분포를 고르게 한다. 전체적인 히스토그램 균등화(global histogram equalization)를 하였으며 사용되는 파라미터는 없다.

3. 실험방법

Xilinx VertexE2000과 2Mbyte의 서로 독립적인 4개의 SRAM Bank, PCI 인터페이스를 갖고 있는 Celoxica의 RC1000-PP 보드를 사용한 실험에서, 염색체(chromosome) 길이는 1200 bit으로 필터 조합 순서와 각각의 파라미터를 결정한다. 그렇게 결정된 파라미터 및 필터조합 순서는 잡영이 섞인 영상의 처리 결과에 대해서 적합도가 평가(evaluation)된다.

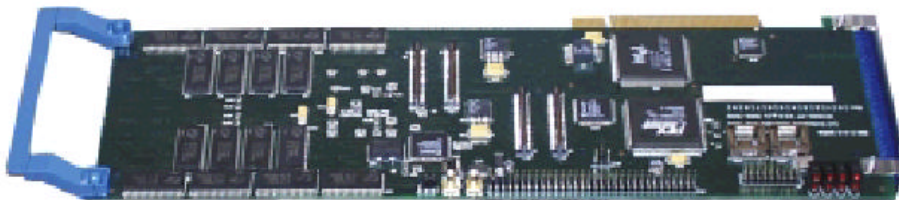


Fig. 6 System architecture hardware

Fig. 6 과 같이 시스템 구조(system architecture)를 하드웨어 상에 구현하여 실험을 하였다. Host PC의 프로그램에서 PCI 보드상의 램의 지정된 주소에 테스트영상을 넣어주면, 보드상의 FPGA에서 영상을 처리해서 다시 보드 상 램의 다른 주소에 영상을 저장한다. 그 후, 다시 Host PC의 프로그램에서 영상을 읽어 적합도를 평가를 한다. 호스트 PC상의 유전자 알고리즘 연산결과 나온 염색체(chromosome)는, 보드상의 FPGA로 전송, 염색체 해독기(chromosome decoder)를 거쳐 유전형(genotype)에서 표현형(phenotype)으로 변하게 된다. 변형된 표현형은 각각 기능블럭의 파라미터를 결정하고 기능블럭들간의 실행 순서 조합을 결정하게된다. Host PC상의 프로그램과 FPGA와의 상호 통신(handshaking)은 제어 레지스터(control register)와 상태 레지스터(status register)로 한다. 영상 데이터, 적합

도 평가 결과 및 검색체 데이터는 보드상의 SRAM 경유, 32bit 버스(bus)를 이용해서 FPGA와 Host PC 사이를 이동한다.

4. 실험결과

실험에 사용된 촬영은 공간영역에서는 랜덤노이즈, 가우시안 노이즈, 주파수 영역에서는 랜덤 노이즈가 영상에 곱해지는 형태를 사용하였다.



Fig. 7 (A) Additive noised image (B) Enhancement image

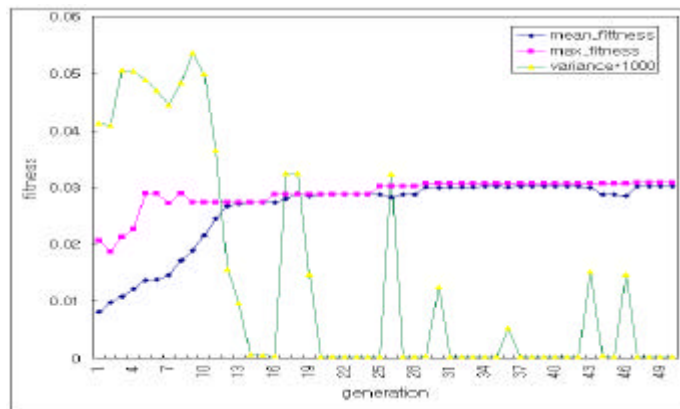


Fig. 8 Genetic algorithm result graph

Fig. 7 은 공간영역에서의 랜덤 노이즈가 혼합된 영상의 처리 결과이다.

최적 후보해의 1200bit 중 필터 조합을 나타내는 부분만을 살펴보면, “ ... **322b5d** ... ”로 써, Fig. 9 와 같은 필터 조합이 나타나게 된다. 임펄스 잡영 제거에 강한 메디안 필터 (Median Filter)가 다른 파라미터를 갖고 중복해서 두 번 쓰인 것을 볼 수 있다.

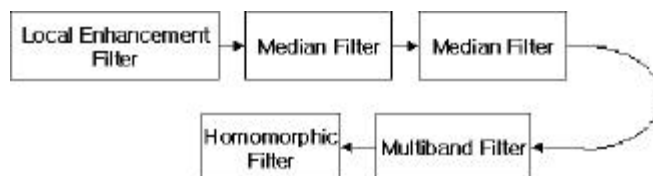


Fig. 9 Optimal filter combination

다음 Fig. 10 은 주파수 영역의 잡영이 섞였을 경우 처리 결과이다.



Fig. 10 (A) Multiplicative noised image (B) Enhancement image

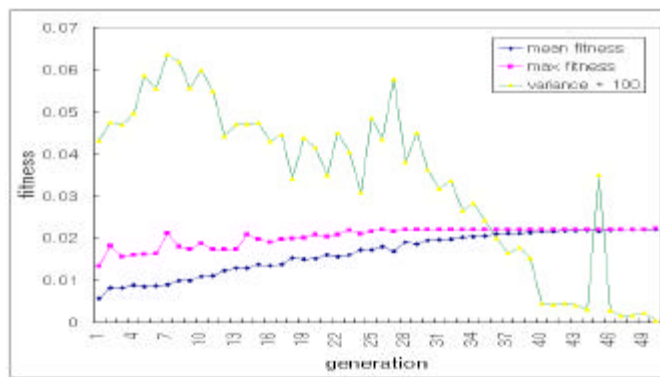


Fig. 11 Genetic algorithm result graph

Fig. 11 은 Fig. 10 실험의 수행결과 그래프를 보여주며 50세대 후의 후보해들은 둘 이상의 다중 대역 필터(multi-band filter) 조합을 보여주고 있다. 이는 주파수 공간상의 임펄스 노이즈로 존재하는 잡영을 없애기 위한 적응 형태로 볼 수 있다.

두 실험 모두 초기 인구의 수는 100개, 세대수는 50, 교배확률은 0.8, 초기 돌연변이 확률은 0.05이다. 필터 조합의 결정은 한가지 필터를 중복해서 사용하거나 사용되지 않은 필터가 생기는 경우도 있기 때문에 잡영이나 영상의 종류에 따라 수행 시간에 모두 다르다. FPGA는 30MHz, PC는 550MHz에서 수행하였다.

5. 결론

이 논문에서는 시스템의 진화성과 환경변화에 대한 자율성을 주기 위해서 유전자 알고리즘을 사용하였고, 이는 설계시 고려되지 않은 환경에 적응해 나갈 수 있는 시스템을 구현하는 것을 목적으로 한다. 그 적응으로 잡영의 종류가 변화하는 경우에 있어서의 적응형 전처리 필터를 구현, 실험하였다. 진화 모듈은 XCV2000E FPGA 상에서 구현되었으며, 컴퓨터(computer)와의 인터페이스의 용이성을 위하여 PCI 방식의 인터페이스를 갖고 있는 RC1000 보드를 이용하였다. 여기서는 진화모듈내의 각 기능(function)을 영상의 잡영 처리를 위한 다양한 종류의 전처리 단계 필터로 규정하였으며, 준동형(homomorphic), 메디안(median), 지역 증대(local-enhance), 다중 대역(multi-band), 히스토그램 등화 필터

(histogram-equalization filter) 등이 이에 속한다. 그 결과 필터 조합의 결정은 한가지 필터를 중복해서 사용하거나 사용되지 않은 필터가 생기는 경우도 있는 등 주파수 공간상의 임펄스 노이즈로 존재하는 잡영을 없애기 위한 적응 형태로 필터 조합이 나타남을 볼 수 있었다. 현재는 전 처리부 만을 단독 실험하였기 때문에 잡영을 제거해서 영상을 복구하는 목적이었으나 전처리 후에 영상압축, 인식, 분석 등의 목적이 뒤에 놓인다면, 이러한 목적을 위하여 적용할 수 있는 시스템의 모델링이 가능할 수 있다.

참고문헌

- (1) Rafael C. Gonzalez, 1992, *Digital Image Processing*, ADDISON-WESLEY.
- (2) Melanie Mitchell, 1996, *An Introduction to Genetic Algorithm*, MIT press.
- (3) Mitsuo Gen, 2000, *Genetic Algorithms & Engineering Optimization*, Wiley-Interscience.
- (4) J. R. Parker, 1997, *Algorithms for Image Processing and Computer Vision*, Wiley & Sons.
- (5) Ikuo Yoshima, 2000, *Kernel-based Pattern Recognition Hardware*, Proceedings The Second NASA/DoD WorkShop.