

국소영역 내의 CCT법을 이용한 고정밀 직선 검출

A High Precision Line Detection Based on Local Area CCT Method

정 남 채*
Nam-chae Jung*

요 약

본 논문에서는 화상에 존재하는 디지털 직선을 고정밀도로 검출하는 방법을 제안한다. 화상의 크기를 $N \times N$ 로 하면, 이 계산량은 $O(N^4)$ 이지만 실제 사용하기는 곤란하므로, 검출 정밀도의 열화를 억제하면서 계산량을 $O(N^3)$ 로 하는 알고리즘을 검토하였다. 국소영역에서 Hough 변환하여 추출된 선분을 연신처리(stretching treatment)하고, 화상으로부터 직선을 검출하는 방법은 길거나 짧은 여러 가지의 직선을 고속으로 검출할 수 있는 훌륭한 방법이지만, 기울어진 선분의 검출 정밀도는 약간 떨어진다. 본 논문에서는 사선의 검출 정밀도를 향상시킨 직선 검출방법을 국소영역에 적용함으로써 처리속도가 감소되지 않고, 직선을 고정밀도로 검출하는 방법에 관해서 논술한다. 실험 결과 제안된 방법은 기존의 방법과 같은 정도 이하의 시간에서 정밀도가 높은 직선을 검출할 수 있다는 것을 확인하였다.

Abstract

A detection method of high precision digital line within image is proposed in this paper. If we set the size of image to $N \times N$, in fact it is difficult to use the resulting values that the amount of computation is $O(N^4)$. Multiple algorithms are examined to reduced the amount of computation to $O(N^3)$, while suppressing the degradation of precision. How to detect line from the image processing, after stretching treatment of line segments extracted by Hough transform in the local area of an image is a great way to be able to detect several long or short line at high speed, but this method is slightly less precision in the detection of tilted line segments. In this paper, a line detection method improving the precision detection of tilted line segment is applied to the local area, thereby this method does not reduce the processing speed, while it is high precision method for detecting line segments. The experimental results confirm that the proposed method can detect a high precision line in a shorter period of time, compared with the existing methods.

Keyword : Digital line, Farey sequence, Hough transformation, Line detection Image processing

I 서 론

디지털 화상 위의 직선 검출법으로는 일반적으로 Hough 변환이 이용되고 있다[1]. Hough 변환은 잡음이나 은폐된 것에 강하고 안정되어 직선을 검출하기 쉽다는 특징이 있지만, 다음과 같은 문제점이 지적되고 있다[2].

- (1) 투표공간(voting space)을 표시하기 위한 큰 메모리 공간이 필요하다.
- (2) 화상 안의 각 점마다 1개의 Hough 곡선을 그리기 때문에 계산량이 많아진다.
- (3) 투표도수(voting frequency)의 적고 많음에 따라 선분인지 아닌지를 평가하기 때문에 짧은 직선 검출은 어렵다.
- (4) 투표 공간 안의 셀과 화상 공간 안의 직선은 1대 1로 대응되지

않기 때문에, 직선을 구성하는 화소수 중의 투표도수가 대응되는 셀에 정확하게 축적되지 않는다.

문제점 (1), (2) 등의 계산량 및 메모리의 효율화를 위하여 확률적 알고리즘의 도입, 병렬화, 국소적인 특징을 이용하는 방법, 계층적으로 투표하는 방법, 투표공간의 분해, 궤적을 고속으로 그리는 것 등의 방법이 제안되고 있다[3,4].

본 논문에서는 문제점 (3)을 해결하기 위하여 국소영역에 Hough 변환을 이용한 직선 검출법을 제안하였다[5]. 이 방법은 화상을 작은 영역으로 분할하고, 각각의 작은 영역에 대하여 Hough 변환을 적용하여 선분후보를 검출한다. 그리고 선소(Line segment)가 국소영역 외에서 늘어날 가능성이 있는 경우에는 연신처리를 하여 선분을 탐색하는 방법이다. 작은 화상영역에 Hough 변환을 적용하기 때문에, 계산량과 메모리를 표준의 Hough 변환의 수분의 1에서 10분의 1 정도까지 감소시킬 수 있다. 또한, 작은 화상 영역을 대상으로 하기 때문에 짧은 선분도 검출할 수 있다. 그러나 Hough 변환을 사용하고 있기 때문에 문제점 (4)는 여전히 해결되지 않는다. 문제점 (4)에 대해서는 직선을

* 초당대학교 정보통신공학과

투고 일자 : 2013. 2. 25 수정완료일자 : 2013. 4. 13

계재확정일자 : 2013. 4. 30

검출하는 정밀도의 균일화를 나타내는 관점에서 Hough 변환의 구조가 의논되어 양자화 오차를 고려하여 투표한 방법이나 Hough 변환의 비선형을 고려하여 파라미터 공간을 양자화한 방법이 제안되고 있다[6-8]. 또한, 크기 $N \times N$ 의 화상에서 구해진 디지털 직선의 수는 $O(N^4)$ 라고 표시되고 있다[9]. 그러므로 디지털 직선을 아주 정밀하게 분리하려면 투표 공간($\rho-\theta$ 공간)의 ρ 와 θ 를 각각 $O(N^2)$ 정도로 양자화할 수 있어야 한다. 이에 반해 보통의 Hough 변환은 메모리 공간 및 계산시간을 감소시키기 위해, ρ 와 θ 를 각각 $O(N)$ 로 양자화하고 있다. θ 에 관한 성기계(roughly) 양자화하는 것은 직선이 증가하게 되므로 정확한 직선과의 차이가 크게 되어 직선을 검출하는 정밀도가 떨어진다.

이에 반하여 본 논문에서는 디지털 화상 위의 양자화된 직선(디지털 직선)의 구조를 조사하여, 투표도수에 편향을 일으키지 않고, 고정밀 직선을 검출하는 CCT(Chain Code Transform) 법을 적용하였다[10]. 그러나 이 방법은 디지털 평면 위의 모든 디지털 직선을 세는 방법이기 때문에 계산량, 메모리 공간 모두가 기존의 Hough 변환보다 비용이 높아서 직접 화상 전체에 적용하기는 어렵다.

그러므로, 본 논문에서는 화상을 작은 영역으로 분할하여 고속으로 직선을 검출하는 방법에 Hough 변환 대신에 CCT 법을 적용함으로써 고속으로 짧은 선분도 검출할 수 있다는 특징을 유지한 채로 고정밀도의 직선을 검출하는 방법을 제안한다. 그리고 투표 실험, 실화상을 이용한 직선 검출 실험, 그리고 국소영역에 Hough 변환을 이용하여 직선 검출법과의 비교실험에 의하여 이 방법의 유효성을 확인한다.

다음 II장에서는 디지털 직선을 정의하고, 디지털 직선이 갖는 기본적인 성질을 나타낸다. III장에서는 직선검출 알고리즘을 제안한다. IV장의 실험 및 검토에서 이 방법의 유효성을 확인한다.

II. 디지털 직선

본 논문에서는 유리계수직선 식 (1)을 검토 대상으로 한다.

$$y = \frac{m}{n}x + \frac{l}{k} \tag{1}$$

여기서, m, n 은 서로 소이고, l, k 도 서로 소로 한다. 또한, 일 반성을 잃지 않고, $0 \leq \frac{m}{n} \leq 1$ 이면 된다. $1 \leq \frac{m}{n}$ 의 경우는

$x = \frac{n}{m}y + \frac{l'}{k'}$ 을 고찰하며, 음의 경우는 $x = -x$ 로 하여 고찰한다.

이 직선의 y 값을 연속으로 하는 정수의 점 $i, i+1, \dots, j$ 로 식 (2)와 같이 정수화한다.

$$\left\lfloor \frac{m}{n}i + \frac{l}{k} \right\rfloor, \left\lfloor \frac{m}{n}(i+1) + \frac{l}{k} \right\rfloor, \dots, \left\lfloor \frac{m}{n}j + \frac{l}{k} \right\rfloor$$

그리고 집합

$$\left(i, \left\lfloor \frac{m}{n}i + \frac{l}{k} \right\rfloor \right), \left(i+1, \left\lfloor \frac{m}{n}(i+1) + \frac{l}{k} \right\rfloor \right),$$

$$\dots, \left(j, \left\lfloor \frac{m}{n}j + \frac{l}{k} \right\rfloor \right) \tag{2}$$

을 구간 $[i, j]$ 위의 디지털 직선으로 정의한다.

크기 $N \times N$ 의 화상에서 각 좌표들 (i, j) , $1 \leq j \leq N$ 으로 나타낸다. 화상 위의 점집합에서 식 (2)를 만족한 것을 화상 구간 $[i, j]$ 에 대한 디지털 직선이다.

디지털 직선은 기울기와 절편의 이산화에 의하여 구해지므로 직선 검출을 위하여 다른 디지털 직선으로 구별할 수 있는 기울기의 분해능과 절편의 분해능에 의하여 구할 수 있다.

우선, 절편의 이산화는 절편을 정수부와 소수부로 분리하여 나누어 고찰한다. 식 (1)에서 $\frac{l}{k} = p + \frac{l'}{k}$ (여기서 $l' < k$)라 하면 다음과 같이 표시된다[10].

[성질 1] 직선 $y = \frac{m}{n}x + p + \frac{l'}{k}$ 에서 $\frac{q}{n} \leq \frac{l'}{k} < \frac{q+1}{n}$ (q 에 대하여 다시 정리하면, $\frac{l'}{k}n - 1 < q < \frac{l'}{k}n$)이면, 직선 $y = \frac{m}{n}x + p + \frac{l'}{k}$ 와 $y = \frac{m}{n}x + p + \frac{q}{n}$ 가 표시하는 디지털 직선은 동일하다.

따라서, 절편의 이산화는 $\frac{1}{n}$ 의 단위로 하는 것이 좋다. 기울기 분해능은 Farey 수열을 이용하여 결정할 수 있다. Farey 수열은 다음과 같이 정의된다[11].

[정의 1] (Farey 수열) 자연수 n 을 1개 취한다. 분모와 분자 모두에 n 을 초과하지 않는 자연수에 의하여 구할 수 있는 기약분수를 0부터 올림차순으로 정리한다.

이것을 n 에 속한 Farey 수열이라고 한다.

[예 1] $n=5$ 에 속한 Farey 수열은 0부터 1의 범위에서

$$\frac{0}{1}, \frac{1}{5}, \frac{1}{4}, \frac{1}{3}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{1}{1}$$

로 된다.

본 논문에서는 기울기가 0에서 1까지의 직선을 고찰하므로, Farey 수열도 0에서 1의 범위까지만 고찰하기로 한다.

크기 $N \times N$ 의 화상이 주어진 경우, 화상 위의 디지털 직선의 기울기는 N 에 속하는 Farey 수열의 분수로 이산화되는 것이 좋다. 즉,

[성질 2] N 에 속하는 Farey 수열의 서로 이웃하는 두 개의 분수를 $\frac{m_1}{n_1}, \frac{m_2}{n_2}$ 라 할 때, 그 사이의 분수 $\frac{m}{n}$ (여기서 $\frac{m_1}{n_1} < \frac{m}{n} <$

$\frac{m_2}{n_2}$)을 기울기로 하는 직선($y = \frac{m}{n}x + \frac{l}{k}$)의 디지털 직선은 화

상의 범위 내에서는 $y = \frac{m_1}{n_1}x + \frac{l}{k}$ 또는 $y = \frac{m_2}{n_2}x + \frac{l}{k}$ 의 디지털

직선과 일치하여 구별할 수 없게 된다.

이것으로부터 크기 $N \times N$ 의 화상에 존재하는 디지털 직선은 좌표축으로 Farey 수열을 나타내는 축, 정수 절편을 나타내는 축, 소수 절편을 나타내는 축으로부터 파라미터 공간의 크기는 Farey 수

열의 항 수가 $O(N^2)$ 이고, 정수 절편 및 소수 절편은 각각 $O(N)$ 으로 이산화되므로 $O(N^4)$ 가 된다.

CCT법은 파라미터 공간의 투표에 근거한 직선검출법이다[10].

III. 직선 검출법

이 장에서는 앞 장에서 논술한 직선검출 알고리즘에 관해서 논술한다.

3.1 CCT 알고리즘

우선, CCT법에서 파라미터 공간의 투표 방법에 관해서 논술한다. 화상의 크기는 $N \times N$ 로 하고, 파라미터 공간은 Farey 수열을 나타내는 축을 F 축, 정수 절편을 나타내는 축을 p 축, 소수 절편을 나타내는 축을 q 축으로 표시한다.

이 공간을 (F, p, q) 또는 $(\frac{a}{b}, p, q)$ 와 같이 표기한다. 이 파라미터 공간은 화상 위에 나타난 식 (2)에서 정의된 디지털 직선을 완전히 분리한다. 따라서 가장 정밀한 알고리즘은 이러한 파라미터를 축으로 하는 파라미터 공간의 투표이다.

아래의 알고리즘은 다음 사실에 근거하여 구성된다.

(1) $\frac{m}{n} \leq \frac{a}{b} < \frac{m+1}{n}$ 이 되는 기울기 $\frac{a}{b}$ 를 갖는 직선 $y = \frac{a}{b}x$ 의 디지털 직선은 화상 평면 위의 점 (n, m) 을 지난다. 따라서 N 에 속하는 Farey 수열의 분수 $\frac{a}{b}$ 에서 $\frac{m}{n} \leq \frac{a}{b} < \frac{m+1}{n}$ 의 범위에 있는 기울기를 갖는 것은 점 (n, m) 을 지난다.

(2) 정수 절편 p 를 갖는 직선 $y = \frac{a}{b}x + p$ 의 디지털 직선은 $\frac{m-p}{n} \leq \frac{a}{b} < \frac{m+1-p}{n}$ 일 때, 점 (n, m) 을 지난다. 단, 기울기가 0과 1 사이에 있다는 전제이므로, 구하는 p 의 범위는 $m-n < p \leq m$ 이 된다.

(3) 직선 $y = \frac{a}{b}x + p - 1 + \frac{q}{b}$, ($1 \leq q < b$)의 디지털 직선은 $m - (p-1) \leq \frac{an+q}{b} < m+1 - (p-1)$ 일 때 점 (n, m) 을 지나고, 직선 $y = \frac{a}{b}x + p + \frac{q}{b}$, ($1 \leq q < b$)의 디지털 직선은 $m - p \leq \frac{an+q}{b} < m+1 - p$ 일 때 점 (n, m) 을 지난다. 이상에 의하여 다음의 CCT 알고리즘이 유도된다.

[완전 CCT 알고리즘]

입력 화상의 크기를 $N \times N$ 로 하고, 기울기 $\frac{m}{n}$ 가 $0 \leq \frac{m}{n} < 1$ 의 범위일 때, 다음과 같은 순서로 처리한다. 그 외의 기울기에 대해서는 좌표축을 변환하여 같은 순서를 적용하며, 투표 공간은 4개로 나눌 수 있다.

[Step 1] 화상 공간의 점 (n, m) 이 선분 후보점이라면 $\frac{m-p}{n} \leq$

$\frac{a}{b} < \frac{m+1-p}{n}$, $m-n < p \leq m$, $q=0$ 가 되는 파라미터 공간 위의 각 점 $(\frac{a}{b}, p, 0)$ 에 투표한다.

[Step 2] $1 \leq q < b$ 범위의 각 정수 q 에 대하여, $m - (p-1) \leq \frac{an+q}{b}$ 인 파라미터 공간 위의 점 $(\frac{a}{b}, p-1, q)$ 에 투표함과 동시에 $\frac{an+q}{b} < m+1-p$ 인 동일 공간 위의 점 $(\frac{a}{b}, p, q)$ 에 투표한다.

[Step 3] 화상공간 위의 각 선분 후보 점에 대하여 Step 1, 2를 반복한다.

완전 CCT 알고리즘은 계산량, 메모리 공간 모두 $O(N^4)$ 가 되므로 간략 알고리즘을 고찰한다. 소수절편은 $\frac{q}{n}$ 의 평행이동을 나타내고 있기 때문에 직선을 분리하는 데에는 거의 도움이 되지 못한다. 그러므로 소수절편 q 축을 삭제하고 $F-p$ 공간에서 투표한다. 간략 CCT 알고리즘은 완전 CCT 알고리즘의 스텝 2를 삭제한 알고리즘이 된다. 간략 CCT는 계산량과 메모리 공간을 모두 $O(N^3)$ 로 감소시킬 수 있다.

간략 CCT를 이용하여도 $\frac{q}{n}$ 는 평행이동의 항이므로 화상의 기울기에 관한 검출 정밀도에는 영향을 미치지 않는다. 그러므로 각도 분해능을 떨어뜨리지 않고 직선을 검출하고, 계산량과 메모리 공간을 감소시킬 수 있다.

3.2 CCT 알고리즘에 대한 좌표축의 변환

기울기 $\frac{m}{n}$ 이 $0 \leq \frac{m}{n} < 1$ 의 범위를 벗어난 경우의 좌표축 변환 방법을 다음에 나타낸다(그림 1). 투표공간을 첨자 0~3을 붙여 표시한다.

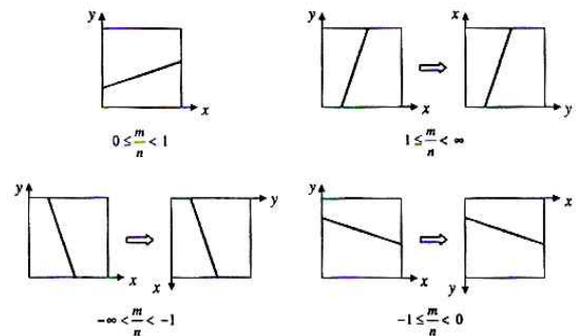


그림 1. 좌표축의 치환방법
Fig. 1. Conversion of axes.

• $0 \leq \frac{m}{n} < 1$ 일 때 (투표공간 0)

화상공간의 점 (n, m) 이 선분 후보점이면 $\frac{m-p}{n} \leq \frac{a}{b} <$

$\frac{m+1-p}{n}$, $m-n < p \leq m$ 이 되는 투표공간 0 위의 각 점에 투표한다.

- $1 \leq \frac{m}{n} < \infty$ 일 때 (투표공간 1)

직선의 기울기가 0~1의 범위에 존재하도록 좌표축을 치환하면, $0 \leq \frac{a}{b} \leq 1$ 의 Farey 수열을 사용할 수 있다. 이를 위하여 좌표 (n, m) 을 (m, n) 으로 치환하여 $\frac{n-p}{m} \leq \frac{a}{b} < \frac{n+1-p}{m}$, $n-m < p \leq n$ 이 되는 투표공간 1 위의 각점에 투표한다.

- $-\infty < \frac{m}{n} < -1$ 일 때 (투표공간 2)

직선의 기울기가 정확히 0~1의 범위에 존재하도록 좌표 (n, m) 을 $(y_size-1-m, n)$ 으로 치환하여 $\frac{n-p}{y_size-1-m} \leq \frac{a}{b} < \frac{n+1-p}{y_size-1-m}$, $m+n+1-y_size < p \leq n$ 이 되는 투표공간 2 위의 각점에 투표한다. 여기서, y_size 는 화상의 y 방향의 크기라고 할 수 있다.

- $-1 < \frac{m}{n} < 0$ 일 때 (투표공간 3)

위와 같은 방법으로, 좌표 (n, m) 을 $(n, y_size-1-m)$ 으로 치환하여 $\frac{y_size-1-m-p}{n} \leq \frac{a}{b} < \frac{y_size-1-m+1-p}{n}$, $y_size-1-m-n < p \leq y_size-1-m$ 이 되는 투표공간 3 위의 각 점에 투표한다.

3.3 특징과 문제점

CCT에는 다음과 같은 특징과 문제점이 있다.

[특징]

- 투표공간에 편향이 일어나지 않는다.
- 완전 CCT를 이용하면 모든 디지털 직선을 완전히 분리할 수 있어 고정밀도의 직선을 검출할 수 있다.
- 4개의 투표공간에 투표를 하기 때문에 투표와 피크 검출부분에서는 병렬화가 가능하다.
- 간략 알고리즘을 이용하는 것으로 충분한 검출 정밀도를 유지하면서 계산량과 메모리 공간을 감소시킬 수 있다.

[문제점]

완전 CCT를 이용하면 계산량과 메모리 공간 모두가 $O(N^4)$ 이 되어 화상전체에 적용하기에는 현실적이지 못하다. 또한, 간략 CCT 알고리즘을 이용하여도 계산량과 메모리 공간 모두가 $O(N^3)$ 이 되어 N 이 크게 되면 실용적이지 못하다.

3.4 국소영역에 간략 CCT를 적용한 직선검출법

국소영역에 Hough 변환을 적용하여 선분후보를 추출하고, 그것을 연신하여 화상으로부터 직선을 검출하는 방법을 제안한다. 이 방법은 화상에 혼재하는 길고 짧은 여러 가지의 직선을 고속으로 추출할 수 있다는 장점을 가지고 있지만, 검출정밀도가 투표공간의

각도분해능으로 제한된다는 단점이 있다. 그러므로 국소영역에서는 선분후보의 검출에 CCT 법을 사용하는 것이 타당하다고 할 수 있다. 다음에는 국소영역 Hough 변환을 적용하는 LHT(Local Hough Transform) 법, 간략 CCT를 적용하는 LCCT(Local Chain Code Transform)법을 검토하기로 한다.

LCCT법의 알고리즘은 다음과 같이 구성된다. 국소영역은 화상의 크기를 $N \times N$ 로 한 경우, $\frac{N}{n}$ 가 정수가 되도록 n 을 선택하여 $n \times n$ 의 크기로 한다. 이 결과 화상은 $\left(\frac{N}{n}\right)^2$ 개의 국소영역으로 나누어진다.

[LCCT 알고리즘]

[Step 1] 국소영역을 1개 선택하여 간략 CCT를 적용한다. 그 결과 선소가 추출된다.

[Step 2] 추출된 각 선소에 대하여 연신처리를 하여 직선을 확정한다.

[Step 3] 각 국소영역에 대하여 Step 1, 2를 반복한다.

위에서 설명한 알고리즘에 관해서 국소영역의 선택법이 가끔 고찰되고 있지만, 본 논문에서는 래스터 주사(raster scan) 순으로 선택하는 방법을 이용하였다.

다음에는 연신처리에 관하여 자세하게 논술한다. 선소는 $F-p$ 투표공간에서의 최대값을 주어진 파라미터로부터 역변환함으로써 생성되지만, 간략 CCT를 이용하기 때문에 거기에는 소수절편 삭제에 수반되는 오차가 발생한다. 이 오차를 고려하여 선소를 연장한다. 그 때 선소의 끊어짐도 고려하여 연신처리한다.

[연신처리 알고리즘]

[Step 1] $F-p$ 투표공간에서 최대값을 주어진 파라미터로 역변환하여 구해진 선소의 방정식으로부터 국소영역내의 선소의 시점과 선소의 중점을 구한다.

[Step 2] 구해진 선소의 방정식에 따라 시점과 중점으로부터 다른 영역에 선소를 늘어간다. 여기서, 소수절편 q 에 관한 오차는 직선을 y 축에 평행 이동한 만큼의 오차이기 때문에 약 ± 1 이다. 그러므로 연신처리할 때의 주목하는 화소의 ± 1 만큼의 화소에 선분 후보점이 존재할 수 있다. 즉, 폭 3화소의 범위에서 선분 후보점이 존재하면 그것을 선소에 더하여 연신처리를 계속한다.

[Step 3] 3화소 내에 선분 후보점이 없는 경우에도, 선분의 gap이 존재할 수 있으므로, gap 길이로 지정된 범위 내라면 연신처리를 계속한다. gap 길이를 초과하여 선분 후보점이 없다면 연신조작을 종료한다.

선소를 연신처리할 때, 선소에 더해진 국소영역 외의 선분 후보점은 label을 붙여준 이후의 처리로 선분 후보점으로부터 제외시켜 같은 선분을 중복하여 검출하는 것을 피한다.

IV. 실험 및 검토

이 방법의 유효성을 확인하기 위해 LHT, LCCT를 이용하여 직선 검출 정밀도와 처리시간에 관한 성능비교 실험을 하였다. 4.1에

서는 국소영역 내에서 적용한 Hough 변환과 간략 CCT의 기본적인 특성을 보기 위하여 전좌소를 선분 후보점으로 하는 좌소를 입력하여 투표공간에서 투표실험을 하고, 투표공간의 구조를 조사하였다.

4.2에서는 LHT와 LCCT에 대하여 샘플 화상을 이용하여 직선을 검출하여 비교실험을 하였다.

4.1 투표실험

전좌소가 선분 후보점으로 좌상 크기가 16×16 의 2차 좌상을 입력하여 Hough 변환과 간략 CCT에 대하여 투표공간의 투표수의 분포를 구하였다. 간략 CCT에 대한 투표 분포를 그림 2, 3에 나타낸다. CCT는 3.2에서 표시된 4개의 투표공간으로 나눌 수 있지만, 그림 2는 투표공간 0과 투표공간 3을 합하여 표시하였고, 그림 3은 투표공간 1과 투표공간 2를 합하여 표시하였다.

또한, 그림 4에 Hough 변환에 대한 투표의 분포를 나타낸다. 투표공간의 분할 수는 문헌 [12]에 근거하여 정하였다.

크기 16×16 의 좌상에는 가장 긴 직선의 길이는 16부터 투표공간의 각 점의 투표수의 최대치는 16이다. 그림에서 $x-y$ 면에 투표공간의 파라미터 축을 표시하여 z 축 방향으로 각 점에 투표된 투표수를 나타내고 있다. 그림에서 알 수 있는 것처럼 간략 CCT는 투표수 16이 되는 점이 마음모꼴 영역을 하고 있다.

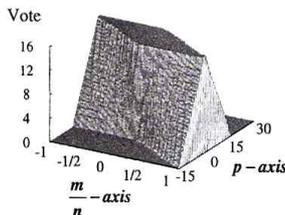


그림 2. CCT($y = ax + b$)
Fig. 2. CCT($y = ax + b$)

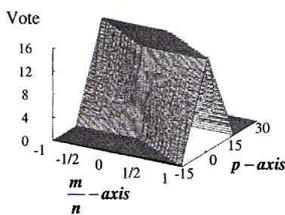


그림 3. CCT($x = ay + b$)
Fig. 3. CCT($x = ay + b$)

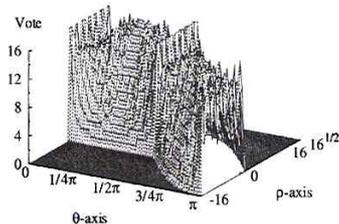


그림 4. Hough 변환
Fig. 4. Hough transform

이것은 직선이 고정밀도로 분리되어 있는 것을 나타내고 있으며,

Hough 변환은 투표수의 분포가 불균일하여 피크 점 근처에서 투표수가 적게 되는 현상을 볼 수 있다. 이 실험에서는 전좌소가 선분 후보점이므로 CCT와 같은 투표결과가 기대되지만, 그렇게 되지는 않았다. 이것은 Hough 변환의 직선검출 능력이 다소 떨어지기 때문이라고 할 수 있다.

4.2 직선검출 실험

LCCT 법의 유효성을 확인하기 위해 실좌상을 이용하여 LHT 법과 비교실험을 하였다. 2가지 방법을 직선 정밀도, 처리시간의 관점에서 비교 검토하였다.

4.2.1 실험 방법

실험에서는 잡음을 포함하지 않은 다른 직선 11개(X 축, Y 축 위에도 직선이 있다)가 그려져 있는 입력좌상 1(그림 5: 64×64)와 잡음을 포함한 직선의 길이와 기울기가 다른 직선이 그려져 있는 입력좌상 2(그림 6: 256×256)를 이용하였다. 입력좌상 1은 기본 성능을 검토하기 위하여 잡음을 인가하지 않았지만, 손상된 좌상의 검사에서는 구하는 좌상이 잡음을 포함하고 있기 때문에 그 점을 고려하여 입력좌상 2에 잡음을 인가하였다. 실험에서 사용된 프로그램은 고립점 제거를 전 처리하여 포함하고 있다.

투표공간의 분할 수는 LHT에 대해서는 문헌 [12]에 근거하였고, LCCT에 대해서는 앞 장에서 논술한 결과에 근거하여 정하였다. 알고리즘의 제어 파라미터에는 국소영역의 크기 외에 투표수의 피크 검출의 임계값, 선소 길이의 임계값, 선분 길이의 임계값 등이 있다.

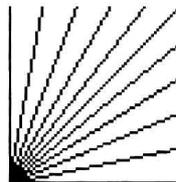


그림 5. 입력좌상 1
($N_G = 64$)
Fig. 5. Input image 1
($N_G = 64$)

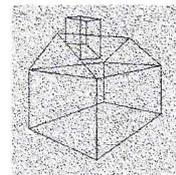


그림 6. 입력좌상 2
($N_G = 256$)
Fig. 6. Input image 2
($N_G = 256$)

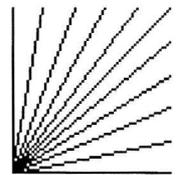


그림 7. LCCT
($N_L = 16$)
Fig. 7. LCCT
($N_L = 16$)

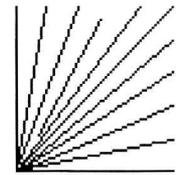


그림 8. LHT
($N_L = 16$)
Fig. 8. LHT
($N_L = 16$)

국소영역의 크기 이외의 파라미터에 대해서는 출력좌상이 외관상 가장 좋아지도록 LCCT($N_L = 16, 32, 64$), LHT($N_L = 16, 32, 64$)로 설정하였다. 구해진 출력좌상을 그림 7~그림 14에 나타낸다.

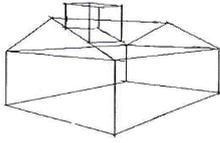


그림 9 LCCT($N_L = 16$)
Fig. 9. LCCT($N_L = 16$)

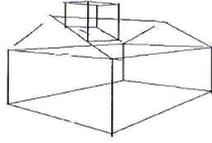


그림 10 LHT($N_L = 16$)
Fig. 10. LHT($N_L = 16$)

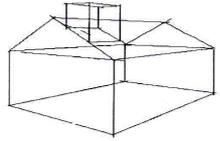


그림 11. LCCT($N_L = 32$)
Fig. 11. LCCT($N_L = 32$)

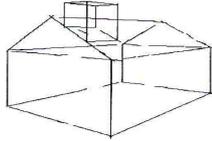


그림 12. LHT($N_L = 32$)
Fig. 12. LHT($N_L = 32$)

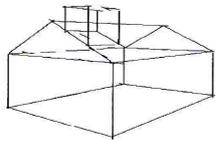


그림 13. LCCT($N_L = 64$)
Fig. 13. LCCT($N_L = 64$)

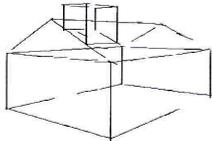


그림 14. LHT($N_L = 64$)
Fig. 14. LHT($N_L = 64$)

표 1. 국소영역 내의 CCT법(LCCT)
Table 1. Chain code transform in local area(LCCT).

입력화상	출력화상	일치도(%)	검출개수	실측시간[s]
그림 5	그림 7	87.09	11	0.0060
그림 6	그림 9	-	50	0.0868
	그림 11	-	38	0.3670
	그림 13	-	44	1.8380

표 2. 국소영역 내의 Hough 변환법(LHT)
Table 2. Hough transform in local area(LHT)

입력화상	출력화상	일치도(%)	검출개수	실측시간[s]
그림 5	그림 8	83.53	11	0.0091
그림 6	그림 10	-	51	0.1368
	그림 12	-	64	0.2363
	그림 14	-	69	0.5810

이러한 출력화상에 대해서 비교평가를 하였다. 비교 항목은 입력 화상과 출력화상 간의 직선의 일치도, 직선 검출 개수, 실행시간이다. 일치도는 식 (3)과 같으며, 결과를 표 1, 표 2에 나타낸다.

$$\text{일치도} = \frac{\text{출력화상의 일치한 점의 수}}{\text{입력화상의 선분후보점의 총 수}} \times 100(\%) \quad (3)$$

4.2.2 실험 결과와 검토

[실험 결과의 검토]

입력화상 1에 대한 직선 검출 처리에는 국소영역의 화상 크기를 2 가지 방법 모두 16×16 으로 하였다. LCCT의 출력 화상(그림 7)은 대개 입력화상과 거의 같은 곳에서 직선을 검출할 수 있었다. 그러나 LHT(그림 8)는 11개 중에서 10개는 정밀도가 양호한 직

선을 검출할 수 있었지만, 나머지 1개의 직선은 선분의 중간까지밖에 검출되지 않았다. 또한, 표 1, 표 2에서 일치도, 처리시간 모두 LCCT 법이 양호하였다.

다음에 입력화상 2에 대하여 국소영역의 크기를 16×16 , 32×32 , 64×64 로 바꿔 실험을 하였다. 결과를 표 1, 표 2, 및 그림 9 ~ 그림 14에 나타내었다. 그림 9 ~ 그림 14에 나타낸 것처럼, 국소영역의 크기를 같은 크기로 설정하였을 때의 2 가지 방법의 출력 결과는 LCCT의 출력화상에는 정밀도가 양호한 직선을 검출할 수 있었지만, LHT에서는 직선의 gap이 많이 발생했다. 이것도 입력화상 1과 같은 이유로 LHT에서는 국소영역 내에서 검출된 선소의 기울기 오차가 LCCT에 비교하여 크게 되는데, 직선의 중간까지밖에 검출할 수 없기 때문이다.

[계산시간의 분석]

2가지 방법의 계산시간을 분석하기 위하여 처리를 [투표]와 [도수 피크의 검출과 연신처리]로 나누어서 검토한다. 주된 계측결과를 표 3, 표 4에 나타낸다.

국소영역의 크기를 $N_L \times N_L$ 로 한다. [투표]에서의 선분 후보 점 1점당의 계산시간을 평가하면 식 (4)와 같이 된다.

LHT: Hough 변환식을 식 (4)로 하였을 때,

$$\rho = x \cos\theta + y \sin\theta \quad (4)$$

θ 의 분할 수만큼 식 (4)를 계산하고, (ρ, θ) 공간에 투표한다. θ 의 분할 수는 $2\pi N_L$ 이므로, $2\pi N_L$ 회의 식 (3)의 계산과 같은 회수의 투표가 계산시간이 된다.

표 3. 입력화상 2에 대한 LCCT의 실험결과(ms)

Table 3. Experimental result of LCCT for input image 2
(Unit of time: ms)

N_L		16	32	64
총실행 시간		86.8	367	1838
투표	시간	10.4	48.3	537
	선분후보점의 수 ⁽¹⁾	4425	4502	5006
투표공간의 크기		(73, 32)×4	(309, 64)×4	(1229, 128)×4
도수피크의	시간	70.4	276	1072
검출과	피크 수	140	237	2019
연신처리	검출 개수	50	38	44

* CPU : Pentium 2.7GHz

(1) 선분 후보 점의 총수는 5481이지만, 연신처리에 관한 label을 붙인 조작에 의해 선분 후보점이 감소된다.

표 4. 입력화상 2에 대한 LHT의 실험결과 (시간단위: ms)

Table 4. Experimental result of LHT for input image 2.
(Unit of time: ms)

N_L		16	32	64
총실행 시간		137	236	581
시간		33.6	80.1	174
투표	선분후보점의 수 ⁽¹⁾	4418	4843	5002
	투표공간의 크기	(101, 55)	(202, 110)	(403, 219)
도수피크의	시간	101	156	401
검출과	피크 수	971	3903	7852
연신처리	검출 개수	51	64	69

* CPU : Pentium 2.7GHz

(1) 선분 후보 점의 총수는 5481이지만, 연신처리에 관한 label을 붙인 조작에 의해 선분 후보점이 감소된다.

LCCT: 간략 CCT 알고리즘에서는 소수절편 축에 투표하지 않기 때문에 완전 CCT 알고리즘의 Step 1의 계산 회수를 평가하면 된다. 점(n, m)을 선분 후보점이라고 하면 $p = m, m-1, \dots, m-n+1$ 에 대하여 $\frac{m-p}{n} \leq \frac{a}{b} < \frac{m+1-p}{n}$ 범위의 파라미터 공간의 점($\frac{a}{b}, p$)에 투표하기 때문에, 결국 $0 < \frac{a}{b} < 1$ 범위의 Farey 수열의 항 수와 동일한 회수의 투표를 한다. N_L 에 속하는 Farey 수열의 항 수는 $\frac{3}{\pi^2} N_L^2 + O(N_L \log N_L)$ 로 주어지기 때문에 [10], 필요한 계산은 n 회의 $\frac{m-p}{n}$ 의 계산과 $\frac{3}{\pi^2} N_L^2 + O(N_L \log N_L)$ 회의 투표이다. 네 가지의 투표공간에 대하여 이 계산이 필요하다.

두 가지의 계산시간을 비교하면 $2\pi N_L \approx \frac{3}{\pi^2} N_L^2$ 일 때, 2가지는 거의 비슷한 계산시간이 되고, N_L 이 커짐에 따라 LHT가 유리하게 된다. 실제 실험결과는 그것을 나타내고 있어서 $N_L = 16$ 에서는 LCCT가 유리한 것으로 되어 있다.

다음은 두 가지 방법의 [도수 피크 검출과 연신처리]의 계산시간을 검토한다. 4.1의 실험결과로부터 알 수 있듯이 LHT는 LCCT에 비하여 직선 분리 능력이 떨어진다. 그러나 LCCT와 같은 정도의 출력 결과를 얻기 위해서는 LHT의 피크 검출에 이용하는 임계값을 낮춰야 한다. 또한, 그림 4로부터 LHT는 투표도수분포가 복잡한 형상을 하고 있어서 피크의 수(local maximum)가 반드시 많게 되는 것을 알 수 있고, LCCT에서는 그림 2, 3으로부터 실제 직선을 적절히 반영한 투표가 이루어지므로 실제 직선 수에 근접한 피크 수의 검출을 기대할 수 있다. 실제 실험결과는 LCCT에서 검출되는 피크 수는 LHT에서 검출되는 피크 수와 비교하여 $1/16 \sim 1/4$ 이다.

도수 피크의 검출시간은 투표공간의 크기에 의존하기 때문에 투표공간이 큰 LCCT는 적당치 못하다. 그러나, 표로부터 알 수 있는 것처럼 검출된 피크 수는 적지 않다. 연신처리시간은 피크 수의 함수가 되므로, 그 시간은 LHT에 비하면 적게 된다. 투표공간의 크기는 N_L 이 작을 때는 LHT와 LCCT에서 균형을 이루기 때문에 (실제 $N_L = 16$ 일 때 $LHT/LCCT = 0.59$), 이 부분의 처리시간은 N_L 이 작을 때 LCCT가 유리하다. 실제 $N_L = 16$ 에서는 LCCT가 LHT보다 짧은 시간에 처리된다.

[직선의 기울기에 관한 검출 정밀도]

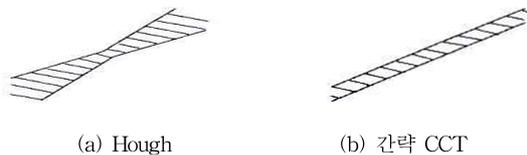


그림 15. 투표공간의 1점이 나타내는 화상 평면 위의 후보직선의 영역

Fig.15. Area of the candidate lines on the image plane corresponding to a point in the voting space.

Hough 변환, 간략 CCT에 있어서 각각의 투표공간의 1점이 나타내는 화상 평면 위의 직선들을 모형으로 표시하면 그림 15와 같은 영역으로 들어가는 직선이 된다. 이 그림으로부터 알 수 있는 것처럼 Hough 변환은 밖으로 벌어지는 영역이 되고, 간략 CCT에서는 평행사변형의 영역이 된다. 이것은 Hough에서는 기울기에 관한 검출 정밀도가 저하되는 것을 나타낸다. 이 결과 LHT에서는 연신처리에 있어서 기울기의 오차에 의하여 본래의 직선으로부터 벗어나 검출 정밀도가 떨어질 수가 있다.

V 결 론

디지털 직선의 구조를 조사한다고 하는 관점으로부터 검토를 위하여 디지털 직선을 완전히 분리하는 알고리즘을 유도하였다. 화상의 크기를 $N \times N$ 로 하면, 이 계산량이 $O(N^4)$ 로 많아서 실용적이지 못하므로, 검출 정밀도의 열화를 억제하면서 계산량을 $O(N^3)$ 로 하는 알고리즘을 제안하였다. 국소영역에 적용하여 선분 후보들을 구하기 위하여 그것을 연신처리하여 직선을 검출하는 방법에 본 방법을 적용하였다. 본 방법은 직선의 기울기의 분해능을 떨어뜨리지 않고, 직선을 고정밀도로 검출하였고, 그리고 처리의 고속성도 잃지 않는 방법이다. 실험 결과 종래의 법[5]과 비교하여 같은 정도 이하의 처리시간에 직선을 검출하였고, 검출 정밀도가 높은 방법이라는 것을 확인하였다.

이후 CCT법의 각도 분해능을 상세하게 검토하여 직선 검출 능력에 주어진 영향을 검토함과 동시에 LCCT 법의 고속화 가능성을 검토하고자 한다.

참고문헌

- [1] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," Commun. ACM, vol. 15, no. 1, pp. 11-15, 1972
- [2] P. K. Ser, W. C. Siu, "Invariant Hough transform with matching technique for the recognition of non-analytic objects," IEEE Int'l Conf. Acoustics, Speech, and Signal Processing, pp. 9-12, vol. 5, April 1993
- [3] Pawan Kumar, Virendra Singh, "Efficient regular expression pattern matching for network intrusion detection systems using modified word-based automata," Proc. 5th Int'l Conf. Security of Information and Networks (SIN '12), pp. 103-110, Oct. 2012
- [4] 정남채, "RANSAC에 기초한 화면내 평면 영역 샘플링에 의한 스테레오 화상의 대응 매칭," 한국신호처리시스템학회논문지, v.12, no.4, 242-249, 2011
- [5] Wang Min, Zhang Yanning, "Hough Transform Relative Nonuniform Parameter Space for the Detection of Line Segments," Int'l Conf. Computer Science and Software Engineering, pp. 764-767, Dec. 2008
- [6] M. Morimoto, S. Akamatsu, and Y. Suenaga, "A high-resolution Hough transform using variable filter," Proc. ICPR '92, pp. 280-284, 1992

- [7] S. Maji and J. Malik, "Object Detection Using a Max-Margin Hough Transform," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 529-539, Jan. 2009
- [8] R. Okada, "Discriminative Generalized Hough Transform for Object Detection," Proc. 12th IEEE Int'l Conf. Computer Vision, pp. 773-784, March 2009
- [9] I. D. Svalve, "Natural representations for straight lines and the Hough transform on discrete arrays," IEEE Trans. Pattern Anal. Mach. Intell., vol. 11, no. 9, pp. 941-950, Sept. 1989
- [10] Zhaoxia Fu, Yan Han, "A Circle Detection Algorithm Based on Mathematical Morphology and Chain Code," Int'l Conf. Computing, Measurement, Control and Sensor Network, pp. 253-256, July 2012
- [11] R. L. Graham, D. E. Knuth, and O. Patashnik, Concrete Mathematics, 2nd ed., Addison-Wesley, 1994
- [12] 後藤英昭, 阿曾弘具, "ハフ變換におけるパラメータの效率的サンプリング間隔," 信學論 (D-II), vol. J-81-D -II, no.4, pp.697-705, April 1998



정 남 채 (Nam-chae Jung)

正會員

1984년 2월 : 조선대학교 전자공학과(공학사)

1987년 2월 : 조선대학교 전자공학과(공학석사)

1992년 8월 : 조선대학교 전자전공(공학박사)

1996년 3월~현재 : 초당대학교 정보통신공학과 교수

※ 관심분야 : Digital 신호처리, Robotics, 의용생체전자공학