

양방향으로 증가하는 허프 변환 : 개선된 디지털 허프 변환 알고리즘

*김수현, 박인철
한국과학기술원 전기 및 전자공학과
e-mail : *shkim.ics@gmail.com*

Bidirectional Incremental Hough Transform : An Improved Algorithm for Digital Hough Transform

*Su-Hyun Kim, In-Cheol Park
Department of Electrical Engineering
Korea Advanced Institute of Science and Technology

Abstract

Hough Transform (HT) has been widely used for object detection in digital images, but it requires long execution time and high computational complexity. Therefore Incremental Hough Transform (IHT) was proposed to reduce requirements of Standard Hough Transform (SHT). IHT reduced execution time and computational complexity by applying parallel, pipeline architectures and approximations. However, IHT has the accumulated errors due to cosine and sine approximation from SHT. It makes that IHT needs to conjugate SHT. In order to reduce the number of SHT conjugation, we propose a Bidirectional Incremental Hough Transform(BIHT).

I. 서론

이미지 분석 기술의 발달로 인하여 스마트 폰, 차량 로봇 등과 같은 다양한 응용분야에서 이미지 내의 물체 인식 및 감지하는 기술이 널리 사용되고 있다. 이러한 이미지 분석 기술 중 하나인 허프 변환은 이미지

내의 선, 원, 타원 뿐만 아니라 임의의 형태를 지닌 물체를 감지해 내는 대표적인 기술로 잘 알려져 있다. 허프 변환은 직교좌표계로 표현되는 이미지의 에지 점들을 극좌표계로 옮겨 검출하고자 하는 물체의 파라미터를 추출하는 방법으로 상대적으로 이미지의 잡음으로부터 적은 영향을 받으며, 견고하다는 장점을 지니고 있다. 직선을 검출하고자 하는 경우에 사용되는 직교 좌표계와 극좌표계에서의 파라미터 관계는 다음과 같다.

$$= x \cos \theta + y \sin \theta \quad (1)$$

여기서 x 와 y 는 이미지 내에 존재하는 에지 점들의 위치이며, ρ 는 원점으로부터 찾고자 하는 직선까지의 수선의 길이, θ 는 x 축과 수선사이의 각도이다.

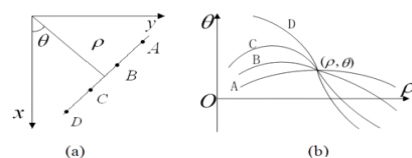


그림 1. 직선 검출을 위한 허프 변환

본 논문에서는 허프 변환을 이용한 선 검출 디지털

하드웨어 제작을 위해 기존에 제시되었던 단방향 증가형 허프변환 (IHT) 알고리즘 [2]을 개선한 양방향 증가형 허프변환 (BIHT) 알고리즘을 제안한다.

II. 본론

2.1 증가형 허프변환

증가형 허프변환 알고리즘은 Fast Incremental Hough Transform (FIHT2) [1]에서 처음 제안 되었으며, 그 정의는 아래의 식으로 표현된다.

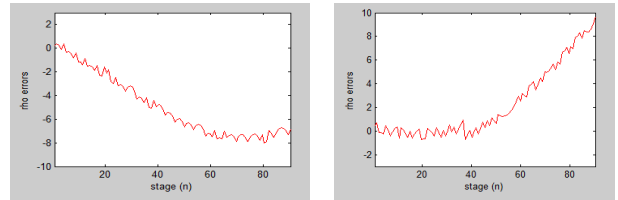
$$\begin{aligned} \rho_{n+1} &= \rho_n + \epsilon \rho'_n, & 0 \leq n < \frac{K}{2} - 1 \\ \rho'_{n+1} &= \rho'_n - \epsilon \rho_{n+1}, & \frac{K}{2} \leq n < K \\ \epsilon &= \frac{\pi}{K}, & \end{aligned} \quad (2)$$

여기서 n , ϵ 그리고 K 는 각각 색인, 해상도 그리고 디지털화를 위한 세타 축을 분할 개수를 의미한다. FIHT2는 삼각함수와 곱셈을 사용하지 않고 SHT를 완벽하게 대체하며, 정상적으로 선을 검출할 수 있다는 것은 [1]에서 입증되어있다. 하지만 FIHT2는 미래 값 ρ'_{n+1} 을 구하기 위해서 미래 값 ρ_{n+1} 과 과거 값 ρ'_n 을 필요로 한다. 따라서 허프변환의 계산 속도 문제를 해결하지 못한다. 이 점을 개선 하기 위해 [2]는 FIHT2 와는 달리 이전 값으로만 미래 값을 계산하는 IHT을 제안하였으며, 그 정의는 아래의 식으로 표현된다.

$$\begin{aligned} \rho_{n+1} &= \rho_n + \epsilon \rho_{(K/2)+n}, & 0 \leq n < \frac{K}{2}, \\ \rho_{(K/2)+n+1} &= \rho_{(K/2)+n} - \epsilon \rho_n, \\ \epsilon &= \frac{\pi}{K}, \rho_0 = x, \rho_{(K/2)} = y & \end{aligned} \quad (3)$$

식 (3) 으로부터 매 n 번째 단계에서 동시에 두 개의 ρ 값을 구 할 수 있다 (이 두 ρ 값에 대한 θ 의 위상은 $\frac{\pi}{2}$ 만큼 차이가 난다). 따라서 IHT에서의 계산시간은 SHT에서의 계산시간보다 상당히 감소시켰다. 하지만 식(1)로부터 식 (2), (3)을 유도하는 과정에서 곱셈과 삼각함수를 제거하기 위해 cosine 과 tangent 를 1과 2^o으로 근사화하게 됨으로써 SHT와 FIHT2 그리고 SHT와 IHT사이의 미세한 오차를 유발한다. 이 미세한 오차는 계산 단계가 증가할수록 누적되며, 누적된 오차는 고해상도의 이미지를 처리할 경우 그림.2

와 같이 증가하여 결과에 심각한 영향을 미치는 원인이 된다.



(a) ρ_{n+1} (b) $\rho_{(K/2)+n+1}$

그림 2. 누적되는 ρ 값의 오차

2.2 양방향으로 증가하는 허프변환

증가형 허프변환 알고리즘에서 사용된 근사화로부터 발생하는 ρ 값의 오차는 점차 누적된다. 이를 해결하기 위한 간단한 방법은 그림 3과 같이 ρ 값의 허용오차 기준을 정하여 해당 기준을 넘지 않을 때까지 증가형 허프변환 알고리즘을 사용하며, 만약 기준을 넘을시 SHT으로부터 정확한 ρ 값을 계산하여 해당 단계의 입력 값을 대체하는 방법이다.

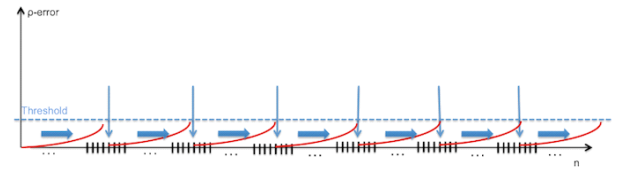


그림 3. 허용오차를 사용한 증가하는 허프변환

하지만 SHT으로 ρ 값을 구하기 위해서는 식 (1)과 같이 해당 단계에서의 삼각함수 값과 곱셈을 필요로 하기 때문에 SHT를 대체하는 횟수가 빈번 할 수 록 하드웨어의 면적과 전력소모는 커진다. 이를 완화하기 위하여 본 논문에서는 그림 4와 같은 양방향으로 증가하는 허프변환을 제안한다.

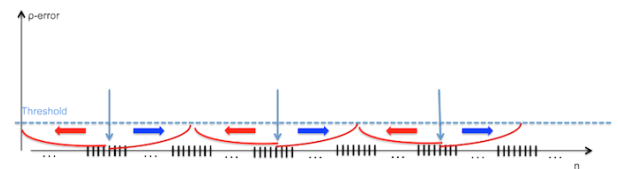


그림 4. 양방향으로 증가하는 허프변환

III. 구현

시스템을 구성하기 위하여 MATLAB과, ncverilog tool을 사용하였다. 우선 이미지 파일을 MATLAB으로

읽어와 grayscale의 raw.hex 파일을 생성하였으며, 이를 ncverilog tool에서 verilog 하드웨어 코드로 작성된 canny 에지 검출기와 양방향 증가형 허프변환의 입력으로 사용하였다. 입력은 canny 에지 검출기를 통하여 에지 정보만을 0과 1로 표현한 이진이미지로 변환되고, 이진 이미지는 허프변환 하드웨어의 입력으로 사용되어 최종적으로 허프변환의 결과로 강인한 직선에 대한 (ρ, θ) 쌍들을 반환하며, 이 결과를 토대로 MATLAB에서 이미지 위에 해당 (ρ, θ) 쌍들을 통한 직선을 그어 실제 이미지에서의 직선과 그어진 직선을 비교하여 결과를 확인 할 수 있도록 하였다.



그림 5. 구현된 양방향 증가형 허프변환 결과

IV. 결론 및 향후 연구 방향

애플리케이션에 특화된 임베디드 하드웨어 시스템을 만들기 위해서는 적은 면적과 간단한 연산을 사용하는 알고리즘을 개발 및 적용 하는 것이 바람직한 방향이다. 이를 위해 본 논문에서는 증가형 허프변환을 개선한 양방향으로 증가하는 허프변환을 제안하였다. 이를 통하여 적은 양의 연산을 추가하여 기존의 증가형 허프변환으로 부터 발생하는 누적 ρ 오류를 극복하였다. 하지만 누적 ρ 오류는 이미지의 해상도가 증가함에 따라 누적되는 정도가 급격하게 변화한다. 실험에 사용된 이미지의 해상도는 1024x768 이었으나 발달하는 영상처리 기술로부터 향상되는 이미지 해상도에 대처 할 수 있도록 누적 ρ 오류의 근본적인 문제를 해결해야 할 것이다.

Acknowledgement

이 논문은 2014년도 정부(미래창조과학부)의 재원으로

한국연구재단의 지원을 받아 수행된 연구임(No. 2010-0028680).

참고문헌

- [1] en.wikipedia.org/wiki/Hough_transform
- [2] Koshimizu H, Numada M. FIHT2 algorithm: a fast incremental Hough transform. IEICE Transactions 1991.
- [3] Tagzout S, Achour K, Djekoune O. Hough transform for FPGA implementation. Elsevier Journal, Signal Processing 2001.
- [4] Ahmed, E. Medhat, M. A Memory Efficient FPGA Implementation of Hough Transform for Line and Circle Detection. In Proceedings of the 25th Canadian Conference on Electrical and Computer Engineering, Montreal, QC, Canada, 2012.
- [5] Shang, E.K. Li, J. An, X.J. Fast hough transform for FPGA-based applications. Comput. Eng. 2010.
- [6] Lu, X. Song, L. Shen, S. He, K. Yu, S. Ling, N. Parallel Hough Transform-based straight line detection and its FPGA implementation in embedded vision. Sensors 2013.