

유전자 알고리즘의 퍼지 결정 함수를 이용한 FGNN 구현 Hardware Implementation of FGNN using Fuzzy Decision Function of the Genetic Algorithm

변오성 · 문성룡

Oh-Sung Byun and Sung-Ryong Moon

원광대학교 전자공학과

요 약

본 논문에서 임의의 데이터가 입력되면 기준 영상 중에서 가장 유사도가 큰 영상을 찾아 국부 승리자로 선택하고, 그 국부 승리자 중에서 전체 승리자를 선택하여 최종 출력값을 얻는 계층적 FGNN(Fuzzy Genetic Neural Network)을 제안하고, 이에 하이브리드 퍼지 소속함수와 유전자 알고리즘을 적용하였다. 하이브리드 퍼지 소속함수는 입력 값을 0~1 사이의 값으로 함으로써 시스템의 속도를 빠르게 하고 유전자 알고리즘은 입력값을 일정한 오차 이내로 하여 최적의 영상을 얻도록 하였다. 위의 계층적 FGNN 알고리즘을 회로 설계 및 검증하였다. 또한 제안한 FGNN을 이용하여 영상에 포함된 잡음을 제거하고, 이와 유사한 구조를 가진 FDNN(Fuzzy Decision Neural Network) 성능보다 FGNN의 성능이 우수함을 여러 가지 영상을 통하여 확인하였다. 또한 모의 실험 결과 영상에 대한 평균자승오차(MSE : Mean Square Error)를 비교하였으며, 그 결과 하이브리드 퍼지 함수와 유전자 알고리즘을 적용한 FGNN이 메디안 필터, OC, CO, FDNN 등에 비해 우수함을 확인하였다. FGNN 알고리즘을 Top-Down 방식으로 VHDL(VHSIC Hardware description Language)을 이용하여 코딩(Coding)하고, Synopsys 툴을 이용하여 하드웨어를 설계하였다. 이 알고리즘의 하드웨어는 총 5개의 블록으로 가지고 있고 각각의 블록은 파이프라인 형태로 구성하고, 이는 Synopsys 툴을 이용하여 동작 및 성능을 검증하였다.

ABSTRACT

In this paper, it can be proposed in order to find the optimal image to the hierarchical FGNN(Fuzzy Genetic Neural Network) which is obtained the last output value selection to a global winner among local winners being selected to find the largest similarity image in the basis images if any data are input. And it apply the hybrid fuzzy membership function and genetic algorithm, where the hybrid fuzzy membership function is fast a speed of system as making the input values by the values [0, 1], and the genetic algorithm can be obtained the optimal image as making the input values in the fixed error. And it is designed of the circuit and verified the above hierarchical FGNN algorithm. Also, the included noise in an image is removed using the proposed FGNN, and it is found through the images that the performance of FGNN is superior to FDNN(Fuzzy Decision Neural Network) which has the similar structure. As a result of the comparison by MSE for the simulation result images, the superiority of the proposed FGNN to Median filter, OC, CO, FDNN and so on is identified. FGNN hardware is coding using VHDL(VHSIC Hardware description Language) by Top-Down method, and consists of the five blocks, also each block have the pipeline form. And this hardware can be verified operation and performance by using Synopsys tool.

1. 서 론

컴퓨터, 집적회로, 통신 및 신호처리 기술의 발달과 더불어 멀티미디어의 정보화에 따른 영상정보의 데이터를 처리, 전송, 저장하는데 시간과 비용이 수반되며, 또한 형성, 전송, 혹은 저장되는 과정에서 변형, 화질의 열화로 인하여 손실된 정보의 복원과 화질 개선에 대한 연구가 진행되고 있다[1,2].

이 논문은 1999년도 원광대학교의 교비지원에 의해서 연구됨.

그리고 영상처리 응용에 있어서 효과적으로 잡음을 제거할 수 없으며, 또한 잡음을 제거하기 위하여 선형 필터를 사용할 경우 신호에 중속된 잡음이 존재 할 때는 성능이 저하된다. 따라서 영상 처리에 있어서 비선형 필터링 기술이 대두되고 있으며, 이런 필터의 종류에는 평활화 필터와 메디안(Median)필터, 그리고 영상에 존재하는 기하학적 특성을 갖는 형상학(Morphology)필터, DWW(Dynamic Weight Warping)를 이용한 필터 등이 있다. 평활화 필터와 메디안 필터는 적은 잡음에 있어서 효과적이지만 일정 이상의

잡음이 포함되면 거의 잡음을 인식하지 못하고 최적의 영상을 구현하지 못하는 문제점을 가지고 있다. 그리고 기하학적 특성을 가지고 있는 형상학 필터는 잡음으로 인한 영상의 번짐(Blurring) 효과가 일어나고, 또한 FDNN(Fuzzy Decision Neural Network) 필터는 잡음과 거의 같은 에지 부분을 잡음으로 인식하고, 또한 번짐 효과가 발생된다. 이에 비해 가장 최근에 발표된 신경망과 퍼지 알고리즘을 이용한 필터가 나오는데 여기 신경망의 지도학습 모델에는 홉필드(Hopfield)망, 역전파 신경망, 퍼셉트론, DBNN(Decision Based Neural Network)등이 있다. 신경망의 지도학습은 검색단계(Retrieving Phase)와 학습단계(Learning Phase)로 구분되고, 검색단계는 출력값에 의해 영상이 어디에 속하는지를 결정하게 되고, 출력값은 판별함수(Discriminant Function), 즉 가중치(Weight)에 의해 결정되어지고, 학습단계에서의 가중치는 영상을 올바르게 분리하기 위해 학습과정이 이루어지며 지도자는 결정이 올바르게 되었는지 그리고 적절하게 조정이 이루어졌는지를 판별하게 된다. 또한 지도학습은 영상들을 정확하게 분류하기 위한 가중치들을 학습시키고, 최상의 판별함수에 의해서 결정 경계치를 결정한다[3]. 그리고 FDNN 알고리즘은 삼각형 모양 퍼지 함수와 결정함수에 DWW를 적용하였는데, 여기서 삼각형 모양 퍼지 함수를 적용할 경우 공복이나 중복 현상이 발생을 하고, 그리고 DWW 결정함수를 적용할 경우 외부에서 임계값을 결정하거나 국부 최적으로 빠지는 경우가 발생을 한다. 그래서 이러한 현상을 보완하기 위해서 본 논문에서는 FGNN을 제안한다.

본 논문에서는 사다리꼴 모양의 퍼지 함수와 삼각형 모양의 퍼지 함수를 합한 하이브리드 퍼지 함수를 적용한다. 또한 최적의 승리자를 찾기 위해 유전자 알고리즘을 적용함으로써 잡음, 에지 부분의 보상 및 복원을 위한 새로운 구조의 FGNN(Fuzzy Genetic Neural Network)을 구현한다.

본 논문의 내용에서 II장은 하이브리드 퍼지 알고리즘 및 유전자 알고리즘의 기술적 배경을 서술하고, II장은 구현된 FGNN 알고리즘의 배경과 FGNN의 하드웨어 설계 즉, 퍼지 블록, 평가자 블록, 신경망 블록, 비퍼지 블록, 유전자 블록을 VHDL(VHSIC Hardware description Language)을 이용한 회로구현하고 설명한다. 그리고 IV장에서는 FGNN의 시스템을 Synopsys 툴을 이용하여 하드웨어의 성능을 검증 및 FGNN을 MATLAB을 이용한 컴퓨터 모의 실험을 통하여 잡음이 섞인 영상에 대한 복원에 대해서 FDNN과 OC(Opening Closing), 메디안 필터들과 비교하고, 결과를 확인한다. 또한 하이브리드 퍼지함수를 적용한

FGNN을 실시간 처리가 가능하도록 회로를 설계하고, 하드웨어를 구현하고자 한다.

2. 퍼지 알고리즘과 유전자 알고리즘

2.1 하이브리드 퍼지 알고리즘

삼각형 퍼지 함수의 빈곳(Emptiness)과 혼란(Confusion)등의 단점을 보완하기 위하여 적용된 하이브리드 퍼지 함수의 구성은 사다리꼴형 퍼지 함수와 거의 유사한데, 이는 사다리꼴형 퍼지 함수와 삼각형 퍼지 함수를 결합한 퍼지 규칙에 파라미터를 적용하여 하이브리드 방식의 퍼지 규칙 및 퍼지 소속 함수를 설명한다[4,5].

먼저, k 의 규칙은 $(X_1$ 은 A_1^k) 그리고 $(X_2$ 는 A_2^k) ... 그리고 $(X_r$ 은 $A_r^k)$ 이면, $(Y$ 는 $B^k)$ 이다. 여기서 $k=1, 2, \dots, r$ 이고, r 은 규칙의 수를 나타내며, 입력(X_1, \dots, X_r)과 출력(Y) 변수는 모두 퍼지집합으로 표현된다. 입력 변수 X_i 는 모든 시스템의 퍼지 규칙에 적용되고 k 인덱스에는 독립적인데, A_1^k, \dots, A_r^k 와 B_k 는 퍼지집합에 의해 표현되며, k 는 규칙의 인덱스를 나타낸다. 그리고 퍼지 전체의 기본적 형식은 다음과 같다.

$$p : X \text{는 } A \text{이다.}$$

여기서 X 는 전체집합 U 로 정의된 삼각형 퍼지 변수이고, A 는 사다리꼴형 퍼지로 정의된다. 퍼지 소속함수값 v_{M_m} 은 식 (1)에서와 같이 Max-Min을 이용하여 계산한다.

$$v_{M_m} = \omega / (\omega + d) = \omega / (\alpha_A + \beta_x) = 1 - d / (\omega + d) = 1 - d / (\alpha_A + \beta_x) = 1 / (1 + d/\omega) \quad (1)$$

여기서 d 는 $\min(\text{core}(X))$ 와 $\max(\text{core}(A))$ 사이의 거

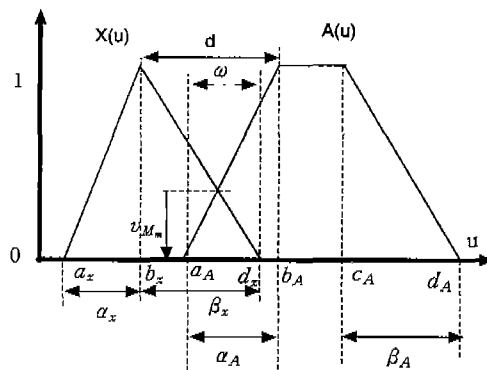


그림 1. 결합형 퍼지 소속 함수(사다리꼴형과 삼각형 퍼지 소속함수)

Fig. 1. Matching of the fuzzy membership function

리이고, ω 는 $\min(\text{supp}(X))$ 와 $\max(\text{supp}(A))$ 사이의 거리이다. α_x 와 β_x 는 각각 $\text{Lbound}(A)$ 와 $\text{Rbound}(X)$ 와 같으며, 식 (1)을 그림 1로 나타내었다.

그리고 Max-product를 적용하면, v_{M_m} 은 식 (2)에 의해서 얻게된다.

$$v_{M_p} = v_{M_m}^2 \quad (2)$$

X 는 일반적인 경우에 삼각형 퍼지 변수($b_x = c_x$)이고, 결합 연산을 수행할 때는 사다리꼴형 퍼지 변수($b_x \neq c_x$)로 식 (3)과 같다[6,7].

$$v_{M_m} = \begin{cases} b_x \leq b_A - (\alpha_A + \beta_x) \text{ 혹은 } b_x \geq c_A + (\beta_A + \alpha_x) \rightarrow 0 \\ b_A \leq b_x \leq c_A \rightarrow 1, \\ b_x < b_A \rightarrow 1 - (b_A - b_x)/(\alpha_A + \beta_x), \\ b_x > c_A \rightarrow 1 - (b_x - c_A)/(\beta_A + \alpha_x) \end{cases} \quad (3)$$

또한 식 (3)을 하이브리드 방식으로 변환하기 위해서, 식 (3)에 식 (4)를 적용하다.

$$\begin{aligned} \alpha'_A &= \alpha_A + \beta_x \\ \beta'_A &= \beta_A + \alpha_x \end{aligned} \quad (4)$$

그래서 v_{M_m} 의 X 와 X' 는 각각 A 와 A' 형태로 변환될 수 있다.

그림 2는 본 논문에서 적용한 하이브리드 방식 퍼지 함수로서 사다리꼴형과 삼각형 퍼지 함수를 합성한 것이다. 여기서 $X' = X(u) = \wedge_c(u; 0, b_x, 0)$ 은 비퍼지 단일 변수이고, $A' = A(u) = \Pi_d(u; \alpha'_A, b_A, c_A, \beta'_A)$ 는 하이브리드 사다리꼴형이며, 식 (5)와 같이 표현된다.

$$v_{M_m} = \begin{cases} b_x \leq b_A - \alpha'_A \text{ 혹은 } b_x \geq c_A + \beta'_A \rightarrow 0, \\ b_A \leq b_x \leq c_A \rightarrow 1, \\ b_x < b_A \rightarrow 1 - (b_A - b_x)/\alpha'_A, \end{cases}$$

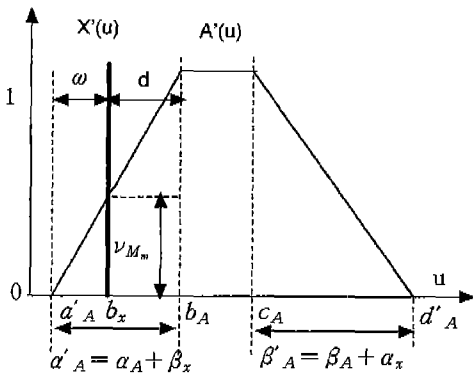


그림 2. 하이브리드 퍼지 함수
Fig. 2. The hybrid fuzzy function

$$b_x > c_A \rightarrow 1 - (b_x - c_A)/\beta'_A \quad (5)$$

퍼지 함수의 계산은 새로운 사다리꼴형 레벨 A' 에 대한 Π_s 를 이용하여 쉽게 결합된다.

$$A' = A'(u) = \Pi_s(u; a'_A, \alpha'_A, \beta'_A, d'_A)$$

여기서

$$\begin{aligned} \alpha'_A &= b_A - \alpha'_A \\ d'_A &= c_A + \beta'_A \end{aligned} \quad (6)$$

과 같다.

그래서 하이브리드 퍼지 함수의 v_{M_m} 은 식 (7)과 같이 표현된다.

$$v_{M_m} = \begin{cases} b_x \leq a'_A \text{ 혹은 } b_x \geq d'_A \rightarrow 0, \\ a'_A + \alpha'_A \leq b_x \leq d'_A - \beta'_A \rightarrow 1, \\ b_x < a'_A + \alpha'_A \rightarrow (b_x - a'_A)/\alpha'_A, \\ b_x > d'_A - \beta'_A \rightarrow (d'_A - b_x)/\beta'_A \end{cases} \quad (7)$$

2.2 유전자 알고리즘

생물학적 진화를 기반으로 부모로부터 진화를 이용하여 시간이 지나감에 따라 최적의 염색체를 찾는 방법으로 코딩, 집단으로부터의 탐색, 보조정보의 불필요, 랜덤화된 연산자들의 직접사용이라는 특성으로 인한 복잡성을 최적화에 사용하고, 그리고 초기 집단으로부터 시간이 경과함에 따라 개선된 집단을 발생시키는 유전자 알고리즘을 이용하여 자동으로 임계값을 설정하여 오차를 줄임으로 최적의 값을 얻을 수 있다[8,9]. 초기 집단으로부터 시간이 경과함에 따라 개선된 집단을 발생시키는 유전자 알고리즘은 유전 연산자로 구성되는데, 많은 실용적인 문제에 있어서 좋은 결과를 낳는 단순한 유전알고리즘은 재생(Reproduction), 교배(Crossover), 돌연변이(Mutation) 연산으로 구성된다.

유전알고리즘은 파라미터 집합을 나타내는 코드화된 문자열 집단에서 동작하며 재생, 교배와 돌연변이는 새로운 문자열 집단을 만들어내기 위해 연속적으로 문자열 집단에 적용된다. 이들 연산자는 난수 발생, 문자열 복제와 부분 문자열 교환이라는 단순한 형태이지만 그 단순성에도 불구하고 뛰어난 탐색기능을 발휘하는데 이는 어떤 문자열 위치에서 유사한 문자열 부분집합에 의하여 설명된다. 문자열로 이루어진 집단 내에 포함된 유사한 문자열은 재생, 교배, 돌연변이 하에서 다음 세대에서 식 (8)에 정의된 수만큼 복제된다.

$$m(H, t+1) \geq m(h, t) \frac{f(H)}{f} \left(1 - p_c L^{-1} - O(H) p_m \right) \quad (8)$$

여기서 $m(H, t)$ 는 시간 t 에서 집단내의 유사한 문자열 H 의 수, $f(H)$ 는 시간 t 에서 유사한 문자열 H 의 평균 적합도, \bar{f} 는 전체 집단의 평균 적합도, $O(H)$ 는 문자열에서 0과 1의 자리 수, $\delta(H)$ 는 유사한 문자열 길이, L 은 문자열의 길이, P_o, P_m 은 각각 교배율과 돌연변이율이다.

Holland는 이를 내재적 병렬성(Implicit Parallelism)이라 불렀다. 이러한 내재적 병렬성은 유전 알고리즘이 다양한 최적화 문제에 효과적으로 적용될 수 있도록 해준다. 3개의 유전 연산자를 사용한 단순 유전 알고리즘은 초기 파라미터 집단을 임의로 발생하고 적합도에 비례하는 수의 문자열을 재생 연산자에 의해 교배집단 안으로 복제한다. 교배집단 안으로 재생된 문자열은 교배 연산자와 돌연변이 연산자에 의해 새로운 문자열을 발생하고 이 과정이 진행됨에 따라 집단은 보다 적합한 문자열로 구성된다[10].

본 논문에서는 데이터 값을 오차이내 값으로 만들기 위해 반복 수행을 하여 최적의 영상을 얻기 위해 기본적 유전 알고리즘을 적용하였다. 여기서는 퍼지 평가 블록을 거친 3×3 영상과 신경망을 거친 비퍼지화 블록의 결과 값을 입력으로 받아 이 값을 교배 및 돌연변이 연산자를 이용하여 최적의 영상을 얻는 부분이다. 유전자 알고리즘은 두 가지의 변화 과정을 가지고 있으며, 이 변화 과정은 교배와 돌연변이이고, 이 변화 과정의 파라미터에 일정한 변화를 주어서 최적의 영상을 얻는 것이다. 실제로 본 논문은 두 영상의 결과 값 중 3 bit를 서로 교환함으로써 교배를 처리하였고, 교배를 한 후 그 두 영상 값에서 최적의 영상 값을 찾은 후 이 값의 일부를 바꿈으로 최적 값을 얻을 수 있도록 돌연변이 과정을 수행하였다. 이 논문에서는 최적의 영상을 얻기 위하여 다음과 같은 절차를 적용하였다.

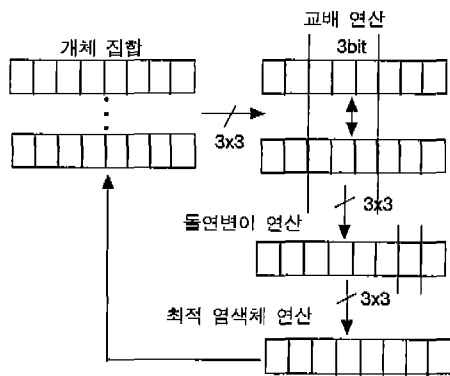


그림 3. 유전자 알고리즘의 구성도
Fig. 3. A Configuration of the Genetic Algorithm

[단계 1] 개체부분에서 퍼지 평가블록 영상과 비퍼지화 블록에서 각각 3×3 영상을 입력으로 받는다.

[단계 2] 교배에서는 두 염색체의 일부를 서로 교환한다(3 bit).

[단계 3] 교배된 두 염색체에서 일부를 변이한다(1 bit).

[단계 4] 돌연 변이를 적용한 염색체의 적합도를 계산한다.

[단계 5] 새로운 개체를 형성하여 이것을 집합부분에 삽입하고, 최적의 값이 나올 때까지 단계 2에서 단계 4까지 반복을 한다.

그림 3은 위의 설명을 그림으로 간단히 나타낸 것이다.

3. FGNN 알고리즘 및 하드웨어 설계

3.1 FGNN 알고리즘

FGNN에서는 교사학습을 이용하여 영상 분류에 적용되는 DBNN(Decision-Based Neural Network) 학습규칙에 하이브리드 퍼지 함수를 적용하면 식 (9)와 같이 나타난다[5-7].

$$\begin{aligned} W_{s_p, q}^{(m+1)} &= W_{s_p, q}^{(m)} + \eta_m v_i \pi_i \\ W_{s_p, q}^{(m+1)} &= W_{s_p, q}^{(m)} - \eta_m v_j \pi_j \end{aligned} \quad (9)$$

여기서 W 는 가중치이고, s_p 는 긍정학습에 대한 클래스의 국부 승리자이고, s_j 는 부정학습에 대한 클래스의 국부 승리자이다. 또한 q 는 국부 승리자의 서브 클래스에 대한 가중치의 집합이며, $\pi_i = \nabla \Phi(x, W)$ 로써 식별함수에 대한 gradient 벡터이고, 시그모이드 함수 η_m 은 학습률을 나타낸다. 또한 n_i 는 하이브리드 퍼지 소속 함수에 사용된 파라미터 $[a, b, d, \alpha, \beta]_{LR}$ 이다. 퍼지 평가부를 생산하기 위해 사용된 퍼지 소속함수는 식 (9)와 같으며, 하이브리드의 퍼지 소속함수는 식 (10)과 같다.

$$\begin{aligned} v_{M_m} &= \{b_x \leq a'_A \text{ 혹은 } b_x \geq d'_A \rightarrow 0, \\ & a'_A + \alpha'_A \leq b_x \leq d'_A - \beta'_A \rightarrow 1, \\ & b_x < a'_A + \alpha'_A \rightarrow (b_x - a'_A) / \alpha'_A, \\ & b_x > d'_A - \beta'_A \rightarrow (d'_A - b_x) / \beta'_A \} \end{aligned} \quad (10)$$

I_A 퍼지 구간이라면, $y_E(\cdot)$ 에 의해서 나타나는 퍼지 평가자 E 는 식 (11)에 의해서 산출된다[11].

$$y_E(X(i, j)) = \frac{X(i, j) * x(i, j)}{v_{M_m}}$$

$$\begin{aligned}
 & \sum_{k=\frac{-(n_1-1)}{2}}^{\frac{n_1-1}{2}} \sum_{l=\frac{-(n_2-1)}{2}}^{\frac{n_2-1}{2}} V_{M_n}(x(i+k,j+l) \times x(i+k,j+l)) \\
 = & \frac{\sum_{k=\frac{-(n_1-1)}{2}}^{\frac{n_1-1}{2}} \sum_{l=\frac{-(n_2-1)}{2}}^{\frac{n_2-1}{2}} V_{M_n}(x(i+k,j+l))}{\sum_{k=\frac{-(n_1-1)}{2}}^{\frac{n_1-1}{2}} \sum_{l=\frac{-(n_2-1)}{2}}^{\frac{n_2-1}{2}} V_{M_n}(x(i+k,j+l))} \quad (11)
 \end{aligned}$$

여기 $\underline{x}(i, j)$ 는 (i, j) 에서 n_1, n_2 샘플 행렬 중심이고, $(i+k, j+l)$ 은 퍼지 구간 하이브리드의 입력 $(i+k, j+l)$ 에 대한 소속 함수의 경사도이다. 여기서 $h(i, j)$ 는 $n_1 \times n_2$ 샘플행렬에 의한 임펄스 응답형식인 컨벌루션(Convolution) $h(i, j) * x(i, j)$ 을 나타내며, Sugeno-type 퍼지규칙의 퍼지평균은 식 (12)와 같다.

IF

$x(i-n_1, j-n_2)$ 은 $f, x(i-n_1, j-n_2+1)$ 은 $f,$
 $\dots, x(i-n_1, j+n_2)$ 은 $f,$
 $x(i-n_1+1, j-n_2)$ 은 $f, x(i-n_1+1, j-n_2+1)$ 은 $f,$
 $\dots, x(i-n_1+1, j+n_2)$ 은 $f, \dots,$
 $x(i+n_1, j-n_2)$ 은 $f, x(i+n_1, j-n_2+1)$ 은 $f,$
 $\dots, x(i+n_1, j+n_2)$ 은 f

THEN

$$\begin{aligned}
 \bar{y}_f(i, j) &= \frac{h(i, j) * x(i, j)}{N_i} \\
 &= \frac{\sum_{k=-n_1}^{n_1} \sum_{l=-n_2}^{n_2} \mu_f(x(i+k, j+l) \times x(i+k, j+l))}{\sum_{k=-n_1}^{n_1} \sum_{l=-n_2}^{n_2} \mu_f(x(i+k, j+l))} \quad (12)
 \end{aligned}$$

FGNN 구조는 퍼지화된 픽셀 값이 신경망 과정의 입력으로 교사학습을 이용하여 어느 일정한 값 이내로 지정된 후, 출력으로 나온 값이 비퍼지화 과정을 통하여 퍼지화된 데이터에서 원래 값을 찾기 위하여 5×5 행렬의 25개의 픽셀 값을 퍼지화한다. 퍼지평균은 식 (12)이고, 퍼지평균에서 하이브리드 퍼지집합인 비퍼지화에 대한 규칙의 추론 결과를 계산한다. 퍼지평균에 의해서 얻어진 값은 식 (11)을 이용하여 오차가 가장 적은 출력 값을 얻기 위해서 평가자 값을 구한다. 퍼지함수의 양자화를 거쳐 퍼지화된 데이터는 신경망을 거친 후 퍼지 평균 연산을 통하여 비퍼지화되고, 유전자 결정 블록으로 평가자 결과 값과 같이 전송된다. 유전자 블록에서 비퍼지화된 데이터들과 평가자 값은 서로의 값을 비교연산 한 후, 그 결과는 가장 작은 오차값을 선택하고, 평가자와 비퍼지값을

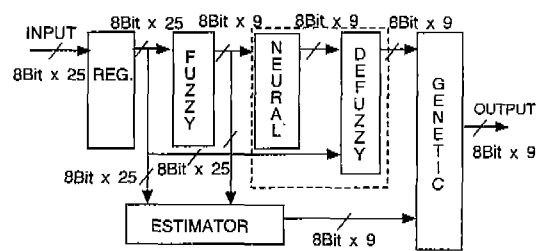


그림 4. FGNN의 연산과정
 Fig. 4. The operation process of FGNN

교배 및 돌연변이 연산자를 이용하여 최적의 값을 선택하는 방식을 채택했다. 그 결과, 유전자 블록에서는 비퍼지값을 오차 이내로 접근하도록 연산과정을 수행한다.

3.2 FGNN의 하드웨어 설계

FGNN의 전체적인 구조는 퍼지 부분에서 연산과정을 수행한 퍼지화된 픽셀 값은 신경망에 3×3 으로 입력이 된다. 그리고 신경구조에 입력된 퍼지 값은 신경망의 AND와 OR 연산자에 의해 오차가 가장 작은 값을 얻는다. 신경망의 출력 값은 비퍼지화 과정에 입력되어 퍼지 평균 연산 식 (12)를 통하여 비퍼지화 과정을 수행하였고, 퍼지 평가블록의 출력값과 같이 유전자 블록에 입력으로 된다. 유전자 블록에서는 3×3 으로 입력된 비퍼지화 값은 최적의 값을 얻기 위하여 교배 및 변이를 취한 후, 일정한 오차 이내로 접근하도록 연산과정을 수행한다. 그림 4는 본 논문에서 FGNN의 전체 연산과정이며, 입력 데이터의 크기는 5×5 의 8bit이고, 출력 데이터의 크기는 3×3 의 8 bit이다.

FGNN 시스템은 퍼지블록, 신경망 블록, 평가자 블록, 비퍼지 블록, 유전자 블록인 5개의 블록으로 구성되었다. 퍼지 블록에 입력되는 데이터는 5×5 로써, 25개의 입력을 가지고 8비트(Bit)로 되었고, 최종 출력은 3×3 로써, 8비트 9개의 출력을 얻는데, FGNN 시스템은 VHDL로 설계하고 Synopsys로 검증을 하였다.

1) 퍼지 블록

퍼지 블록은 5×5 입력을 받아서 이 값 0~255를 양자화하기 위하여 0~1 사이의 값으로 변환하는 부분으로, 큰 데이터를 이용하여 모든 연산을 수행하면 속도와 시간을 너무 많이 소요되기 때문에 이 값들을 0~1 사이의 작은 값으로 변환한다. 여기서는 하이브리드 퍼지 함수를 이용하여 데이터 값을 연산하였다. 각 모듈은 샘플 창에서 $n_1 \times n_2$ 잡음 화소의 기본적인 그레이 레벨이 퍼지 집합의 $n_1 \times n_2$ 매칭정도를 산출하

기 위해 $n_1 \times n_2$ 합축 셀을 적용하였다. 일반적인 매칭 정도이면 각 퍼지 평균 모듈에서 하이브리드 퍼지 집합인 비퍼지에 대한 비퍼지화 셀은 규칙의 추론결과를 계산하도록 하였다. 그림 5는 퍼지 블록을 구성하는 퍼지 셀 부분이고, 여기서 in_i 은 입력값으로 8비트를 가지고 있고, fo_1 은 출력값으로 8비트를 가지고 있다. 그리고 퍼지 전체는 25개 8비트 픽셀값으로써, 전체 25개 입력 값이 9개의 창으로 나누어지고, 나누어진 각 창을 입력으로 한 후 퍼지 블록을 거친 픽셀 값에서 최대값을 구하는 것이고, 한 개의 퍼지 블록에서 한 개의 fo_{1i} 출력이 나온다.

2) 평가자 블록

퍼지 블록의 출력값을 기본으로 퍼지 평균($\sum \sum$ 입력 \times 퍼지값 / $\sum \sum$ 퍼지값)을 얻기 위한 블록으로 5 \times 5 행렬에 대한 평가자 값을 얻고, 이 값들은 비퍼지 블록의 값과 같이 유전자 블록인 결정 블록의 입력 값으로 들어간다. 평가자 값은 유전자 블록에서 비퍼지 값을 평가하는 값으로 사용하게 되며, 또한 평가자 블록은 입력 25개 중 각각 9개를 선택한 in_n 과 퍼지 출력 값 fo_n 9개를 이용하여 9개(8비트) y_{Eon} 인 평가자의 결과 값을 구하였다. 그리고 평가자 과정은 병렬로 구현하여 실시간 처리에 접근하도록 설계하였으며, 평가자 블록은 그림 6과 같다. 여기서 $fo_1 \sim fo_9$ 은 퍼지 결

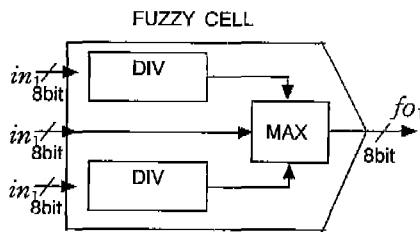


그림 5. 퍼지 셀
Fig. 5. The Fuzzy Cell

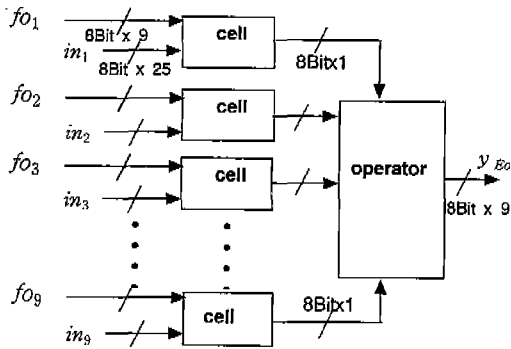


그림 6. 평가자 블록
Fig. 6. The Estimator Block

과 값이고, $in_1 \sim in_9$ 은 입력값, y_{Eon} 은 출력값을 나타낸다. 그리고 평가자 블록은 9개의 셀과 연산과정인 1개의 블록을 포함한 10개의 블록으로 구성되었다.

3) 비퍼지 블록

비퍼지 블록은 평가자 블록 방법과 비슷하게 퍼지 평균을 이용하여, 8비트 3 \times 3 행렬에 대한 결과 값을 얻고, 이 값들은 평가자 값과 함께 유전자 블록의 입력이 된다. 비퍼지의 입력은 신경망을 거친 3 \times 3인 데이터 값이고, 3 \times 3행렬 9개 중 서로 다른 3 \times 3행렬에서 각 MAX 값을 채택한 후 출력된 값을 얻는다. 이것은 총 9개의 서브 블록을 가지고 있으며, 서브 블록은 입력된 값들을 연산한 후 MAX로 선택된 값을 출력한다. 평가자 블록은 25개의 입력과 9개의 퍼지 출력 값을 가지고 있는 것에 비해 비퍼지 블록은 3 \times 3행렬 9개의 입력과 신경망의 3 \times 3행렬 9개의 출력을 입력으로 받는다. 평가자 값으로 비퍼지 값을 최적의 값으로 만드는 블록으로 출력을 한다. 그림 7은 비퍼지 블록이고, 여기서 $NO_1 \sim NO_9$ 은 신경망의 출력값이고, $in_1 \sim in_9$ 은 픽셀 값이다.

4) 유전자 알고리즘 블록

유전자 알고리즘은 초기 집합을 가지고 개선이라는 교배 및 돌연 변이 과정을 수행한 값은 적합성 결정 블록에서 적합도를 계산한 후, 적합성 결정 조건에 만족하면 출력값을 출력하고 조건에 만족되지 않으면 다시 개선 블록으로 환원하여 교배 및 돌연 변이를 과정을 거친 후, 적합성 결정 블록에서 적합도를 계산한 후 적합성 결정 조건에 만족되면 최적 값을 출력하도록 되었다. 비퍼지화된 3 \times 3행렬 데이터들과 평가자 블록의 3 \times 3행렬의 9개 데이터를 유전자 블록 입력으로 받아 연산을 하게 된다. 비퍼지의 데이터와 평가자 값을 개체로 만든 후, 이 개체 9개 중 일부분인 3개 값을 서로 교환하고 교환된 값을 가지고 여기의 6번째에 위치하는 값을 평가자의 MAX 값으로 바꾼 다음 이 값들 중에서 가장 큰 값을 취하게 된다. 그

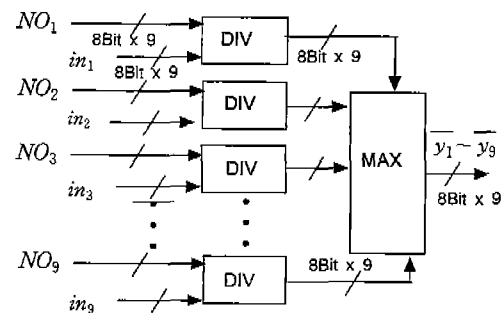


그림 7. 비퍼지 블록
Fig. 7. The Defuzzy Block

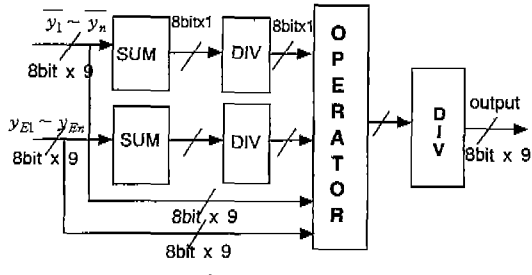


그림 8. 유전자 블록
Fig. 8. The Genetic Block

리고 평가자의 평균값과 여기의 출력값의 각각의 데이터 원소와 비교하여 데이터 값을 30이내로 조정을 하였다. 이 때 30이내의 오차를 갖지 못하면 최적의 값이 나올 때까지 계속해서 반복하도록 설계하였다. 위 설명 중 3개 값을 교환하는 부분을 교배라 하고, 6번째에 위치하는 평가자 값의 MAX값과 교환하는 것을 돌연변이라 한다. 여기에서 출력되는 값은 3x3 행렬 9개를 출력하게 된다. 이는 Top-Down 방식으로 설계하였으며, 그림 8은 유전자 블록을 나타낸 것이다. 여기서 y_n 은 비퍼지 결과 값이고, y_{E_n} 은 평가자 결과 값을 나타낸다.

4. FGNN 하드웨어의 구현 및 성능 분석

하이브리드 퍼지 함수를 적용한 FGNN 회로는 임의의 입력영상과 평가자에 일정한 영상이 저장된 기준 영상을 정규화하고, 이들 영상을 원하는 값 이내로 만든 다음 두 영상사이에 있는 오차를 일련의 조건으로 제한하는 유전자 알고리즘을 적용하여 영상의 값을 비교하여 최적의 값을 구하도록 설계하였다. 영상 입력은 $in_1 \sim in_{25}$ 25개(8비트)가 레지스터에 입력되고, 클럭의 동기여에 의하여 다운에지에서 25개의 데이터가 퍼지 블록과 평가자 블록과 비퍼지 블록의 입력으로 전송된다. 퍼지화 블록은 8비트 상수의 데이터 값을 0~1사이의 값으로 매핑을 한다. 그리고 그 값은 신경망 블록의 입력으로 들어가 일정한 오차이내의 값으로 변환된 후, 비퍼지 블록에서 이값은 퍼지 평균에 의해서 상수의 값으로 복원된다. 그리고 25개 입력을 받은 평가자 블록은 Sugeno-Type의 평균 연산에 의해 값이 복원되어서 9개 값을 출력하는데 이 출력값과 비퍼지 값을 유전자 블록으로 전송하게 된다. 유전자 블록의 입력은 각각 3x3행렬의 9개 값으로 구성되었고, 이 값들은 유전자 블록을 통하여 평가자 값과 가장 오차가 적은 값을 선택하는 것이고, 다시 그 값을 돌연변이를 취하게 되는데 여기서 또 한번 평가자

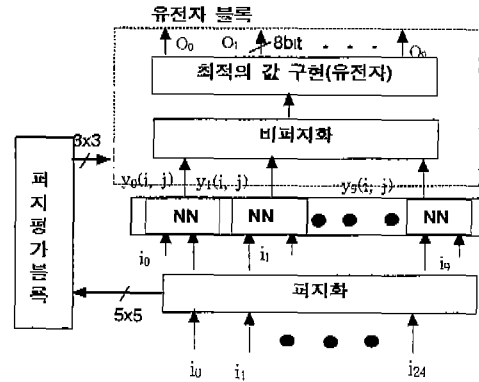


그림 9. FGNN 시스템 구조
Fig. 9. FGNN system structure

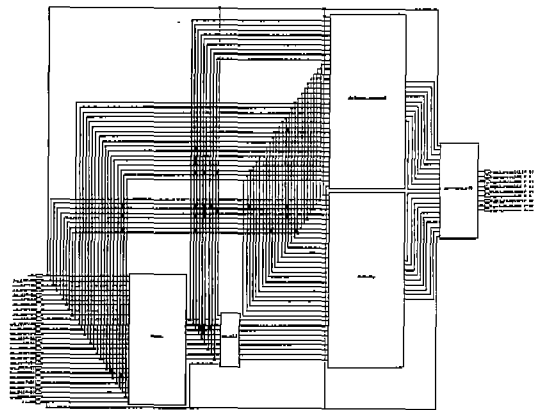


그림 10. FGNN의 회로도
Fig. 10. FGNN Circuit

값과 가장 오차가 적은 값을 취하게 된다. 그래서 그 결과 값이 최적의 값일 경우 최종 출력으로 출력하게 되고, 최적의 값이 아닐 경우 유전자 블록을 다시 수행하여 평가자 값과 가장 근사한 값을 취하도록 한다. 그림 9는 전체 시스템 구조이고, 이 전체 하드웨어는 5개의 퍼지 블록, 신경망 블록, 비퍼지 블록, 평가자 블록, 유전자 블록으로 구성되었다. 또한 그림 10은 FGNN의 내부 전체 회로도이고, 그림 11은 FGNN의 결과 파형으로 25개의 픽셀값이 입력으로 9개의 출력으로 구성되었다.

본 논문은 MATLAB을 이용하여 하이브리드 퍼지 함수와 유전자 알고리즘을 적용한 FGNN에 대한 모의 실험의 영상 결과를 MSE를 통하여 분석을 하였다.

표 1에서 10%, 20%, 30%, 40% 잡음이 포함된 보트 영상에 메디안 필터, OC, FDNN, FGNN을 적용하여 얻은 결과 영상을 MSE를 이용하여 비교한 결과이다. 여기서, 표 1에서 10%, 20%, 30%, 40%의

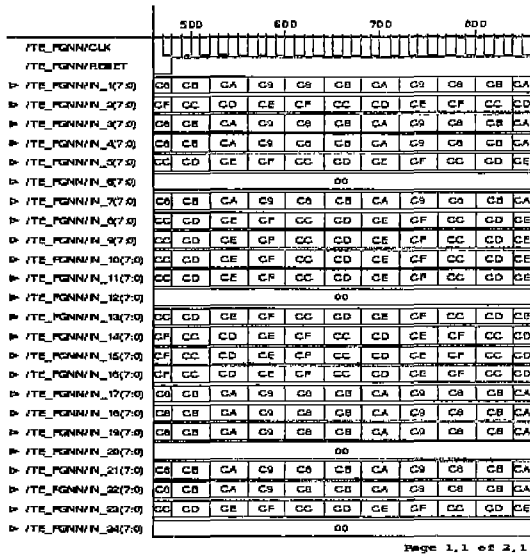


그림 11. FGNN의 결과 파형
Fig. 11. The result wave of FGNN

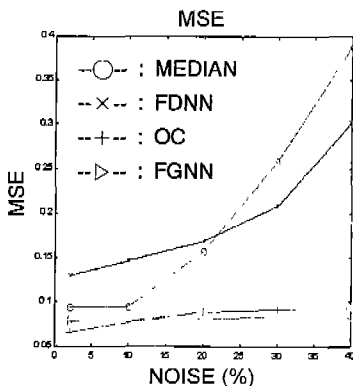


그림 12. 보트 영상의 MSE 결과 그래프
Fig. 12. The MSE result graph for the boats images

잡음이 포함된 영상에 FGNN을 적용한 결과 FDNN을 적용한 영상 결과 보다 0.065, 0.087, 0.125,

표 1. 잡음을 포함한 각 보트 영상에서의 MSE
Table 1. MSE in each boats images included a noise

알고리즘	잡음비율			
	10%잡음	20%잡음	30%잡음	40%잡음
MEDIAN	94.1	157.2	257.9	387.9
OC	77.2	89.5	90.6	93.3
FDNN	146	169	208	301
FGNN	81	82	83	83

0.218정도의 MSE가 적음을 알 수 있었고, 여기서 FGNN을 적용한 영상 결과가 FDNN과 여타 다른 필터 알고리즘보다 우수함을 보였다. 또한 FGNN이 FDNN과 다른 필터들 보다 잡음제거와 복원에 우수함을 본 논문의 실험을 통하여 알 수 있었다. 그리고 그림 12는 보트 영상에 대한 MSE의 결과를 그래프로 나타낸 것이다.

5. 결 론

기존 FDNN은 임의의 데이터가 입력되면 기준 영상 즉 평가자 중에서 유사도가 가장 큰 영상을 찾아 국부 승리자를 선택하고, 국부 승리자 중에서 전체 승리자를 선택하여 최종 출력을 얻는 계층적 FDNN은 결정 블록 그레이 레벨 값을 얻기 위하여 DWW (Dynamic Weight Warping)을 이용하였다. 그런데 이는 원 영상과의 오차가 외부의 파라메타에 의해 제어가 되는데 이 오차는 사람의 경험치에 의해서 일정한 임계값을 설정하는 것으로 최적의 영상을 얻기에는 부적합하였다. 그래서 본 논문에서는 삼각형 퍼지 함수의 단점을 보완한 하이브리드 퍼지 함수와 최적의 값을 얻는 유전자 알고리즘을 적용한 FGNN 알고리즘을 구현하였다. 그리고 MATLAB을 이용한 컴퓨터 모의 실험 결과에서 FGNN이 10%이상의 잡음 영역에서 FDNN 뿐만 아니라 다른 필터들의 성능 보다 우수함을 확인하였다. 특히 미세 정보 영역에서 본 논문에서 FGNN이 다른 필터에 비해 우수함을 보였고, 그리고 보트 영상에서 표 1, 그림 12를 통하여 MSE를 비교하였다. 보트 영상에 대해서 FDNN보다 FGNN을 적용한 결과 보트영상에 대해서 약 0.057~0.087 정도의 오차가 발생하고, OC와는 약 0.007~0.014정도의 오차가 발생한다. 또한 lena 영상에 대해서는 약 0.002~0.011 정도의 오차가 발생하고, OC와는 약 0.008~0.014정도의 오차가 발생하고, 하이브리드 방식의 퍼지 소속함수를 적용한 FGNN의 영상이 더 우수함을 알 수 있었다. FGNN의 보트 영상에 대한 전체적인 MSE의 증가는 약 0.002정도 그리고

lena 영상에 대한 전체적인 MSE의 증가는 약 0.016 정도가 발생한다. FGNN이 FDNN과 다른 필터들 보다 잡음제거와 복원에 우수함을 본 논문의 실험을 통하여 알 수 있었다. 본 논문에서는 삼각형 퍼지 함수의 빈곳(Emptiness)와 혼란(Confusion)등의 단점을 보완한 하이브리드 방식의 퍼지 함수와 최적의 승리자를 찾기 위하여 유전자 알고리즘을 FGNN에 적용함으로써 최적의 영상을 얻을 수 있었다. 그리고 FGNN 시스템의 설계는 VHDL로 코딩(Coding)하였고, 하드웨어 모의 실험은 Synopsys를 이용한 Function Simulation으로 검증을 하였으며, 전체 하드웨어는 Top-Down 방식으로 구성되었으며, 시스템의 각 블록에 대하여 Synopsys 툴을 이용하여 회로의 검증 및 성능을 분석하였고, 이 전체 하드웨어는 파이프라인 형식의 병렬로 구성되었으며, 전체 처리시간은 약 640[ns]이다.

앞으로 하이브리드 방식의 퍼지 함수를 적용한 FGNN을 적용하여 칼라 영상 및 패턴 인식, 에지 검출 등에 적용할 수 있을 것으로 사료된다.

참고문헌

[1] K. Fukushima, "A Neural Networks for Visual Pattern Recognition", IEEE computer Magagine, pp. 65-75, March, 1988.
 [2] J. N. Little, L. Shure, "SIGNAL PROCESSING NEURAL NETWORK", The Math Works Inc, July 1992.
 [3] S. Y. Kung, J. S. Taur, "Decision based neural networks with signal/image classification application", In Proceeding, IEEE Transactions on neural, networks for signal Processing, 1993.
 [4] A. Kandel, G. Langholz, "FUZZY HARDWARE", Kluwer Academic Publishers, 1998.
 [5] C. S. Lee, Y. H. Kuo and P. T. Yu, "Weighted fuzzy mean filters for heavy-tailed noise removal", Proceedings of ISUMANAFIPS'95, IEEE CS.
 [6] C. S. Lee, Y. H. Kuo, "Three dimensional weighted

Fuzzy mean filters for color image filtering", Proceeding of international Joint conference of CFSA/IFIS/SOFT'95 ON Fuzzy Theory and Applications, world scientific Process, pp. 423-428.

[7] C. L. Chen, Y. H. Kuo, "Fuzzy inferenc processor design with generic LR fuzzy cells", proceedings of the international Joint Conference of CFSA/IFIS/SOFT'95 on Fuzzy Theory and Applications.
 [8] K. Yoshida, M. Yamamura and S. Kobayashi, "Generation Pareto Optimal Decision Trees by GAs", Processding of IIZUKA '96, pp. 854-858.
 [9] S. H. Lee, T. W. Kang and C. S. Hwang, "A Genetic Algorithm with a Local Improvement Mechanism for Dynamic Load Balancing in Distributed Systems", Proceedings of IIZUKA '96, pp. 486-489.
 [10] M. Gen, R. Cheng, "GENETIC ALGORITHMS AND ENGINEERING DESIGN", A Wiley-Interscience Publication, 1997.
 [11] C. L. Chen, C. S. Lee and Y. H. Kuo, "Design of high speed weighted fuzzy mean filters with generic LR fuzzy cells", IEEE, 1996. Press, pp. 601-606.



변 오 성 (Oh-Sung Byun)

1997년 : 원광대학교 전자공학과(공학사)
 2000년 : 원광대학교 전자공학과(공학석사)
 2000년~현재 : 원광대학교 전자공학과 (박사과정 재학)
 관심분야 : 신경망 시스템, 퍼지, 유전자 알고리즘, 지능시스템



문 성 룡 (Sung-Ryong Moon)

1982년 : 원광대학교 전자공학과(공학사)
 1986년 : 전북대학교 전자공학과(공학석사)
 1993년 : 전북대학교 전자공학과(공학박사)
 1994년~현재 : 원광대학교 전기전자 및 정보공학부 부교수
 관심분야 : 신경망, 퍼지, 비선형 디지털 시스템 및 지능시스템