

확장 참조표를 활용한 허프변환의 성능 분석

오정수*

Performance Analysis of Hough Transform Using Extended Lookup Table

Jeong-su Oh*

*Professor, Department of Display Engineering, Pukyong National University, Pusan, 48513 Korea

요 약

본 논문은 대표적인 직선 검출 알고리즘인 허프변환이 갖고 있는 계산적인 부담을 줄이기 위해 확장된 참조표를 활용한 허프변환을 제안하고, 성능을 분석하고 있다. 기존 허프변환도 계산 부담을 줄이기 위해 관심 에지 화소를 지나는 모든 직선들의 매개 변수 ρ 계산에 참조표를 적용한다. 그러나 제안된 허프변환은 더 많은 계산 감소를 유도하기 위해 관심 에지 화소뿐만 아니라 그 주변 에지 화소들을 지나는 직선들에도 적용할 수 있는 확장 참조표를 채택하고 있다. 본 논문은 제안된 알고리즘의 유효성을 수학적으로 증명하고 또한 모의실험을 통해 확인하고 있다. 모의 실험 결과는 제안된 허프변환이 기존 허프변환과 비교해 곱셈 계산량을 영상과 적용된 확장 참조표에 따라 49.6%에서 최대 16.1%까지 감소시키는 것을 보여주고 있다.

ABSTRACT

This paper proposes the Hough transform(HT) using an extended lookup table(LUT) to reduce the computational burden of the HT that is a typical straight line detection algorithm, and analyzes its performance. The conventional HT also uses a LUT to the calculation of the parameter ρ of all straight lines passing through an edge pixel of interest(ePel) in order to reduce the computational burden. However, the proposed HT adopts an extended LUT that can be applied to straight lines across the ePel as well as its peripheral edge pixels to induce more computational reduction. This paper proves the validity of the proposed algorithm mathematically and also verifies it through simulation. The simulation results show that the proposed HT reduces the multiplication computation from 49.6% up to 16.1%, depending on the image and the applied extended LUT, compared to the conventional HT.

키워드 : 직선 검출, 허프변환, 참조표, 에지, 영역 분할

Keywords : Straight line detection, Hough transform, Lookup table, Edge, Segmentation

Received 29 September 2021, Revised 5 October 2021, Accepted 18 October 2021

* Corresponding Author Jeong-su Oh(E-mail: ojs@pknu.ac.kr, Tel:+82-51-629-6414)

Professor, Department of Display Engineering, Pukyong National University, Pusan 48513, Korea

Open Access <http://doi.org/10.6109/jkiice.2021.25.12.1868>

print ISSN: 2234-4772 online ISSN: 2288-4165

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

물체 내 혹은 물체 사이의 밝기 불연속에 의해 발생하는 직선은 물체를 특징 짓거나 구분하는 중요한 정보이다. 그래서 이런 직선의 검출은 물체의 기울림 검출, 부품 검출, 차선 검출 같은 컴퓨터 비전 분야의 핵심적인 기술이다[1-3]. 허프변환(Hough transform)은 밝기 불연속으로 표현되는 에지 영상을 근간으로 직선을 검출하는 대표적인 직선 검출 알고리즘이다[4-6]. 허프변환을 이용한 직선 검출은 에지 영상 내 에지 화소들에서 발생 가능한 모든 직선들을 극좌표계의 매개변수 (θ, ρ) 들로 표현하고, 매개 변수들을 $\theta-\rho$ 공간에 대응시켜 보팅(voting)하고, 보팅된 매개 변수 중 조건을 만족하는 것들을 유효 직선으로 검출하는 알고리즘이다[1-6]. 허프변환은 매개 변수 ρ 계산에 수행되는 반복 계산의 부담을 줄이기 위해 일반적으로 참조표(lookup table, LUT)을 이용하고 있다[7, 8]. 그러나 허프변환의 반복 계산은 영상의 고화질화에 따라 크기가 커진 영상이나 활용 영역의 다양화에 따른 낮은 성능 시스템 등에 적용되는 환경에서는 선제적으로 허프변환의 계산 부담을 줄여주는 것이 필요하다.

본 논문은 다양한 환경에서 적용될 허프변환의 반복 계산을 선제적으로 더 줄이기 위해 기존 참조표보다 확장된 참조표를 이용한 허프변환을 제안하고 성능을 분석한다. 기존 허프변환은 참조표를 관심 에지 화소에서 발생 가능한 직선들의 ρ 계산을 위해 반복되는 삼각 함수의 계산에만 적용되고 있다. 그러나 본 논문에서는 관심 에지 화소뿐만 아니라 그 주변 에지 화소들의 ρ 계산에도 적용될 수 있는 확장 참조표를 정의하고, 허프변환에 적용하여 계산량을 더욱 감소시키고 있다. 여기서 확장 참조표가 고려되는 관심 에지 화소의 주변 관심 영역은 마스크 형태로 지정되고, 마스크 크기, 관심 에지 화소의 위치와 이동 간격은 변경될 수 있고 이에 대해 성능을 분석하고 있다. 제안된 확장 참조표를 이용한 허프변환은 다양한 마스크 형태들에서 인위적인 영상들과 실제 영상들에 적용될 것이고, 허프변환을 위한 계산량과 부가 비용을 비교하여 제안된 알고리즘의 유효성을 보일 것이다.

II. 허프변환

허프변환은 그림 1(a)처럼 에지 영상인 $x-y$ 공간의 에지 화소들에서 발생 가능한 직선들의 매개 변수 (θ, ρ) 들을 그림 1(b)의 $\theta-\rho$ 공간에 대응시키는 변환이다[4-6]. 예를 들어 $x-y$ 공간의 에지 화소 $p_1(x_1, y_1)$ 에서 발생 가능한 직선들은 그림처럼 점 p_1 을 지나는 직선들로 점 p_1 을 중심으로 회전되는 직선들로 표현될 수 있고, 이 직선들의 (θ, ρ) 들을 계산하여 $\theta-\rho$ 공간에 보팅하는 것이다. 점 p_1 을 지나는 직선들의 매개변수 (θ, ρ) 들은 식 (1)에서 x, y 에 x_1, y_1 을 대입하고, 회전 각 θ ($0^\circ \sim 179^\circ$)에 따라 구분되는 직선들의 ρ 를 계산하므로 생성된다. θ 를 1° 단위로 변화시킬 때 $x-y$ 공간의 한 점은 $\theta-\rho$ 공간에서 (θ, ρ_l) 를 포함한 180개의 점으로 생성되는 곡선 P_1 에 대응된다. 직선 l 위의 다른 에지 화소 $p_2(x_2, y_2)$ 의 허프변환은 $\theta-\rho$ 공간에 다른 곡선 P_2 가 생성된다. 점 p_2 도 직선 l 위의 한 점이므로 $\theta-\rho$ 공간의 (θ, ρ_l) 을 지나 곡선 P_1 과 P_2 는 (θ, ρ_l) 에서 교차된다. 결국 직선 l 을 구성하는 에지 화소들은 모두 $\theta-\rho$ 공간의 (θ, ρ_l) 에 교차하면서 보팅되고 보팅 횟수는 직선을 구성하는 에지 화소 수 이상이다.

$$\rho(x, y, \theta_k) = x \sin \theta + y \cos \theta, \quad \theta_k = 0 : 1 : 179 \quad (1)$$

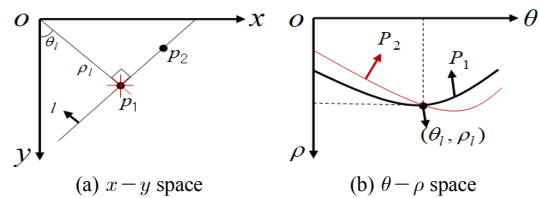


Fig. 1 Hough transform

그림 2는 기존 허프변환인 기본 참조표를 이용한 허프변환(HT_BasLUT)의 핵심 요소이다[7, 8]. 상위 블록은 반복 계산될 $\cos \theta$ 와 $\sin \theta$ 를 미리 계산하여 참조표 sLut[]와 cLut[]을 생성하고, 하위 블록은 관심 에지 화소(ePel[])에서 θ 에 따른 180개 직선들의 ρ 계산을 참조 처리된 삼각함수 계산에 2번의 곱셈과 1번의 덧셈으로 수행하고 보팅한다. 그래서 영상에 에지 화소가 N 개 있다면 순수 허프변환을 위해 수행된 곱셈과 덧셈의 계산량은 식 (2)와 같다. 여기서 $N \times$ 와 $N +$ 는 각각 곱셈과

덧셈의 횟수를 의미한다.

$$\begin{aligned} N \times : N \times 180 \times 2 \\ N + : N \times 180 \end{aligned} \quad (2)$$

```

for (A=0:179){ // BasLUT
  cLut[A] = cos(A);
  sLut[A] = sin(A);
}
for(y=0:imHeight-1){
for(x=0:imWidth-1){
  if(ePel[y][x]){ // edge pixels of interest
    for(A=0:179){ // straight lines
      rho=(int)(x*sLut[A]+y*cLut[A]); // +:1,*:2
      htImage[rho+mDist][A]++; // voting in rho-theta space
    } // mDist : for minus distance
  }
}
}

```

Fig. 2 Hough transform using the BasLUT

III. 확장 참조표를 활용한 허프변환

그림 2에 보듯이 허프변환은 수많은 직선들의 ρ 를 위해 반복 계산되는 $\cos\theta$ 와 $\sin\theta$ 의 계산을 피하려고 참조 처리하고 있다. 그런데 식 (3)에 전개된 식에 의하면 허프변환에 추가적인 참조 처리 영역의 존재를 보여주고 있고 이는 추가적인 허프변환의 계산 감소를 의미한다. 식 (3)을 정리해 보면, $\rho(x, y, \theta_k)$ 는 식 (1)에서 정의된 관심 에지 화소 (x, y) 에서 각이 θ_k 인 직선의 ρ 이고, $\rho(x+i, y+j, \theta_k)$ 는 (x, y) 에서 (i, j) 만큼 떨어진 주변 에지 화소 $(x+i, y+j)$ 에서 각이 θ_k 인 직선의 ρ 이다. $\rho(x+i, y+j, \theta_k)$ 을 전개하면 이는 이미 계산된 $\rho(x, y, \theta_k)$ 와 $i\sin\theta_k + j\cos\theta_k$ 의 덧셈으로 계산된다. 그리고 $i\sin\theta_k + j\cos\theta_k$ 를 확장 참조표 $ExtLUT(i, j, \theta_k)$ 로 대체하면 관심 에지 화소의 주변 에지 화소들에서 생성되는 직선의 $\rho(x+i, y+j, \theta_k)$ 는 식 (4)처럼 표현되고 한 번의 덧셈만으로 계산할 수 있게 된다.

$$\begin{aligned} \rho(x, y, \theta_k) &= x\sin\theta_k + y\cos\theta_k, \quad \theta_k = 0 : 1 : 180 \\ \rho(x+i, y+j, \theta_k) &= (x+i)\sin\theta_k + (y+j)\cos\theta_k \\ &= x\sin\theta_k + i\sin\theta_k + y\cos\theta_k + j\cos\theta_k \\ &= x\sin\theta_k + y\cos\theta_k + i\sin\theta_k + j\cos\theta_k \\ &= \rho(x, y, \theta_k) + i\sin\theta_k + j\cos\theta_k \end{aligned} \quad (3)$$

$$\rho(x+i, y+j, \theta_k) = \rho(x, y, \theta_k) + ExtLUT(i, j, \theta_k) \quad (4)$$

그림 3은 그림 4의 관심 에지 화소(ePel[y][x])와 주변 관심 영역(ROI, [y+j][x+i])을 대상으로 확장 참조표를 활용한 허프변환(HT_ExtLUT)의 핵심 요소이다. 상위 블록은 기존 허프변환보다 사전 계산이 추가된 확장 참조표 예 (그림 6(b) P4)를 생성하고 있고, 하위 블록은 ①관심 에지 화소에서 기존 알고리즘과 같이 직선들의 $\rho(x, y, \theta_k)$ 가 계산되고 있으나 그들의 ②주변 에지 화소들에서는 직선들의 $\rho(x+i, y+j, \theta_k)$ 가 이미 계산된 $\rho(x, y, \theta_k)$ 와 추가참조표 $ExtLUT(i, j, \theta_k)$ 의 덧셈만으로 계산되고 있다. 주변 에지 화소로 처리된 화소는 관심 에지 화소로 선택되지 않도록 제거된다. 따라서 에지 화소가 N개인 영상에서 순수 허프변환을 위해 수행된 계산량은 식 (5)와 같다. N.ePel은 관심 에지 화소로 허프변환이 수행된 화소의 수이다.

$$\begin{aligned} N \times : N.ePel \times 2 \times 180 \\ N + : N \times 180 \end{aligned} \quad (5)$$

```

for (A=0:179){ // ExtLUT (P4)
  cLut[A] = cos(A); sLut[A] = sin(A);
  Lut01[A] = sLut[A]; // LUT(y+0, x+1)
  Lut1_1[A] = cLut[A]-sLut[A]; // LUT(y+1, x-1)
  Lut10[A] = cLut[A]; // LUT(y+1, x+0)
  Lut11[A] = sLut[A]+cLut[A]; // LUT(y+1, x+1)
}
for(y=0:imHeight-1){
for(x=0:imWidth-1){
  if(ePel[y][x]){ // edge pixels of interest
    for(A=0:179){ // rho(x, y, theta_k)
      rhoM[A]=(int)(x*sLut[A]+y*cLut[A]); // ① +:1,*:2
      htImage[rhoM[A]+mDist][A]++;
    }
    if (ePel[y][x+1]) { // pixel (y+0, x+1)
      for (A=0:179) { // rho(x+i, y+j, theta_k)
        rho = (int)(rhoM[A]+Lut01[A]); // ② +:1,*:1
        htImage[rho+mDist][Angle]++;
      }
      ePel[y][x+1] = 0; // remove the edge pixel
    } // other surrounding pixels
    if (ePel[.][.]) {...} if(...){...} if(...){...}
  }
}
}
}

```

Fig. 3 Hough transform using the ExtLUT

	ePel[y][x]	[y][x+1]
[y+1][x-1]	[y+1][x]	[y+1][x+1]

Fig. 4 Pixel of interest and region of interest

IV. 제안된 알고리즘의 성능 분석

제안된 알고리즘의 성능을 평가를 위해 그림 5의 영상들이 사용된다. (a) Im1는 수직/수평 직선, 45° 사선으로 구성된 영상이고, (b) Im2와 (c) Im3는 각각 실제 빌딩과 책장 영상의 에지 영상이고, (d) Im4는 임의의 사선 도형 영상의 에지 영상이고, (e) Im5는 Im4의 원 영상에 15% 잡음이 포함된 영상의 에지 영상이다.

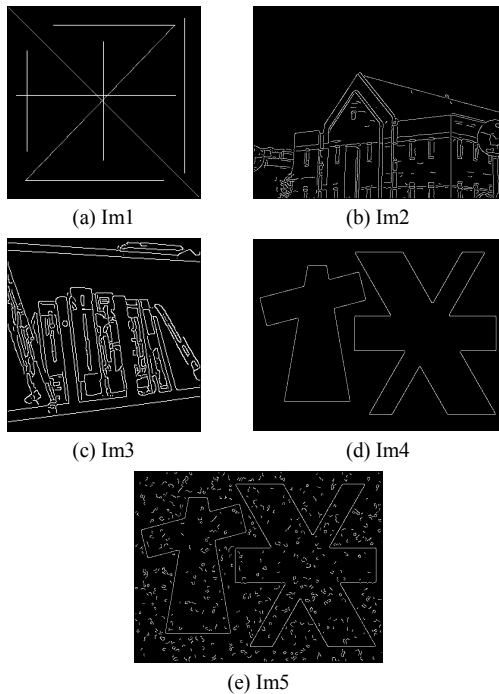


Fig. 5 Images for experiment

그림 6은 실험에 사용될 관심 에지 화소(ePel) 주변의 관심 영역을 위한 마스크(MROI)들이다. × 표시는 ePel의 위치이고, o 표시는 주변 관심 영역에서 확장 참조표가 적용될 화소들이다. 표시가 되지 않은 영역은 무시되는 화소들이다. 그림 6(a) Conv.은 확장 참조표가 사용될 주변 에지 화소들이 없으므로 기존 허프변환과 동일하다. 그림 6(f) P10C는 ePel ×가 관심 영역의 내부에 있는 마스크이다. 이는 ePel의 위치와 이동 간격에 대한 성능 평가를 고려한 마스크이다.

표 1은 ePel의 위치와 이동 간격에 따른 영향을 평가하기 위해 P10과 P10C가 적용된 HT_ExtLUT의 결과이다. 표는 에지 화소를 허프변환 처리 방법에 따라 구분

하여 비율로 보여주고 있다. N.ePel, N.pPel, N.sPel는 각각 관심 에지 화소들, 주변 관심 영역 에지 화소들, 미처리 에지 화소들로 허프변환된 화소들의 비율을 나타내고 있다. N.ePel는 적고, N.pPel이 많을수록 계산량 감소가 크게 된다. ePel이 마스크 내부에 포함된 P10C의 경우 이미 처리된 에지 화소들이 관심 영역 화소로 포함되어, step1에서 P10와 비교해 N.pPel가 50% 정도 적게 나타나 계산량 감소 효율이 떨어지고 있다. 또한 ePel의 이동 간격이 1 화소(step1)일 때는 미처리 화소가 없으나 이동 간격이 2 화소(step2)일 때는 에지 화소 전후로 ePel가 위치하여 이동하면 미처리 화소가 발생하고, 그 양이 40%를 넘기고 있다. 미처리 화소가 발생 시 허프변환을 위한 추가적인 스캔이 요구된다. 위에서 기술한 두 가지 특성은 다른 MROI에서도 유사한 결과를 보여주고 있어 ePel의 위치나 이동 간격의 변화는 적절하지 못하다는 결론을 내릴 수 있다.

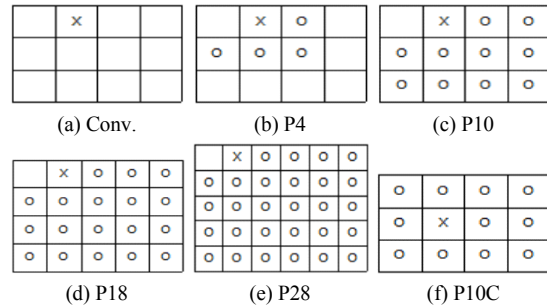


Fig. 6 Masks for the region of interest (MROI)

Table. 1 Results of according to the position and the step size of ePel [%]

method	P10		P10C	
	step1	step2	step1	step2
N.ePel	30.74	14.04	73.52	23.83
N.pPel	69.26	37.13	26.48	32.18
N.sPel	0.00	48.82	0.00	44.00

표 2와 3은 그림 6에 보여준 MROI에 따른 확장 참조표를 이용한 허프변환(HT_ExtLUT)의 성능을 비교하고 있다. Conv.는 앞에서 언급한 것처럼 관심 에지 화소만 사용하고 있어 기본 참조표를 이용한 허프변환(HT_BasLUT)와 같고 기존 알고리즘이라 볼 수 있고, 비교의 기준으로 사용되고 있다. N은 영상 내 에지 화소의 수이고, N.ePel는 허프변환을 위해 기존 방식으로 허

Table. 2 HT_ExtLUT's performance according to MROIs

Image	MROI	Conv.	P4	P10	P18	P28
Im1 (N=1502)	N.ePel	1502	745	521	416	348
	N.×	540720	268200	187560	149760	125280
	N.+	270360	270360	270360	270360	270360
Im2 (N=8553)	N.ePel	8553	3878	2629	1939	1442
	N.×	3079080	1396080	946440	698040	519120
	N.+	1539540	1539540	1539540	1539540	1539540
Im3 (N=6268)	N.ePel	6268	2898	1929	1381	1010
	N.×	2256480	1043280	694440	497160	363600
	N.+	1128240	1128240	1128240	1128240	1128240
Im4 (N=3366)	N.ePel	3366	1438	954	748	639
	N.×	1211760	517680	343440	269280	230040
	N.+	605880	605880	605880	605880	605880
Im5 (N=9681)	N.ePel	9681	4028	2788	2214	1833
	N.×	3485160	1450080	1003680	797040	659880
	N.+	1742580	1742580	1742580	1742580	1742580

프변환된 관심 에지 화소(ePel)의 수이고, N.×와 N.+는 각각 제안된 허프변환 HT_ExtLUT에서 사용된 곱셈과 덧셈의 계산 횟수이다. N.+는 식 (2)와 (5)에서 보여준 것처럼 영상의 N의 180배로 N에 의존한다. 그래서 표 2에서도 MROI에 관계없이 영상 단위로 모두 일치하는 것을 보여주고 있다. N.×는 식 (5)에서 보여준 것처럼 N.ePel의 360배로 N.ePel에 비례하고 그것을 표 2의 결과로 확인되고 있다. MROI의 크기와 N.ePel의 관계는 에지 화소 분포의 영향으로 절대적인 반비례 관계는 안 된다. 에지 화소가 집중된 영상에서 MROI가 클수록 ePel 주변에 많은 에지 화소가 포함되어 확장 참조되는 에지 화소들이 많아져 N.ePel가 작아진다. 반면 에지 화소가 흩어져 있으면 MROI가 커도 ePel 주변에 에지 화소가 적게 포함되어 N.ePel가 덜 작아진다. 결과적으로 에지 화소 집중도가 높은 Im2와 Im3에 큰 MROI가 적용된 경우에 N.ePel가 더 많이 감소한다.

Table. 3 Relative amount of N.× [%]

	Conv.	P4	P10	P18	P28
Im1	100	49.6	34.7	27.7	23.2
Im2	100	45.3	30.7	22.7	16.9
Im3	100	46.2	30.8	22.0	16.1
Im4	100	42.7	28.3	22.2	19.0
Im5	100	41.6	28.8	22.9	18.9

표 3은 MROI 형태에 따른 HT_ExtLUT의 N.×을 HT_BasLUT(Conv.)에 대한 백분율로 비교하고 있다. HT_BasLUT에 비해 적게는 작은 MROI에서 49.6%까지, 많게는 큰 MROI에서 16.1%까지 N.×을 감소시키고 있다. 또한 P4와 P10같이 작은 MROI에서는 단순 수직/수평 에지가 많은 영상에서 N.×을 더 감소시키고, P28처럼 큰 MROI에서는 앞에서 언급한 것처럼 실제 영상들에서 N.×을 더 감소시키는 특성을 보여주고 있다.

그림 7은 표 3에 보여준 결과들을 MROI의 크기 증가에 따른 평균 N.×의 감소 변화 추이를 보여주고 있다. 작은 MROI에서 N.×의 급격한 감소를 해주고 있으나 MROI가 클수록 N.× 감소율의 변화가 작아지는 특성을 보여주고 있다.

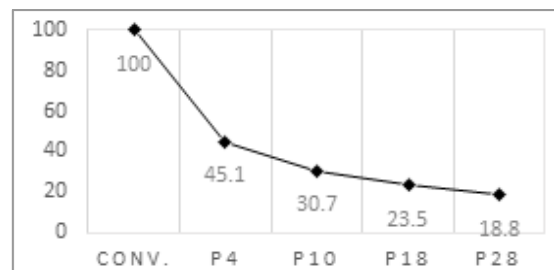


Fig. 7 Decrease trend of average N.× by MROIs

그림 8은 MROI의 크기 증가에 따라 부가적으로 증가하는 덧셈(N.+)과 곱셈(N.×) 계산량의 변화 추이를 보

여주고 있다. 부가적인 계산은 HT_BasLUT와 비교해 HT_ExtLUT에서 추가로 생성되는 확장 참조표를 위해 요구되는 계산량을 평가하였다. 그림에 보이는 것처럼 작은 MROI들에서는 부가 계산량이 무시될 정도로 작으나 MROI가 커질수록 급격하게 증가하는 것을 볼 수 있다. 또한 그림 3에서 예측할 수 있듯이 MROI가 급격하게 커질수록 참조표를 저장해야 할 메모리 공간과 확장 참조표가 적용될 주변 에지 화소를 검사하는 코드 복잡도도 고려되어야 할 것이다.

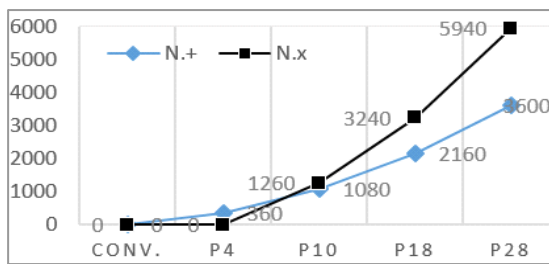


Fig. 8 Increase trend of additional calculations

V. 결론

본 논문은 기존 허프변환이 가진 반복 계산 부담을 줄이기 위해 확장 참조표를 이용한 허프변환을 제안하고 그 성능을 분석하고 있다. 기존 허프변환에서 관심 에지 화소에서 발생 가능한 직선들의 ρ 계산에만 적용하던 참조표를 제안된 허프변환에서는 관심 에지 화소는 물론 그 주변 에지 화소들에도 적용할 수 있는 확장된 참조표를 채택하고 있다. 모의실험 결과 제안된 허프변환에서 관심 에지 화소의 위치와 이동 간격을 변화시키는 것은 무의미함을 확인하였고, 확장 참조표의 적용 대상인 관심 영역이 작을수록 단순 수직/수평 에지 영상에 더 효율적이고, 관심 영역이 클수록 실제 영상처럼 에지 화소의 집중도가 높은 영상에서 더 효율적인 것을 확인하였다. 그리고 제안된 알고리즘은 관심 영역 크기에 따른 계산 감소와 부가 계산과 구현 복잡도를 고려할 때 적용될 환경에 따라 관심 영역 크기의 적절한 선택이 필요함도 보여주었다.

ACKNOWLEDGEMENT

This work was supported by a Research Grant of Pukyong National University (2021)

REFERENCES

- [1] Y. S. Wang, Y. Qi, and Y. Man, "An improved hough transform method for detecting forward vehicle and lane in road," *Journal of Physics: Conference Series*, vol. 1757, no. 1, pp. 24-25, 2021.
- [2] S. Das and R. Kumari, "Application of Extended Hough Transform Technique for Stationary Images in Vehicle License Plate," *International Conference for Convergence in Technology*, Pune, pp. 1-4, 2021.
- [3] N. N. Kamal and E. Tariq, "License Plate Tilt Correction: A Review," *Engineering and Technology Journal*, vol. 39, no. 5, pp. 101-116, Mar. 2021.
- [4] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the Association for Computing Machinery*, vol. 15, no. 1, pp. 11-15, Jan. 1972.
- [5] R. C. Gonzalez and E. W. Richard, *Digital Image Processing*, 3rded. New Jersey, NJ:Prentice Hall, 2008.
- [6] A. Kaehler and G. Bradski, *Learning OpenCV3*, Wikibooks, Paju, 2018.
- [7] J. S. Oh, "An Improved Hough Transform Using Valid Features," *Journal of The Korea Institute of Information and Communication Engineering*, vol. 18, no. 9, pp. 2203-2208, Sep. 2014.
- [8] S. K. Hwang, *Visual C++ Image Processing, Programming*, Gilhut, Seoul, 2015.



오정수(Jeong-su Oh)

중앙대학교 대학원 전자공학과 공학석사
 중앙대학교 첨단영상대학원 영상공학과 공학박사
 현재 부경대학교 융합디스플레이공학과
 ※관심분야: 디지털영상처리, 비디오영상처리,
 적외선 신호처리