

# 제공근 연산 횟수 감소를 이용한 Canny Edge 검출에서의 전력 소모개선

홍석희\*, 이주성\*\*, 안호명\*\*, 구지훈\*\*\*, 김병철\*\*\*\*

## Improvement of Power Consumption of Canny Edge Detection Using Reduction in Number of Calculations at Square Root

Seokhee Hong\*, Juseong Lee\*\*, Ho-Myoung An\*\*, Jihun Koo\*\*\*, Byuncheul Kim\*\*\*\*

**요약** 본 논문에서는 영상처리에 사용되는 Canny edge 검출 알고리즘 중 가장 높은 연산 복잡도를 가진 제공근 연산 횟수를 감소시키는 방법을 제안한다. 제안하는 방법은 기울기 벡터 연산 과정에 사용되는 제공근 연산을 이용할 때 일부 픽셀에 특정한 규칙을 사용해 홀을 만들어 제공근 연산을 직접 하지 않고 주변 픽셀들의 연속성을 이용하여 기울기 벡터를 계산하여 연산 횟수를 감소시킨다. 다양한 테스트 이미지를 이용해 실험한 결과 홀이 1개인 경우 약 97%, 홀을 증가시키면 각각 약 94%, 90%, 88%의 일치율을 보였고, 홀이 1개인 경우에는 0.2ms의 연산시간이 감소되었고, 홀을 증가시키면 각각 약 0.398ms, 0.6ms, 0.8ms의 연산시간이 감소되었다. 이를 바탕으로 hole이 2개인 경우 높은 정확도와 연산 수 절감을 통해 저전력 임베디드 비전 시스템을 구현할 수 있을 것으로 기대한다.

**Abstract** In this paper, we propose a method to reduce the square root computation having high computation complexity in Canny edge detection algorithm using image processing. The proposed method is to reduce the number of operation calculating gradient magnitude using pixel's continuity using make a specific pattern instead of square root computation in gradient magnitude calculating operation. Using various test images and changing number of hole pixels, we can check for calculate match rate about 97% for one hole, and 94%, 90%, 88% when the number of hole is increased and measure decreasing computation time about 0.2ms for one hole, and 0.398ms, 0.6ms, 0.8ms when the number of hole is increased. Through this method, we expect to implement low power embedded vision system through high accuracy and a reduced operation number using two-hole pixels.

**Key Words** : Canny edge detection, computation reuse, embedded vision, low computational complexity, low power image processing.

### 1. 서론

최근 머신 비전 기술을 기반으로 하는 자율주행 자동차 등 다양한 임베디드 비전 관련 연구가 활발히 진행되고 있다. 많은 영상처리 블록 중에서 물체와 차선을 인식하고 이탈 시 경고 알람을 주는 기능

은 자율주행을 수행하는 데 가장 중요한 블록 중 하나다. 자율주행 기술 또한 배터리 전력량에 제한된 임베디드 시스템이기 때문에 저전력으로 구현될 필요가 있다[1]. 이러한 연구 동향에 따라 기존 연구는 저전력 임베디드 비전 시스템을 구현하기 위해 하드웨어 공유[2], 중복 연산 제거[3], 알고리즘 재구성을

This work was supported by Gyeongnam National University of Science and Technology Grant 2020.

\*Department of Electronics, Myongji University.

\*\*Department of Electronics, Osan University.

\*\*\*Department of Smart IT, Osan University.

\*\*\*\*Corresponding Author : Department of Electronic Engineering, Gyeongnam National University of Science and Technology.(bckim@gntech.ac.kr)

Received November 8, 2020

Revised November 13, 2020

Accepted November 22, 2020

통한 연산 간소화[4-6], 연산 재사용 기법[7]과 같이 연구를 수행해 왔다.

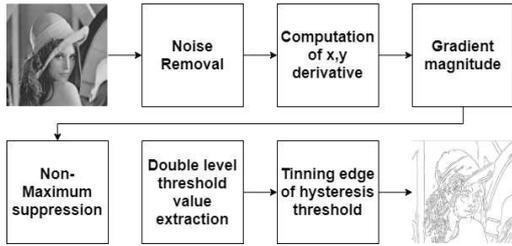


그림 15. Canny 윤곽선 검출 알고리즘 블록 다이어그램.  
Fig. 1. Canny edge detection algorithm block diagram.

자율주행에서 차선을 인식하기 위해 윤곽선 검출 알고리즘이 적용되는데, Canny 윤곽선 검출 알고리즘[8]은 가장 높은 윤곽선 검출 정확도를 보여 가장 많이 적용된다. Canny 윤곽선 검출 알고리즘은 그림 1에 나타낸 블록 다이어그램과 같이 1) 입력 영상의 잡음을 줄여주는 가우시안 필터, 2) X, Y 방향의 derivative를 구하는 Sobel 필터, 3) 윤곽선 후보군을 검출하는 기울기 벡터 연산, 4) 불필요한 후보군을 제거해 윤곽선일 확률이 높은 후보군만 남기는 Non-maximum suppression (NMS), 5) 윤곽선 도출 시 기준값으로 활용되는 double level threshold 값 도출, 6) 윤곽선의 연결성을 분석하기 위한 hysteresis thresholding로 구성되어 있다[8]. 대부분의 연산은 컨볼루션 혹은 메모리에 값을 저장해 통계적 연산을 수행하는 과정으로 구성되어 높은 연산 복잡도를 보인다. 특히 기울기 벡터 연산은 Euclidean distance를 구하는 과정으로 제공된 연산이 포함되어 가장 큰 연산 복잡도를 갖는다. 구현 복잡도 문제를 해결하기 위해 기존에 다양한 연구가 진행되었지만, 여전히 다른 블록에 비해 연산 복잡도가 높아 저전력으로 동작시키기 위한 연구가 필요하다[9].

본 논문에서는 기울기 벡터의 연산 복잡도를 줄이기 위해, 동작 횟수의 감소(reduced number of operations, RNO) 알고리즘을 제안한다. 제안된 알고리즘은 기울기 벡터의 연산 복잡도 문제를 해결하기 위해 특정 패턴에 대해 연산을 생략하고, 홀 생

성 후 이웃 픽셀의 기울기 벡터값 연산 결과를 재사용하여 이미지 전체 해상도 크기만큼 요구되는 연산 수를 줄이도록 한다. 제안한 알고리즘을 적용한 윤곽선 검출 결과와 기존 알고리즘을 사용한 결과의 일치율을 확인하여 적용 효과를 정량적으로 검증하도록 한다.

## 2. 기존 연구

기울기 벡터 블록은 Canny 윤곽선 검출 알고리즘에서 윤곽선 후보군을 도출하는 역할을 한다. Sobel 필터로부터 출력된 X, Y-derivative를 이용해 Euclidean distance를 도출한다. 이는 수식 (1)과 같이 표현된다.

$$M(x,y) = \sqrt{g_x^2 + g_y^2} \quad (1)$$

수식에서 x와 y는 해당 픽셀의 좌표,  $g_x$ 와  $g_y$ 는 Sobel 필터에 의해 도출된 각 X, Y-derivative를 나타낸다. 수식 (1)에 보인 것과 같이 연산을 위해 한 번의 덧셈, 두 번의 곱셈, 그리고 제공된 연산이 요구된다. 제공된 연산을 계산하기 위해 요구되는 알고리즘의 수식은 (2)처럼 표현된다.

$$e_n = \begin{cases} 1 & n=0 \\ (e_{n-1} + (Number/e_{n-1}))/2 & 1 \leq n < i \end{cases} \quad (2)$$

수식(2)에서  $e_n$ 는 제공된 결과값, Number는 입력 값이다. 연산은  $e_n$  값이 미리 지정한 값 이하로 수렴할 때까지 i 번 반복해 실행한다. 이 과정에서 나눗셈이 i 번 반복 실행되는데, 나눗셈 연산은 덧셈 및 곱셈 연산에 비해 많은 컴퓨팅 자원을 사용한다. Canny 윤곽선 알고리즘에서 제공된 연산은 단위 픽셀마다 요구되는데, 이는 입력 영상이 커지면 커질수록 해상도에 비해 상당량의 전력 소모를 유발하는 문제를 유발한다.

## 3. Reduced Number of Operations

### 알고리즘

Canny 윤곽선 검출 알고리즘은 회색조 영상을 입력받아 가우시안 필터로 영상 잡음을 제거한다.

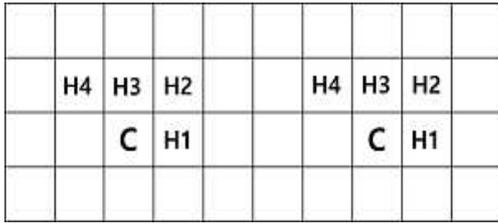


그림 2. 기울기 벡터 홀 생성 패턴.  
Fig. 2. Creation hole pattern of Gradient magnitude.

Sobel 필터를 적용해 X, Y 방향 gradient를 계산하고, 기울기 벡터를 계산한다. 불필요한 윤곽선 후보군을 제거하는 NMS와 동시에 검출 기준값으로 사용할 High, Low 경계값을 계산하고 Hysteresis thresholding을 수행해 윤곽선 검출결과 이미지를 생성한다.

윤곽선 후보를 도출하는 기울기 벡터 블록은 제공된 연산으로 인해 다른 블록에 비해 높은 연산 복잡도가 요구된다. 제안하는 RNO 알고리즘은 요구되는 기울기 벡터 연산의 절대적인 횟수를 줄이는 방법을 제안한다.

제안하는 알고리즘은 다음과 같다. 기울기 벡터를 구하는 과정에서 그림 2에 보인 패턴을 이용하여 홀(H1~H4)을 지정한다. 일정 거리를 두어 홀 픽셀을 지정했는데 이는 홀 간의 거리가 가까우면 worst case의 경우 오류 전파 현상에 의해 정확도가 상당히 감소하기 때문이다. 이때 중심 픽셀(C)을 기준으로 0°, 45°, 90°, 135°에 해당하는 픽셀인 H1, H2, H3, H4를 홀 픽셀로 지정한 뒤 기울기 벡터의 값을 계산하지 않고, 0으로 지정한다. 계산하지 않은 위치의 값은 주변에 존재하는 기울기 벡터를 재사용해 근사화한다. 디지털 이미지는 주변 픽셀 간에 값의 연속성이 존재하기 때문에 특정 위치값을 계산하지 않고 기울기 벡터의 방향성을 확인해 동일 선상에 놓인 주변값으로 근사화 하는 방식을 적용해 제공된 연산 횟수를 감소시켜 에너지 효율적인 영상처리를 수행할 수 있다.

그림 3은 제안하는 RNO 알고리즘의 다이어그램을 나타낸다. 먼저 영상의 Index 순서대로 모든 픽셀에 대해 hole 픽셀 유무를 확인한다. 이때 홀 픽

```

Algorithm 1 Reduced Number of Operations(RNO Algorithm)
1: Set some Gradient Magnitude as hole pixels(Gh)
2: Set the Gradient Magnitude's value to zero and skip computation
3: Calculate theta value of neighborhood pixels around hole pixel
4: Check the two theta values(t1,t2) around the hole pixel
5: if t1=t2
6:   then Gh=(t1+t2)/2
7: else
8:   Gh=average of 8 pixels around a hole pixel
9:   if Having zero value of pixels among the pixels around the hole
10:    then except the average calculate
11:   endif
12: endif
13: return <Gh>
14: This process calculates a hole pixel's Gradient Magnitude using the continuity of the pixels.
15: Gh is Gradient Magnitude's value of hole
16: t1,t2 are pixels those have similar edge's direction with Hole Pixel
    
```

그림 3. 제안하는 Reduced Number of Operations (RNO) 알고리즘 다이어그램.  
Fig. 3. Proposed algorithm diagram Reduced Number of Operations (RNO).

셀인 경우 이를 중심으로 주변 픽셀들의 방향성을 나타내는 theta 값을 확인한다. Theta는 X 및 Y 방향 gradient를 이용해 계산되고, 이웃하는 픽셀들을 모두 고려할 수 있는 0°~360° 범위이며 1) 0°~22.5° 및 337.5°~360°, 157.5°~202.5°에 해당하는 영역은 방향 값을 1로 지정, 2) 22.5°~67.5°, 202.5°~247.5° 영역은 방향 값을 2로 지정, 3) 67.5°~112.5°, 247.5°~292.5° 영역은 방향 값을 3으로 지정, 4) 112.5°~157.5°, 292.5°~337.5° 영역은 방향 값을 4로 지정한다. 이러한 방식으로 방향 값을 결정해 적용한 이유는 디지털 이미지의 방향 연속성 특성에 기반을 둔다 [9]. 이미지 픽셀마다 기울기 벡터값을 확인해 홀 픽셀을 발견하게 되면, 홀 픽셀을 중심으로 동일한 방향에 놓인 두 픽셀의 theta 값을 확인한다. 두 픽셀의 방향 값이 앞서 설정한 네 가지의 경우 내에서 일치할 때, 두 픽셀의 기울기 벡터값의 평균값으로 홀의 기울기 벡터값을 결정한다. 만약 동일 방향성 내 두 픽셀의 방향 값이 일치하지 않는 경우, 주변 8개의 픽셀의 기울기 벡터 평균값을 홀의 기울기 벡터값으로 설정한다. 이때 주변 픽셀에서 기울기 벡터의 값이 0인 픽셀은 제외하고 나머지 픽셀들의 평균값을 사용해서 홀 픽셀의 기울기 벡터를 계산한다.

### 4. 실험 결과

본 Section에서는 제안된 알고리즘을 적용한 결과와 정량적인 분석을 보인다. 그림 4는 제안된 알고리즘을 적용한 결과 이미지를 나타낸다. 대표적인 테스트 이미지인 Lena, Baboon, Peppers, Barbara를 사용하여 결과를 도출했다. 기울기 벡터 블록을 제외하고 모두 기존과 같은 알고리즘[8]이 결과 도출에 적용되었다. 표 1은 알고리즘을 통해 도출된 윤곽선 픽셀의 전체 개수, 기존 알고리즘과 제안된 알고리즘의 공통된 윤곽선 픽셀의 개수와 일치율, 다르게 검출된 픽셀의 수를 나타낸다. 1) 먼저 Lena 이미지를 분석하면 홀이 1개인 경우 17,576개의 edge 픽셀이 검출된다. 기존 알고리즘과 제안된 알고리즘의 공통 edge 픽셀의 개수는 17,096개로 일치율은 97.46%이다. 다음으로 홀이 2개인 경우

17,473의 edge 픽셀이 나오게 되고 픽셀은 16,638개가 나오며 94.83%의 일치율이 나오게 된다. 홀이 3개인 경우 17,481개의 edge 픽셀이 나오고 16,359개의 공통된 edge를 발견하게 된다. 이때 93.23%의 일치율을 확인할 수 있다. 그리고 홀이 4개인 경우 17,517개의 edge 픽셀이 나오고 16,138개의 공통된 edge 픽셀이 나오게 되어 91.985의 일치율을 보여준다. 2) 위의 방식을 사용하여 Baboon 이미지를 분석하면 홀이 1개인 경우 96.44%의 일치율을 확인할 수 있고 홀이 2개인 경우 93.57%의 일치율을 확인할 수 있다. 홀이 3개인 경우 90.56%의 일치율, 홀이 4개인 경우 87.3%의 일치율을 보여준다. 3) 다음으로 Pepper 이미지를 분석하면 홀이 1개인 경

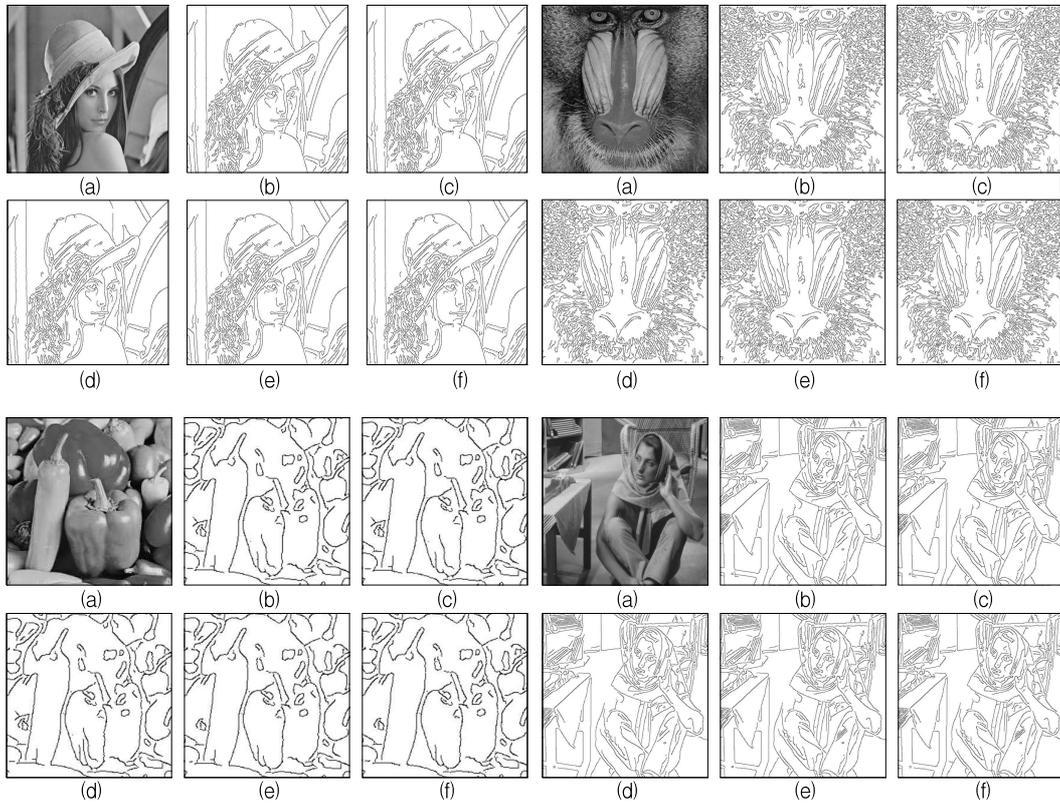


그림 4. 검출 결과 이미지. (a) 입력 이미지, (b) 기존 알고리즘, (c) ~ (f) RNO 알고리즘 결과(홀 1개, 2개, 3개, 4개)  
 Fig. 4. RNO algorithm result image, from (a), original image, local function image form (c)-(f), hole 1, 2, 3, 4.

표 1. RNO 알고리즘 결과 도표  
Table 1. RNO algorithm result chart

LENA (512*512)	Hole 1개 (10,404)	Hole 2개 (20,808)	Hole 3개 (31,212)	Hole 4개 (41,616)
Edge	17,576	17,473	17,481	17,517
Common	17,096	16,638	16,359	16,138
Difference	480	835	1,122	1,379
Match rate	97.46%	94.83%	93.23%	91.98%

PEPPERS (225*255)	Hole 1개 (2,025)	Hole 2개 (4,050)	Hole 3개 (6,075)	Hole 4개 (8,100)
Edge	4,013	4,013	3,802	3,791
Common	3,899	3,786	3,557	3,461
Difference	114	227	245	330
Match rate	97.14%	94.32%	88.61%	86.22%

BABOON (480*500)	Hole 1개 (9,600)	Hole 2개 (19,200)	Hole 3개 (28,800)	Hole 4개 (38,400)
Edge	32,557	32,244	31,966	31,518
Common	31,486	30,548	29,566	28,503
Difference	1,071	2,678	2,400	3,015
Match rate	96.44%	93.57%	90.56%	87.30%

BARBARA (512*512)	Hole 1개 (10,404)	Hole 2개 (20,808)	Hole 3개 (31,212)	Hole 4개 (41,616)
Edge	19,863	19,749	19,779	19,808
Common	19,351	18,862	18,474	18,114
Difference	512	887	1,305	1,694
Match rate	97.50%	95.03%	93.08%	91.26%

우 97.14%의 일치율을 확인할 수 있고 홀이 2개인 경우 94.32%가 나오게 되며 홀이 3개일 때는 88.61%의 일치율, 홀이 4개인 경우에는 86.22%의 일치율을 보여준다. 4) 마지막으로 Barbara 이미지를 분석했다. 홀이 1개인 경우 95.03%의 기존 이미지와의 일치율을 확인할 수 있고 홀이 2개인 경우 95.03%의 일치율을 확인할 수 있다. 홀이 3개인 경우 93.08%의 일치율, 홀이 4개인 경우 91.26%의 일치율을 보여준다. 이러한 과정을 통해 다양한 이미지에서 패턴당 홀의 개수를 증가시키면 홀의 개수만큼 제공근 연산 횟수가 감소하게 되고 위의 과정을 통해 영상처리에 사용되는 연산시간을 감소시킬 수 있다.

제안된 알고리즘의 홀 개수에 따른 연산시간 감소를 확인해보았다. i5-8265U CPU와 8GB의 RAM을 사용하는 pc에서 opencv-3.4.9 환경에서 실험한 결과 제공근 연산의 연산시간을 확인해보면 512\*512의 이미지에서 제공근 연산을 전부 실시한 경우 약 5.2ms의 연산시간이 소모되었다. 제공근 연산을 1회 수행 시, 0.02us가 소모되었고 패턴당 홀을 1개씩 발생시키면 약 0.208ms의 연산시간이 감소되는 것을 확인할 수 있다. 패턴당 홀을 2개씩 발생시키면 0.416ms, 3개면 0.624ms, 4개면 0.832ms의 연산시간 감소를 보인다.

### 5. 결론

본 논문에서는 Canny edge 검출 알고리즘의 기울기 벡터 연산에 요구되는 제공근 연산 횟수를 절감하기 위한 방법을 제안했다. 디지털 이미지에서 발생하는 데이터 연속성을 이용해 기울기 벡터를 구하는 과정에서 특정 픽셀을 홀로 지정해 제공근 연산을 생략하고, 주변 픽셀의 윤곽선 방향을 고려해 연산을 생략한 위치의 기울기 벡터를 주변 값을 이용해 근사값으로 대체하는 방식을 적용했다. 그 결과 매 패턴마다 연산을 두 번씩 생략해도 테스트 이미지에서 93%~95% 같이 높은 윤곽선 검출 일치율을 확인할 수 있으며 제안된 알고리즘을 적용해 저전력 임베디드 비전 시스템을 구축할 수 있을 것으로 기대한다.

### REFERENCES

[1] W. Kim, J. Lee, H. An, and J. Kim, "High-Performance and Low-Complexity image Pre-Processing Method Based on Gradient-Vector Characteristics and Hardware-Block Sharing", *Trans. Electr. Electron. Mater.*, (TEEM), vol. 18, no. 6, pp. 320-322, Dec.2017.

[2] W. Kim, J. Lee, and H. An, "Gradient Magnitude Hardware Architecture based on Hardware Folding Design Method for Low Power image Feature Extraction Hardware Design", *Journal of Korea institute of in-*

formation, electronics, and communication technology (KIIECT), vol. 10, no. 2, pp. 141-146, Apr. 2017.

[3] W. Kim, J. Lee, H. An, and B. Kim, "image Filter Optimization Method based on common sub-expression elimination for Low Power image Feature Extraction Hardware Design", *Journal of Korea institute of information, electronics, and communication technology (KIIECT)*, vol. 10, no. 2, pp. 192-197, Apr. 2017.

[4] W. Kim, J. Lee, and H. An, "Low Complexity Gradient Magnitude Calculator Hardware Architecture Using Characteristic Analysis of Projection Vector and Hardware Resource Sharing", *Journal of Korea institute of information, electronics, and communication technology (KIIECT)*, vol. 9, no. 4, pp. 414-418, Aug. 2016.

[5] J. Lee, H. An, and B. Kim, "Low Complexity image Thresholding Based on Block Type Classification for Implementation of the Low Power Feature Extraction Algorithm", *Journal of Korea institute of information, electronics, and communication technology (KIIECT)*, vol. 10, no. 2, pp. 141-146, Apr. 2017.

[6] J. Lee and H. An, "A Study on Implementation of the High Speed Feature Extraction System Based on Block Type Classification", *Journal of Korea institute of information, electronics, and communication technology (KIIECT)*, vol. 12, no. 3, pp. 186-191, Jun. 2019.

[7] J. Lee, H. An, and J. Kim, "Implementation of the High-Speed Feature Extraction Algorithm Based on Energy Efficient Threshold Value Selection", *Trans. Electr. Electron. Mater. (TEEM)*, vol. 21, pp. 150-156, Feb. 2020.

[8] J. Canny, "A Computational Approach to Edge Detection", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986.

[9] J. Lee, H. Tang, and J. Park, "Energy efficient canny edge detector for advanced mobile vision applications" *IEEE Trans. on Circuits and*

*Systems for Video Technology*, vol. 28, no. 4, pp. 1037-1046, Apr. 2018.

---

저자 약력

---

**홍 석 희(Seokhee Hong)**

[정회원]



- 2017년 2월 : 오산대학교 전자과 (전문학사)
- 2021년 2월 : 명지대학교 전자공학 (공학사)

〈관심 분야〉

저전력 영상처리, 비휘발성메모리, 광소자

**이 주 성(Juseong Lee)**

[정회원]



- 2013년 2월 : 서울과학기술대학교 전자미디어공학과 (공학사)
- 2015년 8월 : 고려대학교 전기전자공학과 (공학석사)
- 2018년 11월 : (재)실감교류인체감응솔루션연구단 (연구원)

〈관심 분야〉

VLSI 설계, 저전력 영상처리

**안 호 명(Ho-Myoung An)**

[종신회원]



- 2001년 2월 : 광운대학교 전자재료공학과 (공학사)
- 2003년 2월 : 광운대학교 전자재료공학과 (공학석사)
- 2009년 2월 : 광운대학교 전자재료공학과 (공학박사)
- 2009년 3월 ~ 2013년 2월 : 고려대학교 전기전자전자공학과 연구교수
- 2013년 3월 ~ 현재 : 오산대학교 전자과 부교수

〈관심 분야〉

반도체 소자, 비휘발성메모리, 광소자, 인공지능

**구 지 훈(Jihun Koo)**

**[정회원]**



- 2000년 2월 : 인하대학교 전기공학  
학과 (공학사)
  - 2002년 2월 : 인하대학교 전기공학  
학과 (공학석사)
  - 2015년 2월 : 연세대학교 전기전  
자공학과 (공학박사)
  - 2002년 4월 ~ 2017년 2월 :삼성  
전자 S.LSI Connectivity 개발팀  
수석연구원
  - 2017년 3월 ~ 현재 : 오산대학교  
스마트IT과 조교수
- 무선통신, MIMO, 사물인터넷, 신호  
처리

〈관심 분야〉

**김 병 철(Byungcheul Kim)**

**[중신회원]**



- 1988년 2월 : 광운대학교 전자재  
료공학과 (공학사)
- 1990년 2월 : 광운대학교 전자재  
료공학과 (공학석사)
- 2001년 2월 : 광운대학교 전자재  
료공학과 (공학박사)
- 2001년 9월 ~ 현재 :  
경남과학기술대학교 전자공학과  
교수

〈관심 분야〉

반도체 메모리, LED 전원회로