

Octree 모델에 근거한 고속 3차원 물체 인식

(Octree Model Based Fast Three-Dimensional Object Recognition)

李永在*, 朴榮泰*

(Young Jae Lee and Young Tae Park)

요 약

2차원 영상으로부터 3차원 물체의 형상을 추론하고 인식하는 연구는 컴퓨터 비전의 중요한 연구분야이다. Octree 모델은 3차원 물체를 계층적으로 묘사할 수 있는 기법으로서 임의의 시각방향에서의 투영영상을 생성할 수 있으므로 3차원 물체를 인식할 수 있는 효율적인 데이터베이스로 사용될 수 있다. 본 논문에서는 시각방향을 추정할 때 요구되는 4쌍의 특징점을 검색하는 빠른 알고리즘을 제안한다. 제안한 기법에서는 49 개의 표준 시각방향으로부터 투영된 2차원 물체형상과 입력형상을 정합하여 초기 시각방향을 추정하고 이들 형상으로부터 4쌍의 특징점 빠르게 검색함으로써 시각방향을 정정하며 입력형상과 계산된 시각방향에서 투영된 형상과의 정합에 의해 검증함으로써 3차원 물체를 고속으로 인식한다. 3차원 모델 물체와 2차원 물체의 특징점의 수를 각각 m 과 n 이라 했을 때, 제안한 기법의 복잡도는 최악의 경우 $O(n^2)$ 이고 단순 조합에 의한 검색기법의 경우는 $O(m^4 \cdot n^4)$ 이다.

Abstract

Inferring and recognizing 3D objects from a 2D occluded image has been an important research area of computer vision. The octree model, a hierarchical volume description of 3D objects, may be utilized to generate projected images from arbitrary viewing directions, thereby providing an efficient means of the data base for 3D object recognition. We present a fast algorithm of finding the 4 pairs of feature points to estimate the viewing direction. The method is based on matching the object contour to the reference occluded shapes of 49 viewing directions. The initially best matched viewing direction is calibrated by searching for the 4 pairs of feature points between the input image and the image projected along the estimated viewing direction. Then the input shape is recognized by matching to the projected shape. The computational complexity of the proposed method is shown to be $O(n^2)$ in the worst case, and that of the simple combinatorial method is $O(m^4 \cdot n^4)$, where m and n denote the number of feature points of the 3D model object and the 2D object, respectively.

I. 서 론

* 正會員, 慶熙大學校 電子工學科

(Department of Electronics Engineering Kyung Hee University)

※ 이 연구는 1995년도 한국과학재단 연구비지원에 의한 결과임.

接受日字: 1996年10月24日, 수정완료일: 1997年8月29日

2차원 영상으로부터 3차원 물체의 형상을 재현하고 인식하는 연구는 컴퓨터 비전의 중요한 연구분야로서 다양한 연구가 진행되어 왔다^[1-10]. 시각정보에 의한 인간의 판단능력과 유사한 기능을 가진 컴퓨터 비전 시스템의 응용 분야로서 2차원 영상을 통해 3차원 물

체를 자동으로 인식할 수 있는 기법은 공장자동화를 위한 로봇 비전, 자동순항 시스템, 군사적 분야에서 Missile의 비전시스템 등 거의 모든 시각정보처리 분야에서 필수적인 핵심기술이다. 기존의 대표적인 3차원 물체인식 기법으로는 영상의 그림자^[7], Texture^[6], 움직임 정보 등을 이용한 Shape from X 기법과 2대의 카메라에서 획득한 Stereo 영상에서의 3차원 깊이 정보 추출에 의한 Stereo vision^[5], Range 정보를 사용한 인식기법^[9] 등의 다양한 기법이 시도되었으나 정확한 3차원 묘사가 어렵고 정합을 위한 처리 과정이 복잡하기 때문에 3차원 컴퓨터 시각장치의 본격적인 활용이 매우 제한적인 상황이다.

동일한 물체라도 3차원 좌표계에서 보는 방향에 따라 매우 다른 2차원 형상을 갖고 있기 때문에 단순히 2차원 형상인식 기법을 적용시키고자 할 경우 모든 방향에 대한 기준 모델의 데이터베이스를 구축해야 한다. 따라서 인식 및 정합을 위한 데이터베이스의 크기가 기하급수적으로 커지며 인식속도가 매우 느려진다. 이러한 문제점을 해결하기 위해 소수의 2차원 영상으로부터 3차원 물체를 묘사하는 방식(Object Centered Model)들이 제안되었다. 그 중 Octree^[1-3] 방식은 효율적인 데이터 구조와 세부 윤곽 정보를 묘사할 수 있는 방식으로 널리 알려져 있다. 다수의 2차원 영상으로부터 Octree를 자동적으로 생성시킬 수 있는 방식도 제안되었다^[3]. 관측자 중심의 영상인 2차원 물체로부터 3차원 물체를 추론하여 표준 3차원 물체의 Octree와 비교하여 3차원 물체를 인식하는 방식이 시도되었으나 2차원 영상의 제한된 정보로 인하여 제한적인 인식만이 가능하였다. 즉 한 개의 2차원 영상으로부터 3차원 물체를 정확히 추론하는 것은 불가능하다. 따라서 반대 방향으로의 정합, 즉 표준 3차원 물체를 2차원으로 적절히 투영하여 주어진 2차원 영상과의 정합 과정을 통하여 인식하는 기법이 필연적이다.

이와 같이 관측자 중심의 영상(2차원)으로부터 물체 중심의 모델(3차원)을 인식하기 위해선 3차원 좌표로부터 2차원 좌표로의 변환이 요구되며 이러한 변환을 위해선 Camera의 초점거리, 물체와의 거리 등의 사전 정보를 필요로 하므로 1개의 2차원 영상정보로부터 3차원 정보를 정확히 산출해 내는 것은 불가능하다. 따라서 영상 좌표 변환의 적절한 가정 하에 변환의 선형화를 통해 문제를 해결하는 방식들이 제안되었다. 예를 들어 Ayache와 Faugeras^[10]는 인식대상 물체가

평평하다는 가정 하에서 물체의 윤곽 정보를 사용하였고 Chien과 Aggarwal^[4]은 평행투영을 가정하였다.

본 논문에서는 평행투영의 가정 하에 3차원 영상과 2차원 영상의 특징점 정합에 의한 Chien과 Aggarwal 방식을 기본으로 한 고속의 3차원 물체 인식 기법을 제시한다. 2차원 입력형상과 모델 3차원 형상의 기하학적 특징점의 대응되는 특징점 쌍을 검색하는 과정에서 3차원 공간을 균일하게 분할한 49개의 표준 시각방향에서의 2차원 형상정보를 사용하여 검색과정을 고속화하고 계산된 시각방향에 대한 2차원 투영영상을 합성하며 입력형상과의 형상정합에 의한 최종적인 검증과정을 통해 인식과정을 완료한다. 평행투영의 가정은 물체의 크기에 비해 카메라의 거리가 상당히 먼 경우에는 오차를 무시할 수 있고 인식 결과와 함께 3차원 물체의 촬영된 각도와 상대위치를 자동으로 계산할 수 있으므로 3차원 시각장치를 위한 기본적인 인식 알고리즘으로 활용할 수 있다. 2장에서는 3개의 직교하는 2차원 영상으로부터 Octree 모델을 자동적으로 생성하는 기법과 임의의 시각방향에서 본 투영영상을 합성하는 기법을 요약한다. 3장에서는 Octree를 사용한 기본적인 3차원 물체인식 기법과 고속 특징점 정합에 의한 인식기법을 제시하며 본 논문에서 제시한 고속 알고리즘의 실험 결과를 4장에 기술한다.

II. Octree 및 2차원 투영영상 생성기법

3차원 물체를 효율적으로 모델링 혹은 묘사하는 기법은 컴퓨터 비전 시스템의 구현을 위한 중요한 연구 분야이다. 3D 묘사기법은 크게 부피 묘사기법과 표면 묘사기법으로 나뉘어 진다. 이 중 Octree는 2차원 물체 묘사 기법인 Quadtree를 3차원으로 확장한 부피 묘사 기법으로서 3차원 물체의 계층적 묘사가 가능하다. 특히 Octree는 3개의 직교하는(orthogonal) 2차원 영상으로부터 Volume intersection 알고리즘^[2]에 의해 자동적으로 생성할 수 있고 3차원 물체의 표면 정보를 추가시킴으로써 Octree 구조로부터 임의의 방향에서 본 2차원 물체 영상을 합성해 낼 수 있으므로 3차원 물체 인식을 위한 효율적인 묘사 기법으로 사용할 수 있다. Octree는 Volume intersection 알고리즘에 의해 생성되며 기본적인 원리는 다음과 같다. 3개의 직교하는 Quadtree의 각 노드를 노드의 공간상 좌표로 연결시키고 3차원 공간에서 교차된 Voxel에 관련된

Quadtree의 3개 노드 모두가 Black 노드일 경우의 Voxel이 Octree의 Object 노드로 결정되며 1 개 이상의 노드가 White인 경우는 Object 노드가 아니다. 모든 3차원 공간상의 Voxel에 대해 이와 같은 과정을 계층적으로 적용함으로써 Octree를 자동으로 생성할 수 있다. 그림 1은 서로 직교하는 3개의 시각방향에서 본 3개의 2차원 영상으로부터 Octree를 자동으로 생성하는 과정의 예를 나타낸다.

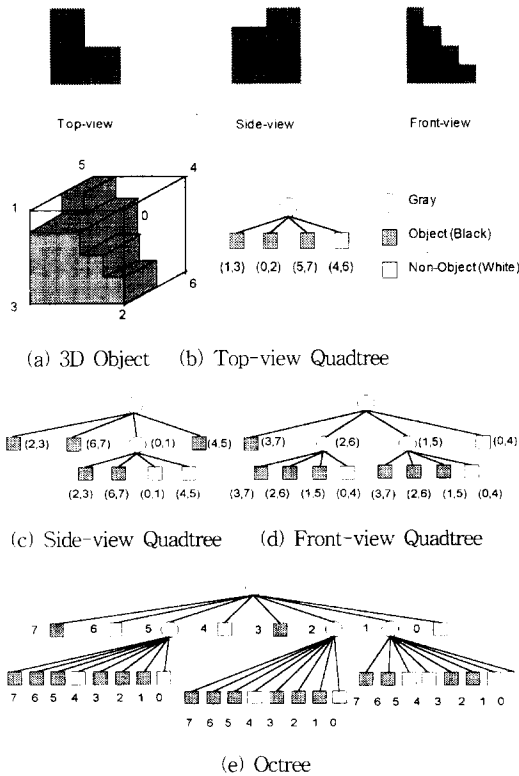


그림 1. Block 모델의 Top, Side, Front view 영상으로부터 생성된 Octree
Fig. 1. Octree generated from Top, Side, Front view block images.

Octree 구조의 유용성은 임의의 시각방향에서 본 3차원 물체의 2차원 투영영상을 Octree로부터 합성할 수 있으므로 인식을 위한 데이터베이스의 크기를 대폭 줄일 수 있는데 있다. Octree의 각 노드에 해당하는 Voxel의 6개 면 중 물체의 표면에 속하는 면을 검출하는 기법으로서 Multi-level boundary search 알고리즘¹²⁾을 사용한다. 검출된 표면 노드 중 Surface normal 벡터와 시각방향 벡터의 곱이 부의 값을 가진 표면 노드가 시각방향에서 보이는 노드이므로 (그림 5

참조) 이들 노드의 표면을 시각방향으로 평행투영하여 모델의 2차원 투영형상을 생성할 수 있으며 이들 벡터의 곱의 절대값을 사용하여 2차원 투영영상을 Pseudo gray로 표현할 수도 있다. 그림 2, 그림 3, 그림 4는 임의의 시각방향에서 본 2차원 투영영상을 Pseudo gray로 합성한 예를 나타내며 생성된 Octree의 Voxel 수는 각각 16729, 22945, 20513 개다.

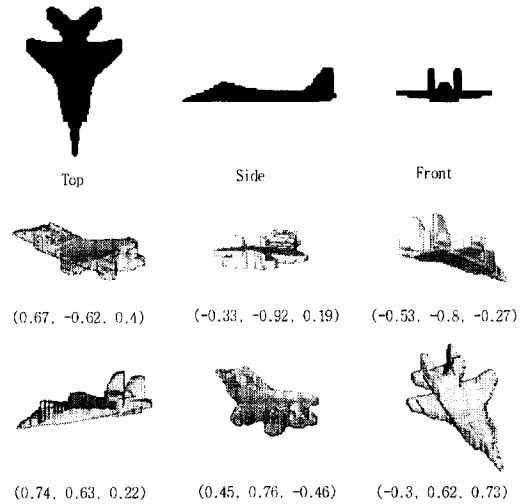


그림 2. Octree로부터 합성된 2차원 투영영상의 예 (Plane 1)
Fig. 2. Example of images projected from octree.

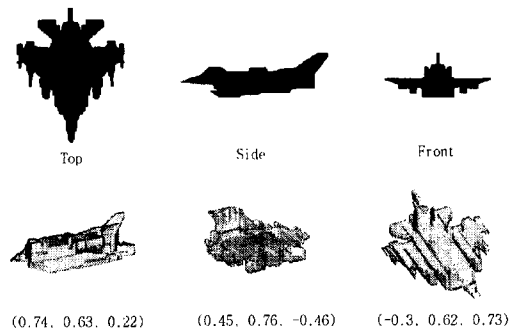


그림 3. Octree로부터 합성된 2차원 투영영상의 예 (Plane 2)
Fig. 3. Example of synthesized 2D projected images.

III. 3차원 물체인식 기법

그림 6에 본 논문에서 제안하는 고속 3차원 물체인식 알고리즘의 개요를 표시한다.

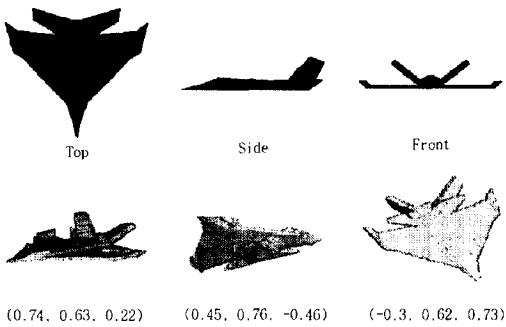


그림 4. Octree 로부터 합성된 2차원 투영영상의 예 (Plane 3)
 Fig. 4. Example of synthesized 2D projected images.

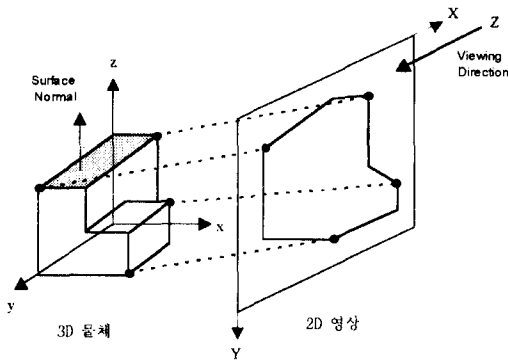


그림 5. 4쌍의 정합된 특징점의 예
 Fig. 5. Example of 4 pairs correspondence.

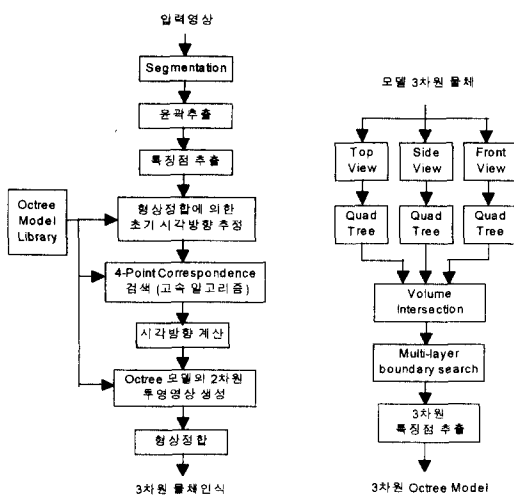


그림 6. 고속 3차원 물체인식 알고리즘과 Octree 모델 생성과정
 Fig. 6. Fast 3D object recognition algorithm and generating octree models.

1. 특징점 추출

2차원으로 투영된 3차원 물체의 인식을 위해 2차원 형상을 효율적으로 묘사할 수 있는 특징점 추출 과정을 사용한다. 특징점은 물체의 윤곽을 근사화하는 윤곽점들의 집합으로서 패턴인식을 위한 전처리 과정으로서 널리 사용되어 왔다^[11, 12]. 특징점은 윤곽의 굴곡률(curvature)이 국부 최대인 점을 검출함으로써 생성된다. 디지털영상의 경우 경사각의 변화가 이산적이므로 굴곡률을 평활화한 k-cosine 방식을 사용한다. m 번째 윤곽점으로부터 n 번째 윤곽점까지의 벡터를 $a_{m,n}$ 으로 표시하면 i 번째 윤곽점의 k-cosine 은 다음과 같이 정의된다.

$$k\text{-cosine} = \frac{a_{i-k,i} \cdot a_{i+k,i}}{|a_{i-k,i}| |a_{i+k,i}|}$$

물체의 윤곽선을 따라 k-cosine 값의 국부 최대점을 검색함으로써 특징점을 추출할 수 있으며 k는 평활화 파라미터로서 클수록 세부 윤곽 묘사가 어렵고 작을수록 지나치게 많은 특징점을 생성하여 비효율적이므로 물체의 크기, 복잡도에 따라 k 값을 적절히 선택해야 하며 본 논문에서는 이를 해결하는 알고리즘으로서 Scale-space filtering 기법^[12]을 사용한다. 특징점은 물체의 불룩한 부분과 오목한 부분에 존재하지만 오목한 부분의 특징점은 시각방향에 따라 보일 확률이 적고 공간상의 위치가 일정치 않으므로 제거하고 불룩한 특징점만을 사용한다.

Top, Side, Front 영상에서 검출된 불룩한 2차원 특징점으로부터 3차원 물체의 특징점은 기본적으로 다음과 같은 원칙에 의해 생성된다. 즉, 2개의 직교하는 시각방향에서 포착된 2차원 영상의 특징점 P_i와 Q_j를 각각의 시각방향으로 연장했을 때 교차한다면 그 교차점은 3차원 물체의 특징점이다. 따라서 2차원 영상에서의 모든 특징점 쌍 P_i, Q_j에 대해 위의 원리를 적용하여 검출된 3차원 특징점과 Octree 각 노드의 Surface normal 정보를 Octree 구조에 추가하여 3차원 물체 인식을 위한 기본 데이터베이스로 사용한다.

2. 4쌍의 특징점 정합조건

2차원 영상의 특징점에 대응되는 3차원 물체의 특징점을 찾기 위해 4쌍의 특징점의 기하학적 변환조건을 사용한다^[8]. 2D 특징점 P_i와 P_j에 대응되는 3D 특징점 p_i의 좌표를 각각 (X_i, Y_i), (x_i, y_i, z_i)라 했을 때 p_j로부터 P_j의 투영은 다음 식에 의해 표현된다.

$$X_i = R_{11} x_j + R_{12} y_j + R_{13} z_j + t_x \quad (1)$$

$$Y_i = R_{21} x_j + R_{22} y_j + R_{23} z_j + t_y$$

즉, 2D 좌표계와 3D 좌표계의 변환 관계는 Rotation matrix의 요소인 $V'_1 = [R_{11} R_{12} R_{13}]$, $V'_2 = [R_{21} R_{22} R_{23}]$ 와 이동변환 요소인 t_x, t_y 에 의해 표현된다. 따라서 4 개의 특징점 쌍이 주어지면 변환 식을 계산할 수 있다. 이 변환 식은 Orthogonal 하므로

$$V_1 \cdot V_2 = 0, \|V_{1\parallel}\| = c, \|V_{2\parallel}\| = c \quad (2)$$

의 조건을 만족하는 4 개의 특징점 쌍의 집합을 검색한다(그림 5 참조). 2D 특징점의 집합 $\{P_1, P_2, \dots, P_n\}$ 과 3D 특징점의 집합 $\{p_1, p_2, \dots, p_m\}$ 이 주어졌을 때 좌표 변환식 (1)과 (2)를 만족하는 4개의 특징점 쌍을 검색함으로써 2D 좌표계와 3D 좌표계의 변환 관계를 구할 수 있다. 특징점의 수가 많을 경우 최적의 특징점 쌍을 구하는 문제는 흔히 LDP (Location Determination Problem)라 불리며 단순 조합에 의한 검색의 경우 $O(m^4 \cdot n^4)$ 에 비례하는 복잡도(Worst case)를 가지므로 많은 검색 시간이 소요된다.

3. 시각방향 계산

시각방향을 관측자 중심 좌표계에서 $[0 \ 0 \ 1]^t$ 로 표시하면 (그림 5 참조) 이에 대응되는 물체 중심 좌표계의 시각방향은 다음과 같이 변환된다. 물체 중심 좌표계에서 관측자 중심 좌표계로의 변환 Matrix R 을

$$R = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \quad (3)$$

로 표기하면 $R^{-1} = R^t$ ($\because R$ is an orthogonal matrix) 이므로

$$\text{시각방향} = R^{-1} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} R_{31} \\ R_{32} \\ R_{33} \end{bmatrix} = \begin{bmatrix} R_{11} \\ R_{12} \\ R_{13} \end{bmatrix} \times \begin{bmatrix} R_{21} \\ R_{22} \\ R_{23} \end{bmatrix} \quad (4)$$

로 계산된다¹⁴⁾. 즉, 식 (1)과 (2)를 만족하는 서로 대응되는 4개의 특징점 쌍이 결정되면 시각방향은 V_1 과 V_2 의 Cross product에 의해 구할 수 있다.

3차원 물체인식은 물체의 형상이 간단할 경우 2차원과 3차원의 특징점 정합만으로도 가능하지만 복잡한 형상의 물체인 경우, 우연의 일치로 식 (1)과 (2)를 만족하는 특징점 쌍이 존재할 수 있다. 따라서 가정된 시각방향에 따라 Octree로부터 합성된 2차원 투영영

상과의 정합을 통하여 확인하는 과정이 필요하다. 즉, 선택된 4쌍의 3D 및 2D 특징점의 정합이 이루어졌을 경우 특징점뿐이 아니고 나머지 윤곽 부분도 제대로 정합이 이루어졌는지 검증하는 최종 단계로서 2차원 형상 정합이 요구된다.

4. 특징점 쌍의 고속 검색 기법

3D 특징점과 2D 특징점의 개수를 각각 m, n 이라 했을 때 식 (1)과 (2)를 만족하는 4개의 정합쌍을 모두 검색하는데 소요되는 시간은 단순 조합에 의한 검색의 경우 $O(m^4 \cdot n^4)$ 에 비례한다. 이를 효율적으로 검색할 수 있는 방법으로서 굴곡률 값을 이용할 수 있다. 3D 특징점 p_i 와 2D 특징점 P_i 를 굴곡률 값이 큰 순서로 재배열하고 첫 번째 3D 특징점이 2D 영상의 특징점으로 대응되는 조건하에서 나머지 3D 특징점의 가시성(visibility)을 판별하고 Visible 한 3D 특징점 중에서 두 번째로 굴곡률 값이 큰 특징점이 2D 영상의 특징점으로 대응되는 조건하에서 위의 과정을 되풀이한다. 이와 같이 여러 개의 3D 특징점으로부터 4개의 특징점과 대응되는 4개의 2D 특징점을 사용하여 식 (2)를 만족시킬 경우 2차원 형상 정합기법에 의해 3차원 물체의 상대위치와 시각방향을 계산한다. 식 (2)를 만족시키지는 디지털 영상에서 양자화 효과 및 잡음에 대한 영향을 고려하여 다음 식에 의해 판정한다.

$$|V_1 \cdot V_2| < \delta_1, 1 - \delta_2 < \frac{\|V_{1\parallel}\|}{\|V_{2\parallel}\|} < 1 + \delta_2 \quad (5)$$

δ_1 과 δ_2 는 작은 양의 상수로서 실험에서는 각각 0.3, 0.2의 값을 사용했다. 4쌍의 특징점이 식 (5)를 만족시키지 못하는 경우는 굴곡률 값이 두 번째로 큰 3D 특징점을 시작으로 위의 과정을 되풀이한다. 굴곡률 값이 큰 순서로 배열을 하는 이유는 굴곡률 값이 클수록 2차원 영상에서 보이는 2D 특징점으로 나타날 확률이 높기 때문이다. 이와 같은 방식은 특징점 쌍의 검색과정의 효율을 평균적으로 높일 수 있지만 최악의 경우 매우 긴 검색 시간이 소요된다. 따라서 본 논문에서는 다양한 각도에서 본 2차원 투영영상을 데이터 베이스로 구축하여 $O(n^2)$ 의 복잡도를 가진 새로운 검색기법을 다음과 같이 제안한다.

동일한 물체라도 3차원 좌표계에서 시각 방향에 따라 매우 다른 2차원 형상을 갖고 있지만 시각 방향이 유사한 경우의 2차원 형상은 서로 유사한 특성이 있다. 따라서 3차원 공간의 시각방향을 균일한 각도로

나누어 각각의 2차원 투영영상을 데이터베이스로 사용하면 특징점 쌍을 효율적으로 검색할 수 있다. 3차원 공간의 세분화 정도에 따라 13개 혹은 49개의 시각 방향을 표준 시각 방향으로 설정할 수 있다. 그림 7은 13개의 시각 방향에 따른 2차원 투영영상과 3차원 특징점 중 시각 방향에서 보이는 특징점을 2차원으로 투영한 예를 나타낸다. 동일한 모델 영상이 존재하는 것은 3차원 물체의 대칭성 때문이다. 보이는 특징점의 수는 시각 방향에 따라 전체 특징점 수 보다 적으므로 검색과정의 효율을 높일 수 있다. 표준 시각 방향에 따른 형상의 윤곽점과 특징점 정보는 3차원 인식을 위한 유용한 정보로 사용될 수 있으며 Octree로부터 자동으로 생성되므로 데이터베이스 구축의 자동화가 가능하다. 이러한 데이터를 기반으로 한 고속 3차원 물체인식 기법은 다음과 같다.

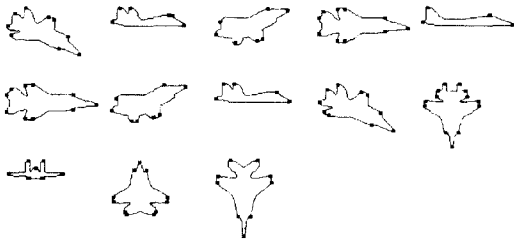


그림 7. 13개의 표준 시각방향에서 투영된 물체형상과 특징점

Fig. 7. Projected images from 13 reference viewing directions and feature points.

고속 3차원 물체인식 기법:

- Step (1): Obtain a pair of feature points $\{p_i, p_j\}$ having the longest distance in the input image;
- Step (2): For all model images {
 - Obtain three pairs of feature points $\{q_k, q_s\}$ having the longest distance;
 - Compute the average boundary point distance D_{avr} of contour points when p_i and p_j are matched to q_k and q_s , respectively;
- Step (3): Let I_m denote the model image associated with the minimum D_{avr} ;
- Step (4): Let $\{Q(p_1), Q(p_2), \dots, Q(p_n)\}$ denote a set of feature points of I_m nearest to $\{p_1, p_2, \dots, p_n\}$;
- Step (5): For (x=0; x<n-1; x++) {
 - For (y=x+1; y<n; y++) {
 - Let $S_2 = \{p_i, p_j, p_x, p_y\}$ denote a set of 4 points in the input image;
 - Let $S_3 = \{q_k, q_s, Q(p_x), Q(p_y)\}$ denote

- a set of 4 points in I_m ;
- Compute V_1 and V_2 using Eq. (1);
- If (V_1 and V_2 satisfy Eq. (2)) {
 - Compute the viewing direction using Eq. (4);
 - Generate the 2D projected image using the octree;
 - Compute D_{avr} between the model and the input images;
 - If ($D_{avr} < \text{Threshold}$) Stop;

입력영상의 특징점 쌍 중 가장 멀리 떨어진 쌍 (p_i, p_j) 과 모델 데이터베이스의 영상 I_1, I_2, \dots 의 각 영상에 대해 특징점 쌍 중 가장 멀리 떨어진 3개의 쌍 (q_k, q_s) 를 정합시켰을 때의 정합률을 계산하고 가장 정합률이 높은 모델영상 I_m 을 Step (2)에서 결정하고 이를 Step (4)와 Step (5)에서 검증한다. 3개의 쌍 $\{q_k, q_s\}$ 을 정합하는 것은 시각방향의 미세한 변화에 따라 가장 멀리 떨어진 특징점 쌍이 달라질 수 있기 때문이다. 정합과정 Step (5)에서 최대 $n(n-1)/2$ 개의 Hypothesis에 대한 검증이 필요하므로 복잡도는 $O(n^2)$ 이다. 정합률의 척도로서는 입력영상과 모델의 윤곽점 사이의 평균거리를 사용한다. 2개의 특징점 쌍이 주어지면 모델 좌표계로부터 입력영상 좌표계로의 변환 요소인 회전각도 θ 와 크기비율 μ 를 계산하고 좌표계의 변환은 다음과 같은 좌표 변환식으로 이루어진다.

$$x' = \mu \cdot \cos(\theta) \cdot x - \mu \cdot \sin(\theta) \cdot y + t_x$$

$$y' = \mu \cdot \sin(\theta) \cdot x + \mu \cdot \cos(\theta) \cdot y + t_y$$

(x', y') 는 입력영상의 좌표계를, (x, y) 는 모델의 좌표계를 나타낸다. 이동벡터 (t_x, t_y) 는 대응되는 특징점 쌍으로부터 평균자승오차가 최소화 되도록 구한다. 좌표 변환식에 의해 모델의 윤곽점을 입력영상의 좌표계로 변환하고 변환된 모델과 입력영상의 윤곽점 집합을 각각 $P = \{p_1, \dots, p_m\}$, $Q = \{q_1, \dots, q_n\}$ 라 했을 때 윤곽점 사이의 평균거리는 $D_{avr} = \frac{1}{m} \sum_i \text{distance}(p_i, q_{i-\mu})$ 로 계산되며 이 값이 작을수록 정합률이 높음을 의미한다. 그림 8은 입력영상에 대해 13개의 표준 시각 방향에서 투영된 모델형상을 각각 정합시킨 결과를 보인다. 첫 번째와 9 번째 모델 영상이 가장 높은 정합률을 나타내고 있으며 입력영상과 가장 유사함을 알 수 있다. 3차원 물체는 시각방향이 정반대일

경우 서로 Mirror 영상인 특성을 가지며 13개 혹은 49개의 표준 시각방향에서는 Mirror 영상 중 1가지만을 포함하기 때문에 Step (2)의 정합과정에서는 Mirror 영상에 대한 정합도 수행해야 한다.



그림 8. 시각방향 (0.52, 0.78, 0.34) 에서 본 입력영상과 정합된 13개의 표준영상
 Fig. 8. Input image from (0.52, 0.78, 0.34) and the matched 13 reference images.

검색과정을 고속으로 처리하기 위해 입력영상의 특징점에 가장 가까운 모델 영상에서의 특징점을 Step (4)에서 검색하고 이미 결정된 2개의 특징점 쌍 (p_i, q_k) 와 (p_j, q_s) 외에 나머지 2개의 특징점 쌍을 Step (5)에서 검색한다. 이때 가정된 2차원과 3차원의 4개의 특징점 쌍 $S_2 = \{p_i, p_j, p_x, p_y\}$ 와 $S_3 = \{q_k, q_s, Q(p_x), Q(p_y)\}$ 가 정합된 쌍임을 검증하기 위해 식 (1)과 식(2)의 조건을 만족시키는지를 판단하고 만족시킬 경우 시각 방향을 식 (4)에 의해 계산한다. 3차원 물체를 계산된 시각 방향에서 본 형상과 입력 형상이 일치하는지를 최종적으로 검증하기 위해 Octree로부터 2차원 투영영상을 생성시켜 D_{avr} 을 계산하고 D_{avr} 이 주어진 임계치보다 작을 경우 3차원 인식과정이 완료된다. 실험에서는 임계치로서 4 pixel 값을 사용했다. 그림 9 에 4쌍의 정합점 검색 기법의 예를 보인다. 그림 9(e)와 (f)는 정합에 사용된 입력영상과 모델영상의 2쌍의 특징점을, 그림 9(g)와 (h)는 정합된 4개의 특징점 쌍을 나타낸다. 그림 9(i)는 모델영상을 입력영상 좌표계로 변환시킨 예를 보이며, 그림 9(j)는 계산된 시각방향에서 투영된 영상을 입력영상 좌표계로 변환하여 정합된 결과를 보인다. 이 경우는 모델의 Mirror 영상이 입력영상에 정합된 예이다.

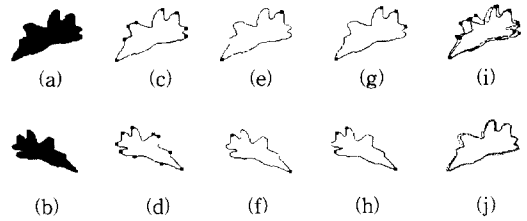


그림 9. 4쌍의 정합점 검색 기법의 예
 Fig. 9. Example of 4 point correspondence.

IV. 실험 결과 및 고찰

제안한 모든 알고리즘은 SUN SPARC 20 워크스테이션에서 C와 X-Window의 Motif 환경에서 구현하였다. 그림 2, 3, 4 에서 주어진 3종류의 항공기 형상에 대해 Octree를 구축하고 그림에서 예로 든 3가지 시각방향에서 투영된 영상에 본 논문에서 제시한 고속 3차원 인식 알고리즘을 적용한 결과를 표 1과 그림 10에 보인다.

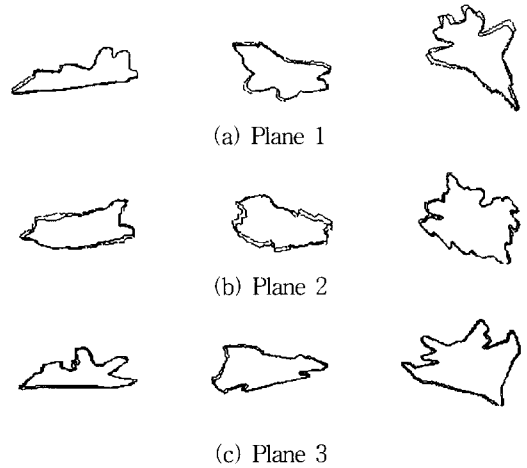


그림 10. 3가지 시각방향에서 본 입력영상에 대한 인식결과
 Fig. 10. Example of the images from 3 viewing directions and recognition results.

Octree 구축에 사용한 3가지 영상의 크기는 모두 129x128이다. 주어진 시각방향 벡터를 a , 계산된 시각방향 벡터를 b 라 했을 때 각도오차는 $\Delta\theta = \cos^{-1}(a \cdot b)$ 로 계산된다. Plane 1의 세 번째와 Plane 3의 첫 번째 시각방향의 경우, 계산된 시각방향의 y 축과 z 축의 부호가 반대로 계산되었는데 이는 모델형상의 대칭성 때문이다. 시각방향에 따라 3차원의 특징점이

2차원으로 투영되는 과정에서 특징점의 위치가 정확히 표현되지 않는 경우 각도오차가 상당히 크게 발생할 수도 있다. 특징점 쌍 검색과정에서 3차원 모델 물체와 입력영상의 특징점의 모든 조합에 대해 검색할 경우 (Full search), 우연의 일치로 4쌍의 특징점이 빨리 검색될 수도 있지만 일반적으로 매우 긴 검색시간이 소요됨을 알 수 있다. 이에 비해 본 논문에서 제시한 인식기법은 2차원 영상의 특징점에 대응되는 3차원 물체의 특징점만을 검색과정에 사용하므로 검색시간이 매우 빠르다.

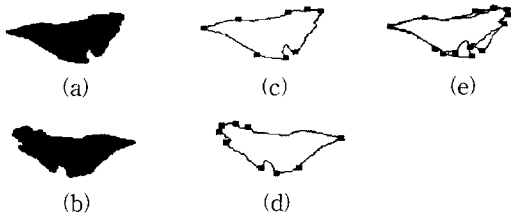


그림 11. 고속 알고리즘에서 거부된 영상의 예
Fig. 11. Example of the image rejected by the fast algorithm.

표 1. 항공기 영상에 대한 3개의 시각방향 영상의 실험결과
Table 1. Experimental results on airplane images from 3 viewing directions.

모델	시각방향	계산된 시각방향	각도 오차	D_m	특징점수		인식시간 (Sec)	
					3차원	2차원	Full Search	고속 알고리즘
Plane 1	(0.74, 0.63, 0.22)	(0.64, 0.72, 0.25)	51°	2.5	18	8	33.9	1.18
	(0.45, 0.76, -0.46)	(0.41, 0.73, -0.46)	26°	1.8	18	8	16.7	1.22
	(-0.3, 0.62, 0.73)	(-0.19, -0.68, -0.71)	73°	1.8	18	15	17.8	1.38
Plane 2	(0.74, 0.63, 0.22)	(0.70, 0.66, 0.22)	43°	2.7	37	8	473.8	1.39
	(0.45, 0.76, -0.46)	(0.53, 0.78, -0.35)	79°	3.4	37	11	2534.0	1.76
	(-0.3, 0.62, 0.73)	(-0.36, 0.64, 0.68)	47°	1.8	37	14	633.5	1.48
Plane 3	(0.74, 0.63, 0.22)	(0.77, -0.6, -0.22)	24°	2.2	30	9	12.7	1.10
	(0.45, 0.76, -0.46)	(0.44, 0.73, -0.43)	27°	1.4	30	10	721.7	1.44
	(-0.3, 0.62, 0.73)	(-0.31, 0.66, 0.68)	37°	2.2	30	12	187.9	1.97

제한한 인식기법의 타당성을 검증하기 위해 난수 발생기에 의한 x, y, z 축의 난수를 시각방향 벡터의 Norm 값이 1이 되도록 정규화 하여 100 개의 임의의 시각방향을 생성하고 각각의 시각방향에서 투영된 영상을 대상으로 3차원 인식 알고리즘을 적용시킨 결과를 표 2에 보인다. 표준 시각방향을 13개로 했을 경우 인식이 거부된 시각방향이 다수 발생하였다. 거부될 경우는 제한한 기법의 모든 4쌍의 특징점들이 식 (5)를 만족하지 않거나 유추된 시각방향의 오차가 심

해서 투영영상과 입력영상의 정합률이 낮을 때 발생한다. 표 2에서 알 수 있듯이 표준 시각방향을 49개로 확장함으로써 거부율을 대폭 줄일 수 있다. 인식이 거부된 경우의 투영영상은 그림 11에 보인 바와 같이 초기 시각방향 추정과정에서 선택된 2쌍의 특징점 외에 나머지 2개의 특징점 쌍을 검색하는 과정에서 2D 영상의 특징점에 가장 가까운 투영영상의 특징점의 위치가 기하학적으로 부정확하게 설정된 경우이다. 이와 같이 형상의 특징부분이 명확히 나타나 있지 않은 특수한 경우를 제외하고 제안한 고속 알고리즘은 Octree에 기반한 3차원 형상 정합과정에서 시각방향에 무관하게 매우 빠르고 안정된 인식 능력을 보인다. 3차원 모델의 특징점 수가 많을수록 Full search의 경우 검색시간이 기하급수적으로 증가하는데 비해 고속 알고리즘의 경우의 검색시간은 거의 일정한데 이는 2차원 영상의 특징점에 대응되는 3차원 물체의 특징점만을 사용하기 때문이다.

표 2. 임의의 100개의 시각방향 영상의 실험결과 (고속 알고리즘)

Table 2. Experimental results on the images from 100 arbitrary viewing directions.

모델	평균 각도오차		평균 D_m		Rejection 개수		평균 인식시간 (Sec)	
	13개	49개	13개	49개	13개	49개	13개	49개
Plane 1	7.5°	6.6°	2.3	2.3	11	1	1.54	1.45
Plane 2	8.1°	6.2°	2.5	2.5	18	0	2.13	2.60
Plane 3	6.9°	7.0°	2.8	2.7	29	4	1.31	1.39

V. 결론

Octree 모델은 3차원 물체를 계층적으로 묘사할 수 있는 기법으로서 임의의 시각방향에서의 투영영상을 생성할 수 있으므로 3차원 물체 인식을 위한 효율적인 데이터베이스로 사용될 수 있다. 본 논문에서는 49 개의 표준 시각방향으로부터 투영된 2차원 물체형상과 입력영상을 정합하여 초기 시각방향을 추정하고 이들 형상의 기하학적 정보로부터 4쌍의 특징점을 고속으로 검색하는 기법을 제시하였다. 검색된 4쌍의 특징점으로부터 시각방향을 계산하고 입력형상과 계산된 시각방향에서 투영된 형상과의 정합에 의한 검증과정을 통하여 3차원 물체를 고속으로 인식할 수 있음을 실험을 통하여 확인하였다. 3차원 물체가 겹쳐져 있는 경우는 표준 시각방향 추정과정과 검증과정에서 부분형상 정

합기법을 도입함으로써 제안된 기법을 보완할 수 있다. 3차원 물체인식 기법의 실용화가 어려운 이유는 자연적인 영상에서 주변환경으로부터 물체 형상을 추출하는 Segmentation 과정이 불완전하기 때문이다. Segmentation이 어려운 경우는 영상의 Line 세그먼트 혹은 특징점 정합에 의한 시각방향 추정과 검증과정의 적용이 가능할 것으로 보인다. 즉, 시각방향 추정과정에서는 본 논문에서 제시한 4개의 특징점 쌍의 정합조건을 사용할 수 있고 추정된 시각방향에서의 투영영상을 Octree로부터 합성하여 입력영상의 Line 세그먼트와의 정합률을 계산하여 추정된 시각방향을 검증할 수 있다. 따라서 Octree에 의한 3차원 물체 모델링 기법은 임의의 시각방향에서의 투영영상을 생성할 수 있고 데이터베이스 구축과정의 자동화가 가능하므로 3차원 시각장치의 기본적인 인식 기법으로서 활용될 수 있다. 위에서 기술한 자연적인 영상에서의 3차원 물체인식 기법에 관한 연구는 본 논문에서 제안한 Octree 모델을 활용하여 수행할 예정이다.

참 고 문 헌

- [1] C.L. Jackins and S.L. Tanimoto, "Oct-trees and their use in representing three-dimensional objects," CGIP vol. 14, pp. 249-270, (1980)
- [2] C.H. Chien and J.K. Aggarwal, "Volume/Surface octrees for the representation of 3-D objects," CGIP, vol. 36, pp. 100-113, (1986)
- [3] H. Noborio, S. Fukuda and S. Arimoto, "Construction of the octree approximating three-dimensional objects by using multiple views," IEEE PAMI, vol. 10, no. 6, pp. 769-782, (1988)
- [4] C.H. Chien and J.K. Aggarwal, "Model construction and shape recognition from occluding contours," IEEE PAMI, vol. 11, no. 4, pp. 372-389, (1989)
- [5] Pollard et al. "PMF: A stereo correspondence algorithm using a disparity gradient limit," *Perception* 14, pp. 449-470, (1985)
- [6] A. Witkin, "Recovering surface shape and orientation from texture," *Artificial Intelligence* 17, pp. 17-45, (1981)
- [7] B. Horn and M. Brooks, *Shape from shading*, MIT Press, (1989)
- [8] T. M. Silberger, L. Davis and D. Harwood, "An iterative Hough procedure for three dimensional object recognition," *Pattern Recognition*, vol. 17, no. 6, pp. 621-629, (1984)
- [9] R. Krishnapuram and D. Casasent, "Termination of three-dimensional object location and orientation from range images," IEEE PAMI, vol. 11, no. 11, pp. 1158-1166, (1989)
- [10] N. Ayache and O.D. Faugeras, "HYPER: A new approach for the recognition and positioning of two-dimensional objects", IEEE PAMI, vol. 8, no. 1, pp. 44-54, (1986)
- [11] C. H. Tch and R. T. Chin, "On the detection of dominant points on digital curves," IEEE PAMI vol. 11, 859-872, (1989)
- [12] S. Pei and C. Lin, "The detection of dominant points on digital curves by scale-space filtering," *Pattern Recognition*, vol. 25, 1307-1314 (1992)

저 자 소 개



李永在外(正會員)

1962년 4월 8일생. 1984년 2월 충남대 전자 교육공학과 졸업. 1994년 8월 연세대 전자 공학과 졸업(공학 석사). 1986년 ~ 1995년 LG 전자 부품(주) 근무. 1996년 3월 ~ 현재 경희대 전자과 박사과정. 주관심분야는

컴퓨터 비전, 신경 회로망

朴榮泰(正會員) 第33卷 第5號 參照

현재 경희대 전자공학과 부교수

論文97-34C-9-11

다중 물체 검출을 위한 관심 영역 추출자의 신경망 구현 (An Implementation of an Attention Operator Using Neural Network for Multiple Object Detection)

許娟珠*, 朴昶俊*, 崔興文*

(Youn-Joo Heo, Chang-Joon Park, and Heung-Moon Choi)

요 약

본 논문에서는 RBF 망을 이용하여 물체에 대한 사전 정보 없이도 다중 물체를 효과적으로 검출할 수 있는 관심 영역 추출자(attention operator)를 구현하였다. 다중 물체 검출을 위한 관심 영역 추출에 적합하도록 구현하기 위해 RBF 망의 은닉층과 출력층 사이에 곱셈을 수행하는 파이 뉴런을 추가하여 일반화 대칭 변환을 변형 구현하였다. 은닉층 뉴런의 중심 벡터 및 파이 뉴런과 출력층 뉴런 사이의 연결세기를 입력 영상 화소의 명도변화 방향의 수렴성, 발산성 및 일관성을 잘 반영하도록 설정함으로써 기존의 관심 영역 추출자에서와는 달리 물체와 물체 사이의 영역은 검출되지 않고 물체만을 검출할 수 있도록 하였으며 잡음에 대해서도 둔감하도록 하였다. 제한한 알고리즘을 여러 물체가 포함된 영상에 대해 시뮬레이션 하여 물체에 대한 사전 정보 없이도 다중 물체들을 잘 검출할 수 있음을 확인하였다.

Abstract

This paper presents an implementation of an attention operator for context-free multiple object detection using modified radial basis function network. In order to implement the attention operator for multiple object detection, we modified the conventional generalized symmetry transform and inserted Π -neurons between the hidden layer and the output layer of the radial basis function network. Unlike the conventional attention operator, we can detect only the inner points of each object without an erroneous detection of the interobject region by adjusting the mean vector of the hidden layer neurons and the synaptic weight between the Π -neurons and the output layer neurons to reflect the convergence, divergence and the consistency of the orientation of the intensity gradient. The simulation results show that the proposed algorithm can efficiently detect multiple objects without *a priori* knowledge.

I. 서 론

최근 컴퓨터 비전이나 패턴 인식과 같은 영상처리 관련 연구에서 물체 인식은 그 응용 분야가 다양하여 많이 연구되어 왔으며, 특히 여러 물체가 혼재하는 영

상에서 각 물체들을 인식하는 다중 물체 인식에 관한 관심이 고조되고 있다^[1-3]. 이와 같은 다중 물체 인식을 위해서는 대상 물체에 대한 사전 정보 없이도 그 내부점을 찾아서 그 물체의 위치를 구하는 물체 검출에 관한 연구가 선행되어야 하며, 이에 관한 연구들이 활발히 진행되고 있다^[4-6].

물체에 대한 사전 정보가 주어진 경우에 있어서 Goudail 등^[4]은 물체 형태에 따른 창함수(window function)와 Bayes 이론을 이용하여 물체를 검출하였고, Ghosal 등^[5]은 검출 대상 물체의 형태에 따라

* 正會員, 慶北大學校 電子電氣工學部

(The School of Electronics and Electrical Engineering, Kyungpook National University)

接受日字:1997年3月14日, 수정완료일:1997年9月1日

설계한 Zernike 필터를 이용하여 물체를 검출하였다. 한편, Trivedi 등^[6]은 입력 영상에 대해 물체 존재 후보 영역을 설정하고 그의 크기와 외곽선 모양 등을 물체에 대한 사전 정보와 비교하여 물체를 검출하였다. 이들 방법에서는 물체의 크기나 형태 등의 사전 정보를 이용하여 물체를 검출하므로 검출 대상 물체의 종류가 증가함에 따라 반복 처리에 따른 처리 복잡도와 처리 시간이 증대되어 이를 다중 물체 검출에 적용하기가 어렵다. 따라서, 효과적으로 다중 물체를 검출하기 위해서는 물체에 대한 사전 정보 없이도 여러 형태의 물체를 검출할 수 있는 방안이 필요하다.

한편, 검출 대상 물체에 대한 사전 정보가 없는 경우 Reisfeld 등^[7]은 관심 영역 추출자(attention operator)로서 일반화 대칭 변환(generalized symmetry transform)을 제안하고, 대상에 대한 사전 정보 없이도 물체, 사람의 얼굴, 얼굴 내의 눈, 입 등의 관심 영역(region of interest)을 검출하였다. 그러나, 이 일반화 대칭 변환은 그 대칭 변환 과정이 복잡할 뿐만 아니라 잡음에 의한 대칭 기여도(symmetry contribution)가 계속 누적되는 등의 문제점이 있다. 또한, 다중 물체 검출에 적용할 경우 대칭 마스크 크기보다 가까이 있는 두 물체 사이의 영역도 관심 영역으로 검출하며 물체의 크기 변화에 따라 대칭 마스크 크기를 가변 해야 하고 또 반복 처리에 따른 많은 계산 시간이 요구된다.

따라서, 이러한 관심 영역 추출자를 다중 물체 검출에 효과적으로 적용하기 위해서는 복잡한 처리 과정을 단순화시키고, 인접한 물체간 대칭성(interobject symmetry)에 의한 대칭도(symmetry magnitude)와 잡음에 의한 대칭도를 억제하여 물체 사이의 공간을 물체로 오인하는 일이 없도록 하는 방법과 고정된 크기의 단일 대칭 마스크로도 여러 크기의 다중 물체를 검출할 수 있는 방법이 연구되어야 한다.

본 논문에서는 RBF 망(radial basis function network)을 이용하여 물체들에 대한 사전 정보 없이도 다중 물체를 효과적으로 검출할 수 있는 관심 영역 추출자를 구현하였다. 제안한 방법에서는 신경망 중 대칭성을 잘 반영하는 RBF 망의 은닉층과 출력층 사이에 곱셈을 수행하는 파이 뉴런(Π -neuron)을 추가하여 일반화 대칭 변환을 변형 구현하였다. 이렇게 변형 구현한 RBF 망(modified radial basis function network: MRBFN)의 은닉층 뉴런의 중심 벡터 및

파이 뉴런과 출력층 뉴런 사이의 연결세기를 입력 영상 화소의 명도변화 방향(orientation of intensity gradient)의 수렴성, 발산성 및 일관성을 잘 반영하도록 설정함으로써 인접한 물체간 대칭성에 의한 대칭도를 억제하여 물체 사이의 좁은 영역은 물체로 검출하지 않고 순수 물체들만을 검출할 수 있도록 하였고 잡음에 의한 대칭도도 누적되지 않도록 하였다. 관심 영역 추출자를 신경망으로 구현함으로써 그 처리 과정을 단순화시키고 하드웨어 구현도 용이하도록 하였다. 또한, 입력 영상을 피라미드 구조로 변환한 후 제안한 MRBFN을 이용함으로써 단일 크기의 신경망으로도 여러 크기의 다중 물체를 효과적으로 검출할 수 있도록 하였다. 제안한 구현 방법의 타당성을 확인하기 위해 여러 물체가 혼재하는 영상에 대해 시뮬레이션하고 그 결과를 검토 고찰하였다.

II. RBF 망을 이용한 관심 영역 추출자의 구현과 다중 물체 검출

본 논문에서는 RBF 망을 이용하여 일반화 대칭 변환을 다중 물체 검출에 적합하도록 변형 구현하고, 이를 이용하여 다중 물체를 효과적으로 검출하였다. 제안한 물체 검출 알고리즘에서는 그림 1과 같이 입력된 영상을 피라미드 구조로 변환한 후 피라미드 각 층에 대해 경사 연산자(gradient operator)를 사용하여 대칭 변환에 필요한 각 화소의 명도변화의 크기(magnitude of intensity gradient)와 방향을 구하였다. 각 화소의 명도변화 방향의 수렴성, 발산성 및 일관성을 잘 반영하도록 RBF 망을 변형하고, 이를 이용하여 각 피라미드 층의 대칭도 맵(symmetry magnitude map)을 구한 후, 이들을 누적한 3차원 대칭도 맵(3-dimensional symmetry magnitude map)에서 첨두치를 구하여 물체를 검출하였다.

1. 일반화 대칭 변환^[7]

Reisfeld 등이 제안한 일반화 대칭 변환은 각 화소의 명도변화의 크기와 방향을 이용하여 일정 영역 안에서 화소들의 대칭 기여도를 누적하여 영상 내에서 대칭성이 강한 부분을 부각시킨다. 먼저, 명도변화 방향으로부터 위상 가중 함수(phase weight function) $P(i, j)$ 는

$$P(i, j) = [1 - \cos(\theta_i + \theta_j - 2\alpha_{ij})] \times [1 - \cos(\theta_i - \theta_j)] \quad (1)$$

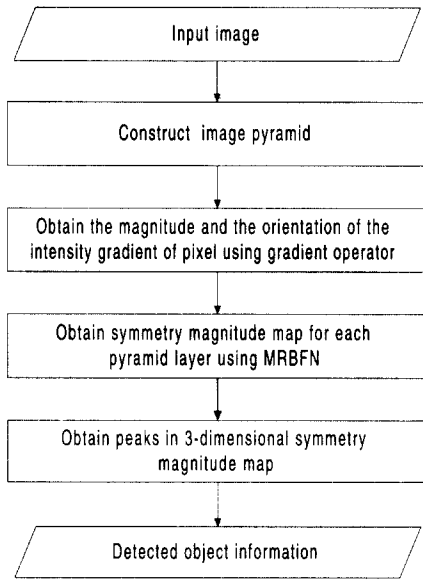


그림 1. 제안한 물체 검출 알고리즘의 흐름도
 Fig. 1. Flow chart for the proposed object detection algorithm.

와 같이 정의된다.

식에서 θ_i 와 θ_j 는 위치 (x_i, y_i) 와 (x_j, y_j) 에서의 화소 p_i 와 p_j 의 명도변화 방향을, α_{ij} 는 화소 p_i 와 p_j 를 연결하는 직선이 수평선과 이루는 각도를 의미한다. 그리고 거리 가중 함수(distance weight function) $D_\sigma(i, j)$ 는

$$D_\sigma(i, j) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\|p_i - p_j\|}{2\sigma^2}\right) \quad (2)$$

와 같이 정의된다. 식에서 σ^2 은 대칭 마스크의 크기를 나타낸다. 그리고 두 화소의 대칭 정도를 의미하는 대칭 기여도 $C(i, j)$ 는

$$C(i, j) = D_\sigma(i, j)F(i, j)r_i r_j \quad (3)$$

와 같이 구해진다. 식에서 r_i 와 r_j 는 화소 p_i 및 p_j 의 명도변화 크기를 각각 의미한다. 따라서, 대칭 기여도는 두 화소의 명도변화 방향이 대칭일수록, 두 화소의 거리가 가까울수록, 두 화소의 명도변화 크기가 클수록 큰 값을 가지게 된다. 그리고 대칭 화소 집합을

$$\Gamma(p) = \left\{ (i, j) \mid \frac{p_i + p_j}{2} = p \right\} \quad (4)$$

이라 할 때, 대칭 화소 집합에서 정의되는 중심 위치의 화소 p 의 대칭 정도를 의미하는 대칭도 $M_\sigma(p)$ 는

$$M_\sigma(p) = \sum_{(i, j) \in \Gamma(p)} C(i, j) \quad (5)$$

와 같이 구할 수 있다. 일반적으로 물체들은 대칭성이 강하므로 일반화 대칭 변환을 물체 검출에 적용할 경우 물체가 존재하는 위치, 즉, 물체의 내부점에서 높은 대칭도가 나타나므로 이러한 대칭점을 구함으로써 물체를 검출할 수 있게 된다.

그러나, 식 (3)과 식 (5)에 의해 잡음에 의한 대칭 기여도가 계속 누적되어 잡음이 많이 존재하는 영상에서는 물체를 검출하는 데에 어려움이 따른다. 그리고 다중 물체 검출에 적용할 경우, 대칭도에 포함될 영역의 범위, 즉, 대칭 마스크의 크기를 다르게 하여 여러 번 적용해야 한다. 또한, 대칭 마스크 크기보다 가까이 있는 두 물체 사이의 영역에서도 식 (1)에 의해 높은 대칭 기여도가 나타난다.

2. 다중 물체 검출을 위한 MRBFN의 구현

본 논문에서는 방사형 기반 함수로 인해 대칭성을 잘 반영하는 RBF 망을 이용하여 일반화 대칭 변환을 다중 물체 검출에 적합하도록 변형 구현하였다. 변형 구현한 RBF 망, 즉 MRBFN의 구조는 그림 2와 같다.

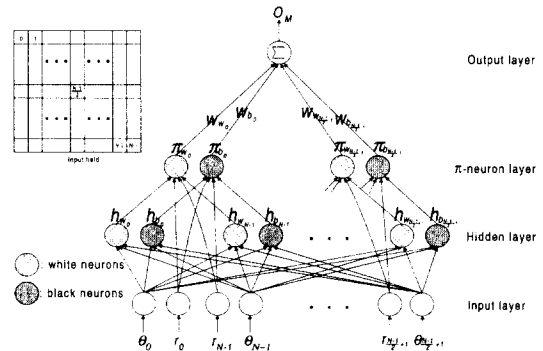


그림 2. MRBFN의 구조
 Fig. 2. Structure of the MRBFN.

그림에서 보는 바와 같이 은닉층과 출력층 사이에 곱셈을 수행하는 파이 뉴런이 추가되었으며, 그림 3에 나타낸 입력 영역 화소의 명도변화 방향의 수렴성과 발산성을 잘 반영시키기 위해 은닉층 뉴런과 파이 뉴런을 흑색 뉴런(black neurons)과 백색 뉴런(white neurons)으로 나누었다. 그림에서 θ_i 와 r_i 는 신경망의

입력으로서 입력 영역의 i 번째 화소 m_i 에서의 명도 변화의 방향과 크기를, h_b 와 h_w 는 은닉층의 흑색 뉴런과 백색 뉴런의 출력을 각각 의미하며 π_b 와 π_w 는 흑색 파이 뉴런과 백색 파이 뉴런의 출력을, w_b 는 흑색 파이 뉴런과 출력층 뉴런 사이의 연결 세기를, w_w 는 백색 파이 뉴런과 출력층 뉴런 사이의 연결 세기를 각각 나타내며 O_M 은 출력층 뉴런의 출력을 의미한다. MRBFN의 은닉층의 입력 벡터 I 를

$$I = [\theta_0, \theta_1, \dots, \theta_i, \dots, \theta_{N-1}] \quad (6)$$

와 같이 정의하였다. 식에서 N 은 입력 영역의 화소 개수를 의미한다. 그리고, 은닉층의 흑색 뉴런과 백색 뉴런의 출력 h_b 와 h_w 를 각각

$$h_b = \exp\left(-\frac{\|I - U_b\|^2}{2 \sigma_h^2}\right) \quad (7)$$

$$h_w = \exp\left(-\frac{\|I - U_w\|^2}{2 \sigma_h^2}\right) \quad (8)$$

와 같이 정의하였다. 이때 중심 벡터 U_b 와 U_w 를

$$U_b = [u_{b_0}, u_{b_1}, \dots, u_{b_k}, \dots, u_{b_{(N-1)}}],$$

$$u_{b_k} = \begin{cases} \arctan\left(-\frac{\partial}{\partial y}(m_k - m_{(N-1)/2}), \frac{\partial}{\partial x}(m_{(N-1)/2} - m_k)\right), & k=1 \\ 0, & k \neq 1 \end{cases} \quad (9)$$

$$U_w = [u_{w_0}, u_{w_1}, \dots, u_{w_k}, \dots, u_{w_{(N-1)}}],$$

$$u_{w_k} = \begin{cases} \arctan\left(\frac{\partial}{\partial y}(m_{(N-1)/2} - m_k), -\frac{\partial}{\partial x}(m_k - m_{(N-1)/2})\right), & k=1 \\ 0, & k \neq 1 \end{cases} \quad (10)$$

와 같이 각각 설정하였다. 식에서 m_k 는 입력 영역의 k 번째 위치의 화소를, $m_{(N-1)/2}$ 은 입력 영역의 중심에 위치하는 화소를 각각 의미한다. 즉, 흑색 뉴런의 중심 벡터를 명도변화 방향이 입력 영역의 중심으로 수렴하는 방향으로 설정하였으며 백색 뉴런의 중심 벡터를 입력 영역의 중심에서 외곽으로 발산하는 방향으로 설정하였다. 물체를 포함한 영역과 인접한 물체간 영역에서 화소의 명도변화 방향을 그림 3에 각각 나타내었다.

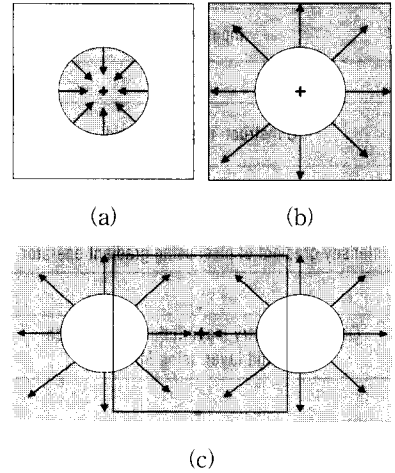


그림 3. 물체를 포함하는 영역에서 명도변화 방향의 수렴(a)과 발산(b) 및 물체간 영역에서의 명도 변화 방향(c)

Fig. 3. (a) The convergence and (b) the divergence of the orientation of the intensity gradient in the input field with object, and (c) the orientation of the intensity gradient in the input field with interobject region.

그림 3(a)에서 보는 바와 같이 배경보다 낮은 명도를 가지는 물체의 외곽선을 구성하는 화소의 명도변화 방향은 물체의 내부로 수렴하는 방향이며, 그림 3(b)와 같이 배경보다 높은 명도를 가지는 물체의 경우에는 물체의 내부에서 외곽으로 발산하는 방향이므로 물체를 포함한 영역에서는 흑색 뉴런이나 백색 뉴런이 큰 출력값을 나타낸다. 그리고, 그림 3(c)와 같이 인접한 물체간 영역에서는 명도변화 방향이 영역의 중심에 대해 수렴이나 발산을 하지 않는 방향이므로 흑색 뉴런과 백색 뉴런은 작은 출력값을 나타내게 된다.

한편, 파이 뉴런층의 흑색 파이 뉴런과 백색 파이 뉴런의 출력 π_b 와 π_w 를 각각

$$\pi_b = r_i r_{(N-1)-i} h_b, h_{b_{(N-1)-i}} \quad (11)$$

$$\pi_w = r_i r_{(N-1)-i} h_w, h_{w_{(N-1)-i}} \quad (12)$$

와 같이 정의하였다. 식에서 r_i 와 $r_{(N-1)-i}$ 는 입력 영역의 중심에 대해 대칭인 위치에 존재하는 화소들의 명도변화 크기를 각각 의미한다. 입력 영역의 중심에 대해 대칭인 위치에 있는 화소들의 명도변화 크기를 곱함으로써 화소의 명도변화 크기를 대칭도에 반영하도록 하였다. 또한, 은닉층 뉴런의 출력값들을 서로 곱

함으로써 대칭인 위치에 있는 화소들의 명도변화 방향의 대칭성도 반영하도록 하여 일반화 대칭 변환의 위상 가중 함수의 효과를 얻도록 하였다.

그리고, 흑색 파이 뉴런과 출력층 뉴런 사이의 연결세기 w_{b_i} 와 백색 파이 뉴런과 출력층 뉴런 사이의 연결세기 w_{w_i} 를

$$w_{b_i} = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{\|m_i - m^{(N-1)-i}\|^2}{2\sigma^2}\right) \quad (13)$$

$$w_{w_i} = -\frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{\|m_i - m^{(N-1)-i}\|^2}{2\sigma^2}\right) \quad (14)$$

와 같이 각각 설정하였다. 식에서 σ^2 은 입력 영역의 크기, 즉 마스크의 크기를 의미한다. 입력 영역에서 서로 대칭 위치에 존재하는 화소들간의 거리에 대한 가우시안 함수 값으로 연결세기를 설정하여 대칭 위치에 있는 화소간의 거리를 대칭도에 반영함으로써 일반화 대칭 변환의 거리 가중 함수와 같은 역할을 하도록 하였다.

또한, 흑색 파이 뉴런과 출력층 뉴런 사이의 연결세기를 양수로 설정하고, 백색 파이 뉴런과 출력층 뉴런과의 연결세기를 음수로 설정함으로써 입력 영역 화소의 명도변화 방향의 일관성을 대칭도에 반영하도록 하였다. 물체가 존재하는 영역에서는 은닉층의 흑색 뉴런과 백색 뉴런 중 한 종류만이 큰 값을 출력하므로 파이 뉴런과 출력층 뉴런 사이의 연결세기에 의해 서로 상쇄되지 않는다. 그러나, 잡음이 존재하는 영역에서는 명도변화 방향이 일관적으로 수렴하거나 발산하지 않으므로 은닉층의 흑색 뉴런 및 백색 뉴런의 출력값은 작으면서도 비슷하게 되어 그들 출력값은 파이 뉴런과 출력층 뉴런 사이의 연결세기에 의해 서로 상쇄되므로 잡음에 의한 대칭도는 억제된다. 따라서, MRBFN의 최종 출력값은

$$O_M = \sum_{i=0}^{\frac{(N-1)}{2}-1} (\pi_b w_{b_i} + \pi_w w_{w_i}) \quad (15)$$

으로 되어 입력 영역 화소 쌍의 명도변화 방향이 대칭일수록, 명도변화 크기가 클수록, 그리고 화소간의 거리가 가까울수록 큰 값을 출력하여 일반화 대칭 변환과 같은 효과를 얻을 수 있도록 하였다. 또한, 입력 영역 화소의 명도변화 방향의 수렴성, 발산성 및 일관성을 대칭도에 반영할 수 있도록 MRBFN의 중심 벡

터 및 연결세기를 설정함으로써 인접한 물체간 대칭성에 의한 대칭도와 잡음에 의한 대칭도를 억제할 수 있었다.

3. 피라미드 구조를 이용한 효과적인 다중 물체 검출 MRBFN의 입력 영역보다 물체가 클 경우, MRBFN은 물체의 부분적인 대칭성만을 반영하므로 물체의 모서리 부분이나 외곽선이 추출되며, 망의 입력 영역보다 물체가 작을 경우에는 물체의 전체적인 대칭성을 반영하므로 물체 내부의 대칭점이 검출된다. 따라서, 크기가 다른 여러 물체가 존재하는 영상에서 각 물체를 검출하기 위해서는 입력층 뉴런 및 은닉층 뉴런의 개수가 다른 여러 개의 신경망을 이용해야 하므로 효과적이지 못하다. 따라서, 효과적인 다중 물체 검출을 위해서는 고정된 크기를 갖는 한 개의 신경망을 이용하여 다중 물체를 검출하는 방안이 필요하다.

본 논문에서는 입력 영상을 피라미드 구조로 변환한 후, MRBFN을 이용하여 다중 물체를 검출하도록 하였다. 하위층의 4개 화소들의 명도를 평균하여 상위층의 1개 화소의 명도로 사상(mapping)시키는 평균 피라미드(mean-pyramid) 기법을 사용하였으며, 따라서, 사상 $g(i, j)$ 은

$$g(i, j) = \frac{1}{4} \sum_{x=0}^1 \sum_{y=0}^1 g_{l+1}(2i+x, 2j+y), \quad l = 0, \dots, L \quad (16)$$

와 같이 정의된다. 식에서 L 은 피라미드 층의 개수를 의미한다.

식을 통해 상위층으로 갈수록 물체의 크기가 1/4씩 축소됨을 알 수 있다. 따라서, MRBFN을 이용하여 하위층에서는 물체의 모서리나 외곽선에서 침투치가 나타나는 대칭도 맵을 얻고, 상위층에서는 물체의 내부에서 침투치가 나타나는 대칭도 맵을 얻은 후, 각 층에서 구해진 대칭도 맵을 누적한 3차원 대칭도 맵에서 침투치를 구하여 다중 물체를 검출하였다. 또한, 상위층으로 갈수록 영상의 크기가 1/4씩 감소되므로 물체간의 거리도 줄어들게 되어 물체간 대칭성이 증가되지만 MRBFN의 중심 벡터는 물체간 대칭성에 의한 대칭도를 억제하도록 설정되었으므로, 물체와 물체 사이의 영역을 검출하지 않고 순수 물체만을 검출할 수 있다. 그러나, 최상위층에서 물체들이 서로 붙지 않아야 하므로 입력 영상의 물체들은 최소 2^{L-1} 화소 이상 떨어져 있어야 한다.

III. 시뮬레이션 결과 및 고찰

제안한 다중 물체 검출 방법의 타당성과 성능을 확인하기 위하여 CCD 카메라를 이용해서 입력받은 여러 물체가 혼재하는 영상에 대해 시뮬레이션 하였다. 이때, 각 물체의 크기는 256×256 영상의 1%에서 10%까지 차지하도록 하였다. 제안한 방법에서는 먼저 입력된 영상에 대해 2×2 화소들의 명도 평균을 구하여 상위층의 1개 화소의 명도로 사상시키는 평균 피라미드 구조를 형성하였으며, 이때, 최상위층 크기를 물체의 형태가 보존될 수 있는 최소 크기 32×32로 설정하여 4개의 층으로 구성되는 피라미드를 형성하였다. 그리고, 각 층에 대해 경사 연산자를 사용하여 각 화소의 명도변화의 크기와 방향을 구한 후, MRBFN을 이용하여 각 층의 대칭도 맵을 구하고, 각 층의 대칭도 맵을 누적한 3차원 대칭도 맵의 침두치를 구하여 다중 물체를 검출하였다. 이때, MRBFN의 입력 영역 크기를 9×9로 설정하였다.

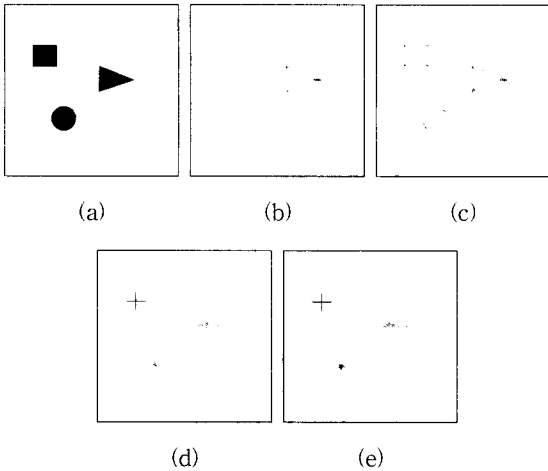


그림 4. 일반화 대칭 변환과 MRBFN 적용 결과 비교 (a) 여러 가지 다각형의 원영상, (b) 9×9 마스크의 일반화 대칭 변환 결과, (c) 9×9 입력 영역의 MRBFN 적용 결과, (d) 55×55 마스크의 일반화 대칭 변환 결과 및 (e) 55×55 입력 영역의 MRBFN 적용 결과

Fig. 4. Comparison of the MRBFN with the GST: (a) The original image with various polygons, (b) the result of the GST with 9×9 mask, (c) the output of the MRBFN with 9×9 input field, (d) the result of the GST with 55×55 mask, and (e) the output of the MRBFN with 55×55 input field.

먼저, MRBFN이 일반화 대칭 변환의 특성을 잘 반

영하는지 확인하기 위해 여러 가지 도형을 포함한 영상에 대해 시뮬레이션 하였다. 입력 영역에 대해 일반화 대칭 변환을 적용한 결과와 MRBFN을 이용하여 얻은 결과를 그림 4에 나타내었다. 그림에서 보면 MRBFN을 이용하여 얻은 결과와 일반화 대칭 변환을 적용한 결과가 동일함을 알 수 있으며, 따라서, MRBFN은 일반화 대칭 변환의 특성을 잘 반영하고 있음을 알 수 있다. 그림 5에는 인접한 물체들을 포함한 영상에 대해 일반화 대칭 변환을 적용한 결과와 MRBFN을 이용하여 얻은 결과를 나타내었다. 이때, 물체와 물체 사이의 거리는 1 화소이며 일반화 대칭 변환의 대칭 마스크 크기와 MRBFN의 입력 영역의 크기는 32×32이다.

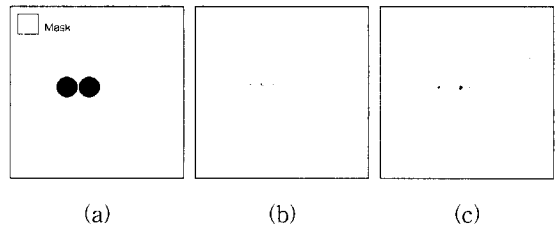


그림 5. 인접 물체의 일반화 대칭 변환과 MRBFN을 적용한 결과 비교 (a) 서로 인접한 두 물체의 원영상, (b) 일반화 대칭 변환 결과 및 (c) MRBFN 적용 결과

Fig. 5. Comparison of the MRBFN with the GST for adjacent objects: (a) The original image with two adjacent objects, (b) the result of the GST, and (c) the output of the MRBFN.

그림에서 보는 바와 같이 일반화 대칭 변환을 적용하면 물체와 물체 사이의 한 점에서도 대칭도 맵의 침두치가 나타나지만, MRBFN을 이용할 때는 각 물체의 내부에서만 침두치가 나타난다. 따라서, MRBFN의 은닉층 뉴런의 중심 벡터를 입력 영역 화소의 명도 변화 방향의 수렴성 및 발산성을 반영하도록 설정함으로써 서로 인접한 물체간 대칭성에 의한 대칭도가 억제됨을 알 수 있다. 그러므로, MRBFN은 일반화 대칭 변환을 영상에 적용할 때의 물체간의 거리에 대한 제약 조건을 완화시킬 수 있다. 그러나, 접촉된 두 물체를 하나의 물체로 검출하는 경우도 있으므로 물체와 물체 사이의 거리는 최소 1화소 이상이어야 한다.

한편, 15% 및 35% 가우시안 잡음이 첨가된 영상에 대해 일반화 대칭 변환을 적용하여 얻은 결과와 MRBFN을 이용하여 얻은 결과를 그림 6에 나타내었다.

그림 6을 보면 MRBFN은 일반화 대칭 변환보다

가우시안 잡음에 대해 둔감한 특성을 가지고 있음을 확인할 수 있다.

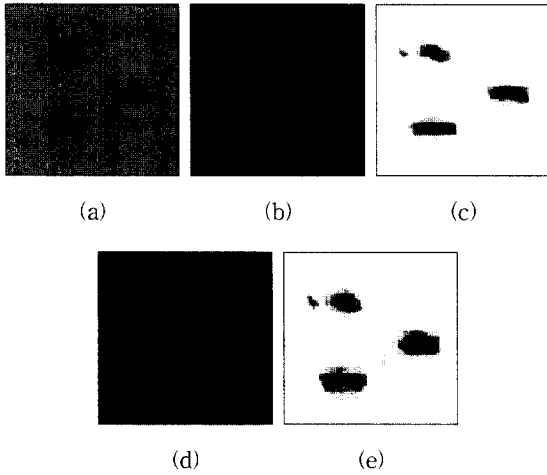


그림 6. 가우시안 잡음이 있는 영상에 대한 일반화 대칭 변환과 MRBFN의 물체 검출 결과 비교 (a) 원영상, (b) 15% 가우시안 잡음이 첨가된 영상의 일반화 대칭 변환 결과, (c) 같은 영상의 MRBFN 적용 결과, (d) 35% 가우시안 잡음이 첨가된 영상의 일반화 대칭 변환 결과 및 (e) 같은 영상의 MRBFN 적용 결과

Fig. 6. Comparison of the output of the MRBFN with the result of the GST in case for the noisy image: (a) The original image, (b) the result of the GST for the image with 15% Gaussian noise, (c) the output of the MRBFN for the same image, (d) the result of the GST for the image with 35% Gaussian noise, and (e) the output of the MRBFN for the same image.

따라서, MRBFN에서는 잡음에 의한 대칭도가 파이 뉴런과 출력층 뉴런 사이의 연결세기에 의해 서로 상쇄되므로 많은 잡음이 있는 영상에 대해서도 물체들을 잘 검출할 수 있음을 알 수 있다. 그림 7에는 MRBFN과 피라미드 구조를 사용하여 다중 물체들 검출하는 과정을 나타내었다.

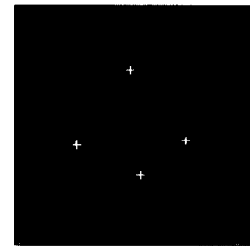
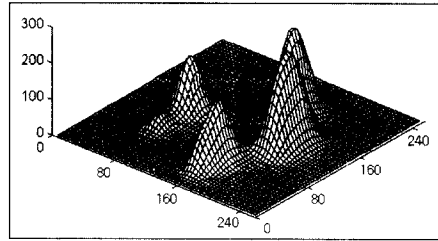
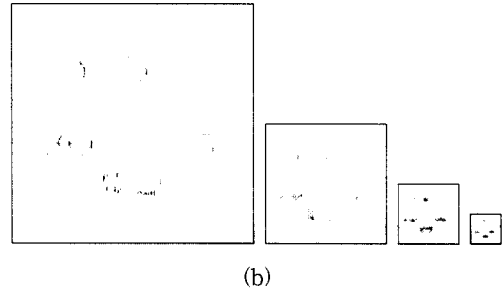
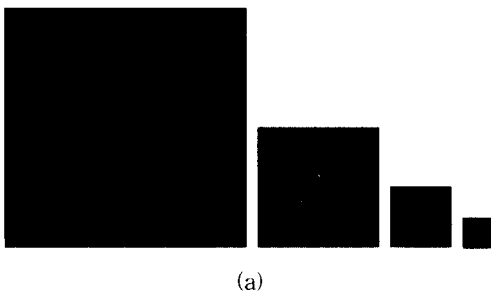


그림 7. (a) 다중 물체를 포함한 원영상 및 그 피라미드 영상, (b) 각 층의 대칭도 맵, (c) 3차원 대칭도 맵 및 (d) 검출된 물체

Fig. 7. (a) The original image with multiple objects and its image pyramid, (b) symmetry magnitude map of each layer of the pyramid, (c) 3-dimensional symmetry magnitude map, and (d) the detected objects.

그림 7(b)에서 보는 바와 같이 피라미드의 최하위층에서는 신경망 입력 영역의 크기가 물체의 크기보다 작기 때문에 물체의 모서리나 외곽선이 추출된다. 또한, 상위층으로 갈수록 물체가 축소되어 신경망 입력 영역 안에 물체가 포함됨으로써 물체의 내부에서 높은 대칭도가 나타난다. 각 층의 대칭도 맵을 누적하여 그림 7(c)와 같은 3차원 대칭도 맵을 얻을 수 있으며, 첨두치를 구함으로써 그림 7(d)와 같은 물체의 존재 위치를 십자(+) 표시로 나타낸 물체 검출 영상을 얻을 수 있다. 그림을 통해 MRBFN과 피라미드 구조를 사용하여 형태와 크기가 다른 다중 물체들을 모두 검

출할 수 있으며, 서로 인접한 물체들에 대해서도 물체와 물체 사이의 영역을 검출하지 않고 순수 물체만 검출할 수 있음을 알 수 있다. 제한한 물체 검출 방법을 배경 숲과 잔디밭에 강아지가 있는 실영상에 대해 적용한 결과를 그림 8에 나타내었다.

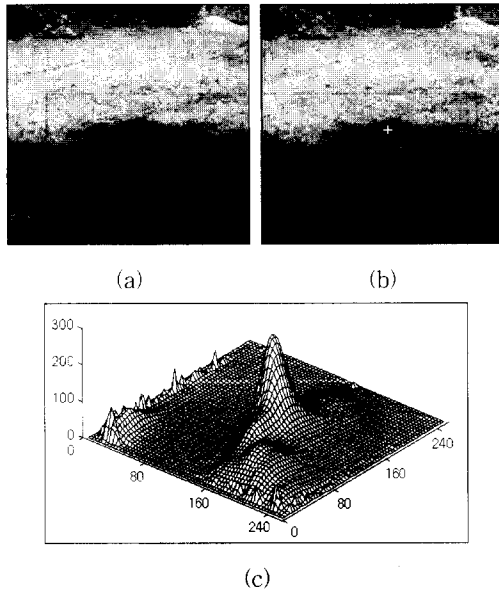


그림 8. (a) 풀밭에 강아지가 있는 실영상과 (b) 물체 검출 영상 및 (c) 3차원 대칭도 맵
Fig. 8. (a) The real image with a puppy on the grass, (b) the object detected image, and (c) 3-dimensional symmetry magnitude map.

그림에서 보면 잔디 등과 같이 화소의 명도변화 방향의 수렴성과 발산성 및 일관성이 없는 영역은 검출되지 않으며 강아지와 같이 대칭성이 강한 영역이 검출됨을 알 수 있다. 그러나, 그림 8(b)의 검출된 대칭점 위치는 물체를 충분히 포함하는 크기의 입력 영역을 가지는 MRBFN을 이용하여 얻은 실제 대칭점의 위치와 x축으로 3 화소, y축으로 4 화소만큼의 차이가 있다. 이는 입력 영상을 피라미드 구조로 변환하는 과정에서 하위층의 4개 화소가 상위층의 1개 화소로 사상되므로 상위층으로 갈수록 물체의 위치가 실제 존재 위치에서 최대 2 화소씩 차이가 나기 때문이다. 이 방법을 쓰게 되면 4층 피라미드로 인한 검출 위치 변동은 최대 4 화소를 넘지 않는다. 따라서, 본 논문에서와 같이 물체의 크기를 입력 영상 크기의 1% 이상으로 가정했을 때 최대 오차가 발생하더라도 검출된 대칭점은 물체 내부점을 벗어나지 않게 된다. 그러나,

물체 내부의 정확한 대칭점의 위치를 필요로 하는 응용 분야에 이용하기 위해서는 피라미드 구조를 사용함으로써 발생하는 실제 대칭점 위치와 검출된 대칭점 위치와의 차이를 보정해 줄 수 있는 방법이 필요하다.

IV. 결 론

본 논문에서는 RBF 망을 이용하여 물체에 대한 사전 정보 없이도 다중 물체를 효과적으로 검출할 수 있는 관심 영역 추출자를 구현하였다. RBF 망의 은닉층과 출력층 사이에 피어 뉴런을 추가하여 일반화 대칭 변환을 변형 구현하여 물체 사이의 좁은 영역은 물체로 검출하지 않고 순수 물체들만을 검출하였으며 잡음에 의한 대칭도도 누적되지 않도록 하였다. 또한, 입력 영상을 피라미드 구조로 변환한 후 제안한 MRBFN을 이용함으로써 단일 크기의 신경망으로도 여러 크기의 다중 물체를 효과적으로 검출할 수 있었다.

제안한 알고리즘을 여러 물체가 포함된 영상에 대해 시뮬레이션해 본 결과, 단일 MRBFN을 가지고도 크기가 다른 다중 물체들을 물체에 대한 사전 정보 없이 쉽게 검출할 수 있었다. 뿐만 아니라, 피라미드의 최상위층에서 두 물체가 최소한 1 화소 이상 떨어져 있으면 물체와 물체 사이의 영역을 검출하지 않고 순수 물체만을 검출할 수 있으며, 35%의 가우시안 잡음이 있는 영상에서도 물체들을 잘 검출할 수 있음을 확인하였다. 다만, 피라미드 구조를 사용하기 때문에 발생하는 실제 대칭점과 검출된 대칭점 사이에 위치 오차를 보정하고 또한, 서로 겹쳐진 여러 개의 물체들도 구분 검출할 수 있는 방법에 대하여도 좀 더 연구되어야 하겠다.

참 고 문 헌

- [1] Z. Yao, M. Wu, G. Jin, G. Huang, and Y. Yan, "New optoelectronic morphological scheme for multiple-object recognition," *Optical Engineering*, vol. 33, no. 11, pp. 3727-3732, Nov. 1994
- [2] S. D. You, "Preprocessing network for multiple objects," *IEEE International Conference on Neural Networks*, pp. 4149-4153, Orlando, USA, 1994
- [3] R. J. Liou and M. R. Azimi-Sadjadi, "Mul-

- tiple target detection and track identification using modified high order correlations," *IEEE International Conference on Neural Networks*, pp. 3277-3282, Orlando, USA, 1994
- [4] F. Goudail and P. Réfrégier, "Optimal detection of a target with random gray levels on a spatially disjoint background noise," *Optics Letters*, vol. 21, no. 7, pp. 495-497, April 1996
- [5] S. Ghosal and D. C. McKee, "Target detection in foveal ATR systems," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 714-719, San Francisco, California, 1996
- [6] M. M. Trivedi and C. Chen, "Object detection by step-wise analysis of spectral, spatial, and topographic features," *Computer Vision, Graphics, and Image Processing*, vol. 51, no. 3, pp. 235-255, Sep. 1990
- [7] D. Reisfeld, H. Wolfson, and Y. Yeshurun, "Context-free attentional operators: The generalized symmetry transform," *International Journal of Computer Vision*, vol. 14, pp. 119-130, 1995
- [8] S. Haykin, *Neural Networks*, Macmillan Publishing Company, New York, 1994
- [9] I. Pitas, *Digital Image Processing Algorithms*, Prentice Hall, New York, 1993

 저자 소개



許 娟 珠(正會員)

1972년 1월생. 1995년 2월 경북대학교 전자공학과 졸업(공학사). 1997년 2월 경북대학교 전자공학과 석사과정 졸업(공학석사). 1997년 2월 ~ 1997년 9월 현재 국방과학연구소 연구원 재직중. 주관심 분야는 컴퓨터

비전, 신경회로망 및 신호처리 등임

朴 昶 俊(正會員) 第33卷 B編 第10號 參照

현재 경북대학교 전자공학과 박사과정 재학중

崔 興 文(正會員) 第33卷 B編 第10號 參照

현재 경북대학교 전자전기공학부 교수

論文97-34C-10-1

효율적인 실시간 영상처리용 2-D 컨볼루션 필터 칩

(An Efficient 2-D Convolver Chip for Real-Time Image Processing)

殷世英*, 鮮于明勳*

(Se Young Eun and Myung Hoon Sunwoo)

요 약

본 논문은 곱셈기가 없는 새로운 구조의 영상처리용 디지털 필터를 제안한다. 2차원 컨볼루션 필터링은 화질개선의 방법으로 가장 많이 사용된다. 현재 상용화된 2차원 컨볼루션 필터칩은 영상 데이터의 실시간 처리를 만족하기 위해서 다수의 곱셈기를 사용한다. 따라서 칩의 크기가 크며 처리속도 대 크기의 비율이 낮은 단점이 있다. 본 논문에서 제안하는 실시간 영상처리용 컨볼루션 필터칩은 곱셈기를 사용하지 않는 대신 하나의 shift-and-accumulator만을 사용함으로써 기존의 상용 칩에 비해 크기를 70% 이상 줄일 수 있다. 또한 2차원 영상 신호 뿐만 아니라 1차원 신호도 처리할 수 있도록 설계하였다. 제안된 필터 칩은 VHDL로 모델링한 후 SYNOPSIS™ CAD tool로 논리합성하여 구현하였다. 논리합성은 삼성 SOG 셀 라이브러리를 사용하였으며 타이밍 시뮬레이션을 수행하였다. 구현된 필터는 3,893개의 게이트로 구성되었으며 최대 동작 주파수는 125 MHz이고 MPEG-2 영상 규격인 720×480 크기 영상의 실시간 처리 조건(10.4 Mpixel/second)을 만족한다.

Abstract

This paper proposes a new real-time 2-D convolver filter architecture without using any multiplier. To meet the massive amount of computations for real-time image processing, several commercial 2-D convolver chips have many multipliers occupying large VLSI area. The proposed architecture using only one shift-and-accumulator can reduce the chip size by more than 70 % of commercial 2-D convolver filter chips and can meet the real-time image processing requirement, i.e., the standard of CCIR601. In addition, the proposed chip can be used for not only 2-D image processing but also 1-D signal processing and has good scalability for higher speed applications. We have simulated the architecture by using VHDL models and have performed logic synthesis. We used the Samsung SOG cell library(KG60K) and verified completely function and timing simulations. The implemented filter chip consists of only 3,893 gates, operates at 125 MHz and can meet the real-time image processing requirement, that is, 720×480 pixels per frame and 30 frames per second(10.4 Mpixels/second).

I. 서 론

영상 신호의 필터링은 화질 개선(image enhancement)의 가장 기본적인 수단으로서 널리 사용되고 있다^[1-3]. 그러나 영상 신호는 데이터 속도와 연산량이 방대하여 실시간 필터링을 위해서는 고속의 필터가 필수적이다. 상용의 컨볼루션 필터의 경우 필터링 마

* 正會員, 亞州大學校 電機電子工學府

(School of Electrical and Electronic Eng., Ajou Univ.)

※ 이 연구는 1995년도 한국과학재단 연구비 지원에 의한 결과임.(과제번호 : 951-0915-124-2)

接受日: 1997년 4월 8일, 수정완료일: 1997년 9월 23일

스크의 크기에 비례한 다수의 곱셈기가 사용되며^[4] 이러한 곱셈기는 필터의 VLSI 구현시 칩(chip)의 면적을 많이 차지하므로 하드웨어 비용이 증가된다^[4,6], 따라서 필터의 응용분야에서 요구되는 처리속도를 감안하여 하드웨어 비용을 줄일 수 있는 구조로서 곱셈기를 사용하지 않는 필터의 설계가 제안되어왔다^[7-11]. 이러한 아키텍처는 일반적으로 power-of-two 계수 알고리즘을 바탕으로 하고 있으며 특히 특성 계수가 결정된 특별용도의 필터의 경우에는 별도의 SA(shift-and-accumulator)를 필요하지 않으므로 쉽게 처리속도 대 하드웨어 비용이 우수한 필터를 설계할 수 있다^[7-9].

특성 계수를 제어할 수 있는 프로그래머블 필터를 설계시 SA는 필수적이며 SA를 이용한 효과적인 필터 아키텍처가 제안되어 왔다^[10-11]. 그러나 제안된 방식의 아키텍처는 곱셈기 대신 SA를 모든 필터링 마스크나 탭에 사용하므로 하드웨어 비용 감소를 크게 기대할 수 없다.

본 논문에서는 곱셈기를 사용하지 않으며 하나의 SA를 사용하므로써 하드웨어 비용을 최소화할 수 있는 컨볼루션 필터 칩의 아키텍처를 제안하고 구현하였다. 구현된 필터 칩은 MPEG-2 표준안(CCIR601)인 초당 30 프레임, 720×480의 영상 데이터(약 10.4 Mpixel/sec)^[12]를 실시간 필터링 할 수 있다. 제안된 필터는 VHDL(VHSIC Hardware Description Language)로 모델링하여 논리 합성하였으며 삼성 SOG(Sea-of-Gate) 셀을 사용하여 그 기능과 타이밍 시뮬레이션을 확인하였다. 최고 동작 주파수는 125 MHz로서 15.625 Mpixel/sec의 전송율을 가지는 영상 데이터를 실시간 처리할 수 있다. 전체 게이트 수는 3,893개이며 현재 상용화된 HARRIS semiconductor사의 3×3 마스크 영상 필터 칩인 HSP 48901^[4]과 비교했을 때 약 70% 이상 게이트 수가 감소하였다. 보다 효과적인 비교로서 처리속도 대 크기의 비율(speed/size ratio)은 약 2배 증가하였다.

본 논문은 다음과 같이 구성되어있다. II절에서는 설계된 필터의 수식적인 알고리즘과 아키텍처를 설명하고, III절에서 제안한 필터의 설계에 대해 설명하며, IV절에서 곱셈기가 있는 필터 및 기존의 SA를 사용한 필터와의 비교 및 성능 평가를 서술하고 마지막으로 V절에서는 연구에서 얻은 성과 및 개선방안을 서술한다.

II. 설계된 필터의 알고리즘과 아키텍처

원도우 마스크의 크기가 NxM인 영상 필터의 입력과 출력의 수열이 각각 $F(x,y)$, $G(x,y)$ 이고 마스크의 계수(mask coefficient)가 $H(x,y)$ 이면 컨볼루션 필터의 입력과 출력의 관계는 식 (1)로 주어진다. 결국 필터를 하드웨어로 구현하면 그 연산부는 각각 NxM개의 데이터와 계수의 곱의 합(sum of products)을 구하는 구조가 됨을 알 수 있다. 식 (1)에서 x, y 는 영상의 좌표를 표시하는 정수이다.

$$G(x,y) = \sum_{n=0}^x \sum_{m=0}^y H(n,m)F(x-n,y-m) \quad (1)$$

본 논문에서 제안하는 필터를 설명하기 위해 8 비트 데이터와 계수에 대한 3x3 마스크를 갖는 필터로 가정하면 식 (1)은 식 (2)와 같다. 식 (2)를 직접 구현하면 그림 1과 같은 구조를 가지며 8 비트 데이터 레지스터(F) 9개와 8 비트 계수 레지스터(H) 9개의 출력을 각각 쌍으로 9개의 곱셈기에서 곱을 구하고 곱셈기의 16 비트 결과를 8개의 16 비트 가산기로 구성된 16 비트 트리형 가산기에서 모두 더함으로써 하나의 필터링 출력 샘플을 구한다. 이와 같은 구조는 9개의 곱셈기가 이미 언급한 바와 같이 VLSI 구현시 연산 구조의 대부분의 면적을 차지하는 구조로서 데이터처리 속도대 게이트 수 면에서 효과적이지 못하다.

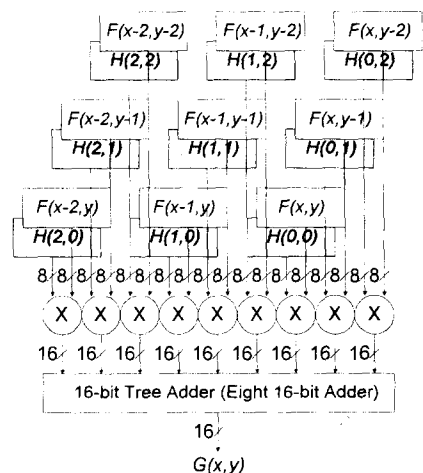


그림 1. 곱셈기를 사용한 필터의 구조
Fig. 1. The filter architecture using multipliers.

$$G(x,y) = \sum_{n=0}^2 \sum_{m=0}^2 H(n,m)F(x-n,y-m) \quad (2)$$

곱셈기대신 SA를 이용한 기존의 필터의 구조^[10-11]

1]를 본 논문에서 제안하는 필터와 비교하기 위해서 식 (2)를 이진수 체계로 표현하면 식 (3)과 같고 식 (3)에서 $H(n,m)=h_k(n,m)2^k$ 이며 $h_k(n,m)$ 는 0 또는 1 인 이진수이다. 식 (3)을 하드웨어로 구현하면 그림 2 와 같다.그림 2에서 계수 $H(n,m)$ 를 저장하는 레지스터는 쉬프트 로테이트 레지스터로서 하나의 클럭 사이 클에 한 비트씩 쉬프트하면서 계수의 각 비트를 MSB 부터 순차적으로 출력함으로써 논리곱 게이트가 데이터와 계수의 모든 부분곱을 출력 하도록 한다. 하나의 SA는 16비트 레지스터와 16비트 가산기로 구성되며 한 쌍의 데이터와 계수의 부분곱들을 쉬프트와 누적을 반복하여 8클럭 사이클 동안에 곱셈을 완성한다. 이와 같은 9개의 SA의 출력을 하단의 16 비트 트리형 가산기가 모두 더함으로써 하나의 필터링 출력 샘플을 구하게 된다. 이와 같은 구조는 그림 1과 같이 곱셈기를 사용하는 구조에 비하여 속도면에서는 단점이 있으나 하드웨어 비용이 적게 드는 장점이 있다. 그러나 9 개의 8x8 곱셈기를 72개의 논리곱 게이트, 9개의 16 비트 가산기, 9개의 accumulate 레지스터로 대체하는 것은 실제로 하드웨어 비용의 큰 감소를 가져오기 힘들다.

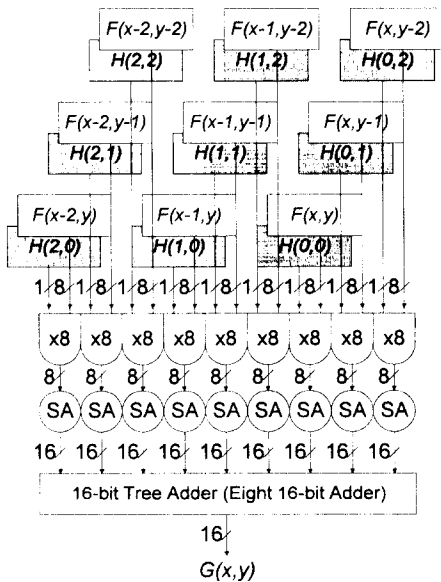


그림 2. 기존의 SA를 이용한 필터의 구조
Fig. 2. The filter architecture using SAs.

$$G(x,y) = \sum_{n=0}^2 \sum_{m=0}^2 \sum_{k=0}^7 F(x-n,y-m) h_k(n,m)2^k \quad (3)$$

본 논문에서 제안하는 필터의 아키텍처는 식 (4)를

구현한 것으로 식 (4)는 식 (3)을 분배법칙과 결합법칙을 수행하여 구할 수 있으며 식을 적용한 구조는 그림 3과 같다.

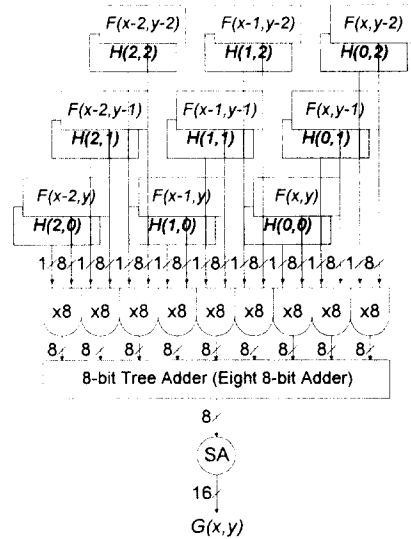


그림 3. 제안하는 필터의 구조
Fig. 3. The proposed filter architecture.

그림 2의 구조와 비교하면 각 데이터와 계수의 부분곱을 쉬프트 누적하여 곱을 구하기 전에 다른 데이터와 계수의 부분곱들을 모두 더한 후 SA에서 그 결과를 쉬프트하면서 누적하는 구조이다. 이렇게 함으로써 그림 1의 구조와 비교하였을 때에는 9개의 8x8 곱셈기와 16 비트 트리형 가산기 대신 하나의 SA와 8 비트 트리형 가산기를 사용할 수 있으며 그림 2의 구조와 비교하였을 때에는 9개의 SA와 16 비트 트리형 가산기 대신 1개의 SA와 8 비트 트리형 가산기를 사용할 수 있다. 따라서 하드웨어 비용을 그림 1의 구조와 그림 2의 구조에 비교하여 크게 감소시킬 수 있음을 알 수 있다. 또한 그림 2의 구조는 SA가 8 클럭 사이클동안 누적한 후 트리형 가산기에 유효한 데이터를 줄 수 있지만 그림 3의 구조에서는 부분곱을 논리곱 게이트에서 구한 후 바로 트리형 가산기에 유효한 데이터를 주므로 트리형 가산기에 파이프라인 아키텍처를 적용하기에 용이하다.

$$G(x,y) = \sum_{k=0}^7 \sum_{n=0}^2 \sum_{m=0}^2 F(x-n,y-m) h_k(n,m)2^k \quad (4)$$

III. 제안한 필터의 설계

본 논문에서 제안하는 필터는 3x3 마스크나 9개의

탭을 가지는 2차원 및 1차원 검용 필터이다. 연산부에는 각 계수 레지스터의 MSB와 각 데이터들을 논리곱한 9개의 부분곱을 더하는 트리 가산기가 있으며 트리 가산기의 출력을 계수 레지스터가 로테이트하여 MSB가 본래 값의 LSB가 되는 8클럭 동안 레프트 쉬프트 시키면서 더해주는 기능을 하는 16비트 가산기와 누적 레지스터가 있다.

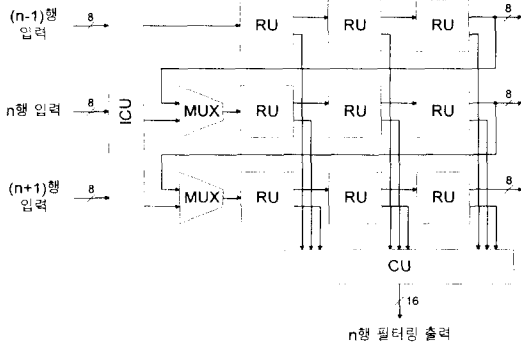


그림 4. 설계된 필터 칩의 전체 아키텍처
Fig. 4. The the designed filter chip.

그림 4는 제안된 필터를 구현하기 위해 구성한 아키텍처로서 9개의 레지스터 유니트(RU), 연산 유니트(CU), 입력 제어 유니트(ICU)의 부분력으로 구성된다. 맥스(Mux)와 디맥스(Demux)를 통해 1차원 신호와 2차원 신호에 맞게 데이터와 계수의 흐름을 조정한다. 따라서 1차원 신호의 필터링을 할 때에는 9개의 레지스터 유니트가 직렬로 연결된 9 탭으로 사용되며 2차원 영상 신호일 때에는 9개의 레지스터 유니트가 3×3의 계수 마스크와 데이터 윈도우로 사용된다. 레지스터 유니트들에서 출력된 9개의 부분곱을 입력 받은 연산 유니트는 모두 더한후 쉬프트 누적함으로써 여덟 클럭에 한번씩 필터링된 데이터를 출력한다.

레지스터 유니트의 구조는 그림 5와 같이 8비트의 데이터 레지스터와 계수 레지스터로 구성되어 있다. 각 레지스터 유니트의 출력은 데이터와 계수 한 비트의 부분곱이며 첫번째 클럭에 데이터와 계수 MSB의 부분곱을 출력하기 시작하여 마지막 여덟번째 클럭에 데이터와 계수 LSB의 부분곱을 출력한다. 계수 레지스터는 외부에서 인가되는 클럭(시스템 클럭)을 사용하며 데이터는 각 계수의 모든 비트와 한 번씩 부분곱을 해야하므로 시스템 클럭에서 주기가 8배로 분주된 클럭(데이터 클럭)에 동기되어 다음단의 레지스터 유니트로 이동된다.

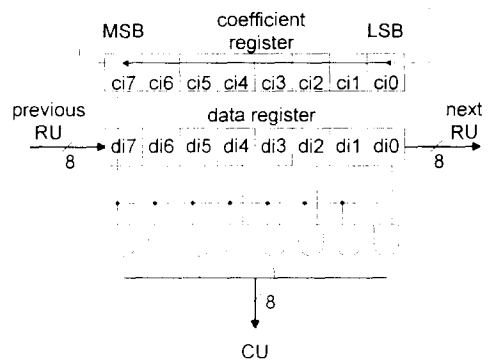


그림 5. 레지스터 유니트의 구조
Fig. 5. The Register Unit.

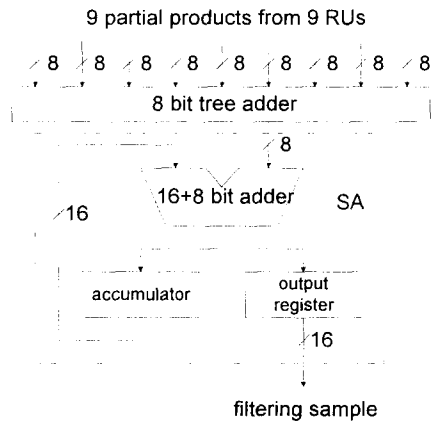


그림 6. 연산 유니트의 구조
Fig. 6. The Computation Unit.

연산 유니트의 구조는 그림 6과 같다. 8 비트 트리 가산기(tree adder)는 8개의 8비트 가산기가 3단으로 파이프라인되어 구성되며, 8비트 가산기는 2개의 4비트 CLA(carry look ahead) 가산기를 연결한 그룹 CLA이다. 16+8비트 가산기(16+8 bit adder)는 8비트 CLA 전 가산기(full adder)와 8비트 반 가산기(half adder)를 연결한 그룹 CLA이다. 따라서 전체 필터의 지연시간(delay time)은 비트수가 가장 큰 16+8비트 가산기에 의해 결정된다. 또 16+8비트 가산기의 출력은 누적 레지스터(acc register)를 거쳐 다시 16+8비트 가산기의 입력으로 궤환(feedback)되어 입력는데 누적 레지스터를 쉬프트 레지스터로 쓰는 대신 누적 레지스터의 출력을 가산기 입력에 직접 1비트씩 상위 비트로 쉬프트시켜 연결하여 곱셈 기능을 수행하며 이는 쉬프트에 소요되는 한 클럭을 단축시킨다. 따라서 16+8비트 가산기의 현재의 출력은 과거 출력의 2배(1비트 쉬프트)값에 현재 트리 가산기의 출력

값을 더한 값이 되며 이 과정은 계수 레지스터가 로테이트하여 다시 본래의 값을 갖는 8클럭을 주기로 반복 수행한다. 또한 한 주기의 마지막 클럭에서는 출력 레지스터(output register)는 가산기의 출력을 새로 저장하고 누적 레지스터는 다음 주기에 현 주기의 값을 넘겨 주지 않도록 초기화 된다.

컨볼루션 필터링에서는 한 행의 필터 출력 값을 구하기 위해 세개 행의 입력이 필요하며 따라서 두개의 과거행 데이터를 저장하는 행 버퍼가 필요하다. 이러한 행 버퍼는 칩의 크기와 확장성 문제로 하나의 칩에 집적하기에 어려우며 따라서 행 버퍼를 외부에 두거나 필터의 3개 행 입력을 메모리로 부터 모두 받는 방법을 사용할 수 있는데 메모리로 부터 3개 행의 데이터를 받는 경우 같은 영상 데이터를 3개의 메모리에 중복하여 저장하여야 한다. 본 논문에서 제안하는 필터는 입력 제어 유니트라는 간단한 FSM(Finite State Machine)을 두어 3개의 메모리에 각각 행의 수를 모듈로 3으로 구분하여 중복 없이 저장하여 사용할 수 있다. 즉 영상 데이터가 그림 7와 같이 (3k+1)행, (3k+2)행, 3k행 별로 구분되어 저장되어 있으면 각 행의 입력이 종료될 때 행전환 신호를 줌으로서 스위칭 상태(switching state)를 변화시켜줌으로써 그림 7의 우측 출력 열과 같이 영상 데이터를 필터링에 적합하도록 행 단위로 스위칭하여 출력하여 준다.

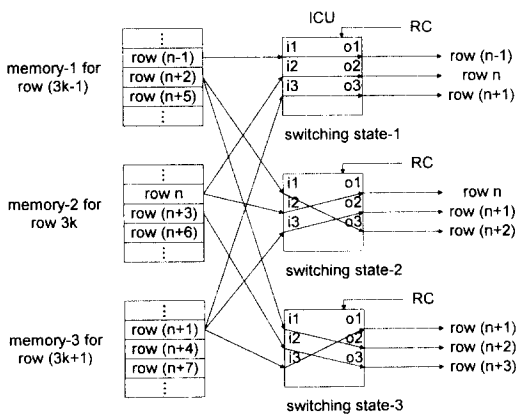


그림 7. 입력 제어 유니트의 구조
Fig. 7. The Input Control Unit.

IV. 기존의 필터와의 비교 및 성능평가

제안된 필터는 VHDL로 모델링 한후 SYNOP-

SYS™ CAD tool로 삼성 SOG 셀을 사용하여 논리 합성과 시뮬레이션을 수행하였다. 타이밍 시뮬레이션을 통하여 최대 시스템 클럭이 125 MHz(8 nsec 주기), 데이터의 입출력 속도가 15.625 MHz(64 nsec 주기)가 됨을 검증하였다. 그림 8은 제작된 필터 칩의 사진으로 3,893개의 게이트로 구성되었으며 47개의 동작 핀과 17개의 전력 핀을 가지고 있다.

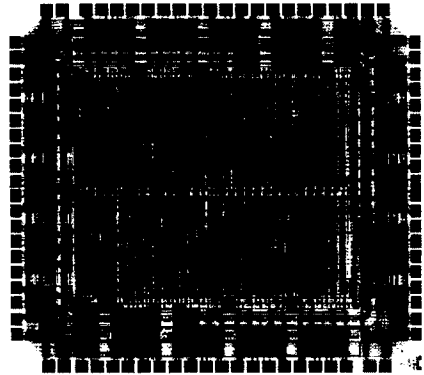


그림 8. 제작된 필터 칩의 사진
Fig. 8. The photograph of the implemented filter chip.

표 1의 곱셈기가 있는 HARRIS semiconductor사의 HSP48901 칩은 3×3 영상 필터로서^[4] 본 논문에서 제안한 필터와 같은 기능을 갖는 필터이며 객관적으로 비교하기 위해 제시하였다. 크기의 단위는 객관성을 유지하기 위해 공정 과정에 따라 변화 가능성이 많은 실제 물리적인 크기를 사용하지 않고 게이트 수로 하였다. HSP48901의 규격은 HARRIS semiconductor사의 매뉴얼^[4]을 참조했고 연산 블록의 게이트 수는 공개되지 않은 관계로 매뉴얼에 도식된 구조를 바탕으로하여 4개의 트랜지스터(transistor)를 1개의 게이트로 계산하여 얻은 수치이며 곱셈기는 9개의 배열형 곱셈기(array multiplier)를 사용한 것으로 가정하였다.

표 1에서 보인바와 같이 HSP48901은 13,594개의 게이트로 구성되며 본 논문에서 제안된 필터의 게이트 수는 3,893개이므로 약 71%의 게이트 감소 효과를 보였다. 또한 본 논문에서 제안된 필터는 720×480 크기의 영상신호를 초당 30 프레임으로 실시간 처리할 수 있다. 데이터 속도 대 크기의 비율면에서 1.8배 효과적이며 실제 필터링을 실행하는 연산 블록의 비율은 약 3.1배 우수하다.

표 1. 기존의 필터 칩과의 비교

Table 1. Comparison with other filter chips.

규격 \ 필터	HSP48001	기존의 SA를 이용한 필터	제안된 필터
시스템 클럭 속도	30 MHz	175 MHz	125 MHz
데이터 처리 속도	30 Mpixel/sec	176 Mpixel/sec	156 Mpixel/sec
게이트 수	13,594 gates	약 7,500 gates	3,893 gates
처리 속도/게이트 수	2.20 Kpixel/gate	약 2.34 Kpixel/gate	4.02 Kpixel/gate

또한 기존의 SA를 이용한 필터와의 비교에서는 데이터 처리속도는 동일한 수준을 유지할 수 있으며 SA를 9개에서 1개로 트리형 가산기의 비트수를 16 비트에서 8비트로 감소시켜 역시 하드웨어 비용을 크게 줄일 수 있다.

V. 결론

본 논문에서는 처리속도 대 크기의 비율을 향상시킬 수 있는 컨볼루션 필터 칩의 새로운 구조를 제안하고 설계하였다. 이는 곱셈기를 사용하지 않고 하나의 SA만을 사용하는 새로운 구조로 게이트 수를 기존의 곱셈기를 사용한 필터의 70% 이상 감소시키고도 720×480 크기의 영상 데이터를 초당 30 프레임으로 실시간 처리할 수 있는 필터를 구현하였으며 처리속도 대 크기면에서도 우수함을 확인하였다. 또한 기존의 SA를 사용하는 필터에 비하여 SA의 수를 9개에서 1개로 줄였으며 트리형 가산기의 비트수는 16 비트에서 8비트로 감소시켰다. 제안된 필터는 삼성 SOG 셀 라이브러리를 사용하여 구현되었으며, 3,893개의 게이트로 구성되며 최대 동작 주파수는 125 MHz이다.

참고 문헌

- [1] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Addison Wesley, June 1993, pp. 189-200.
- [2] Milan Sonka, Vaclav Hlavac and Roger Boyle, *Image Processing, Analysis and Machine Vision*, Chapman and Hall, 1993, pp. 67-96.
- [3] Edward R. Dougherty, *Digital Image Processing Methods*, Marcel Dekker, 1994, pp. 1-19.
- [4] HARRIS semiconductor Inc., *Digital Signal Processing*, 1994.
- [5] Vijai K. Madisetti, *VLSI Digital Signal Processors*, Butterworth-Heinemann, 1995, pp. 121-129.
- [6] Neil H.E. Weste and Kamran Eshraghian, *Principles of CMOS VLSI Design*, Addison Wesley, 1993, pp. 317-325.
- [7] R. Jain, R. T. Yang, T. Yoshino, FIR-GEN: A computer-aided design system for high performance FIR filter integrated circuit, *IEEE Trans. Signal process.*, vol. 39, pp. 1655-1668, July 1991.
- [8] T. Yoshino, R. Jain, P. T. Yang, H. Davis, W. Gass, and A. H. Shah, A 100-MHz 64-tap FIR digital filter in 0.8- μ m BiCMOS gate array, *IEEE J. Solid-State Circuit.*, vol. 25, pp. 1494-1501, Dec. 1990.
- [9] M. Ishikawa *et al.*, Automatic layout synthesis for FIR filters using a silicon compiler, in *Proc., 1990 IEEE Int. Symp. Circuit Syst.*, May 1990, New Orleans, LA, pp. 2588-2591.
- [10] Woo Jin Oh and Yong Hoon Lee, Implementation of Programmable Multiplierless FIR Filters with Power-of-two Coefficient, *IEEE Trans. Circuits Syst.*, vol. 42, pp. 553-555, August 1995.
- [11] K. Khoo, A. Kwentus, and A. N. Willson, Jr., An efficient 175MHz Programmable FIR digital Filter, in *Proc. 1993 IEEE Int. Conf. Circuits Syst.*, Chicago, IL, pp. 72-75, May 1993.
- [12] ISO-IEC/JTC1/SC29/WG11, MPEG92/229 (revised), *Information on requirements for MPEG-2 Video*, Jul. 1992.

저 자 소 개



殷 世 英(正會員)

1996년 2월 아주대학교 전자공학 학
사. 1996년 2월 ~ 현재 아주대학교
전자공학 석사과정. 관심분야는 통신
및 신호처리용 ASIC 설계

鮮于 明勳(正會員) 第 34卷 C編 第 8號 參照