

# 기울기 기반 빠른 정상점 탐색에 대한 연구

안정호\*

## 요약

본 논문에서는 기울기 기반 빠른 정상점 탐색 방법론을 제시하고 시뮬레이션 결과를 제시한다. 제안하는 방법론은 유한 구간에서 단봉 그래프를 가지는 함수의 정상점(최대치)을 최대한 적은 수의 함수값을 참조하여 찾는 문제에 대한 솔루션으로, 대표적인 적용분야로는 자동초점이 있다. 우리는 정상점 탐색을 위해 주기적 탐색, 기울기 기반 탐색, 상세 탐색과 같이 세가지 탐색 전략을 통해 정상점을 찾는다. 노이즈가 첨가된 가우시안 함수를 발생시켜 시뮬레이션 한 결과 대략 8번탐색 횟수와 1.04의 오차를 보임을 알 수 있었다.

## A Simulation Study on the Fast Gradient-based Peak Searching Method

Ahn, Jung-Ho\*

### Abstract

In this paper we propose a new fast peak searching method using the gradient and present simulation results. The proposed method is a solution to the problem that finds the peak(maximum) of the unimodal function on a finite interval with minimum searching steps. Its main application is the auto-focus in the mobile phone. We propose the three steps to find the peak; periodic search, gradient-based search and detail search. In simulation we generated the Gaussian functions with white noise and have the result of about 8 searching steps and 1.04 errors on average.

**Keywords :** 정상점 탐색, 모드, 주기적 탐색, 기울기 탐색, 상세 탐색

### 1. 서론

멀티미디어에 의한 화려한 IT 기술은 20세기 말에 시작되어 21세기에 도 각광을 받을 것으로 기대되고 있다. 멀티미디어 콘텐츠를 생성해 내기 위한 다양한 방법론들이 연구되어 왔지만 멀티미디어 원 데이터가 숫자로 표현되는 이유로 멀티미디어 관련 원천기술의 개발은 많은 수학적 해법에 기반을 두고 있다. 이러한 문제들은 특히 최대치나 최소치를 찾는 최적화(Optimization) 문제로 귀결되는 경우가 많다. 우리는 본 논문에서 단봉함수의 모드(mode)를 빠르게 추정하는 방법론을 제안함으로써, 최대우도 추정(Maximum Likelihood Estimation)과 같은

통계적 추정이나 디지털 카메라나 핸드폰 카메라에서 제공되는 자동초점(AF: Autofocus)기술과 같은 문제의 해결 방법론을 제공하고자 한다.

제약조건이 없을 시 단봉함수의 모드를 찾는 문제는 간단하다. 정의역의 함수값을 모두 조사하고 그 중 최대값을 발생시키는 정의역의 값, 즉 모드를 찾아내면 된다. 하지만 이런 전 검사(full search) 방법론은 시간이나 프로세서와 같은 리소스(resource)가 제한적인 상황에는 맞지 않는다.

예를 들어 핸드폰에서의 자동초점 기술은 렌즈를 앞 뒤로 움직이면서 최대 대비(Contrast)를 발생시키는 렌즈의 위치를 찾아 영상을 촬영하게 된다. 이 때, 각 렌즈의 위치(x값)에 대해 영상을 모두 촬영해서 대비 값(y값)을 모두 계산하여 최대값을 찾은 후 최종 영상을 촬영한다면, 사용자가 기다리는 시간으로 인해 불편함을 느끼게 되어 자동초점이라는 기술의 실효성 문제를 야기한다.

이러한 이유로 실시간 탐색에 적합한 정상점

※ 제일저자(First Author) : 안정호  
접수일:2010년 01월 01일, 완료일:2010년 03월 30일  
\* 강남대학교 컴퓨터미디어정보공학부  
jungho@kangnam.ac.kr

(또는 모드) 추정을 위한 탐색방법들이 개발되어 왔다[5]. SOM과 같은 신경망(Neural Network)[1]을 이용한 방식, 차 방정식(Difference equation)[2]을 이용한 방식, 힐 클라이밍(Hill climbing)[3,8] 방식, 규칙성(Rule based method)[6]을 이용한 방식, 황금분할(Golden section) 방식, 이중 분할[9] 방식 등과 저 조도(Low illumination)에 강인한 방식 등 과거 10년 이상 빠른 정상점 탐색을 위한 다양한 방법론들이 개발되어 왔다. 본 논문에서는 기울기에 기반한 새로운 정상점 추정 방식을 제안한다.

본 논문의 1.1 절에서는 정상점 탐색 문제를 정의하고 제약조건을 기술하였고, 2절에서 제안하는 정상점 탐색 알고리즘을 주기적 탐색, 기울기 탐색, 상세 탐색 등 세 가지로 나누어 기술하였다. 3절에서는 제안한 알고리즘의 시뮬레이션 결과를 분석하였으며 4절에서는 결론과 앞으로의 연구 진행 방향을 제시하였다.

### 1.1. 정상점 탐색 문제

우리의 빠른 정상점 탐색의 문제는 아래와 같은 제약 조건을 가진다.

정상점 탐색을 위한 함수  $f$ 는 정상점  $(m_0, f(m_0))$ 을 중심으로 좌측은 단조증가, 우측은 단조감소하는 그래프를 가지는 단봉(unimodal) 함수이다. 하지만 실제 문제에서는 노이즈(Noise)가 첨가되는 경우가 많아 엄격한 의미의 단조성(monotonicity)이 보장되는 경우는 많지 않다. 따라서 우리는 다음과 같은 약한 의미의 단조성을 가정한다.

- (1)  $x_1 \leq x_2 \leq m_0$ 이면  $f(x_1) \leq f(x_2) + e$ ,
- (2)  $m_0 \leq x_1 \leq x_2$ 이면  $f(x_1) + e \geq f(x_2)$ .

여기서  $e$ 는 작은 양의 상수이다.

함수  $f$ 의 정의역은  $\{1, 2, \dots, N\}$ 이다. AF에서  $N$ 은 최대 스텝 사이즈라 불린다.

시작점  $x = 1$ 에서의 관측값  $f(1)$ 은 무조건 관측한다. AF와 같이 관측시에 시간 등의 자원이 소요되는 경우 관측 횟수가 적을수록 시스템 효율이 높아진다. 이와 같이 최소의 관측횟수를 요하는 시스템에서 첫 관측 지점을  $x = 1$ 로 고정한다는 것은 알고리즘 설계에 있어서 자유도가 줄었음을 의미한다.

본 논문의 궁극적인 목표는 최소의 관측(참

조)으로 함수  $f$ 의 최대치가 발생하는 지점  $x = m_0$ 를 찾거나  $m_0$ 와 가깝게 결정한다. 많은 응용분야에서 최고점은 안정적(stable)이기 때문에  $m_0$ 와 그 이웃점들 사이의 함수값은 큰 차이가 없다. 따라서 큰 차이가 발생하지 않는 범위에서  $m_0$ 와 다른 값을 솔루션으로 인정할 수 있는 경우가 많다.

## 2. 정상점 탐색 알고리즘

노이즈를 포함하는 단봉 함수의 탐색은 국소 최대값(Local minimum)을 찾을 위험이 있다. 우리는 이러한 위험을 최소화시키기 위해 주기적 탐색을 통해 주어진 구간을 탐색하여 부분 구간을 추출한다. 이 부분 구간은 기울기에 기반한 탐색 방법을 통해 모드를 포함할 확률이 높은 구간으로 줄여진 후, 상세탐색으로 정상점을 추정한다. 따라서 주기적 탐색, 기울기 탐색은 구간 추정(Interval estimation)을, 상세 탐색은 점 추정(point estimation)을 수행한다.

### 2.1. 주기적 탐색

주기적인 탐색은 국소 최대값에 빠지는 오류를 피하기 위한 목적으로 행해진다. 우리는 구간을 표시하는 기호  $[1, N]$ 을 자연수의 집합  $\{1, \dots, N\}$ 으로 정의한다. 먼저 정의역  $[1, N]$ 를  $n$ 등분하여 피벗치  $x_0 < x_1 < \dots < x_n$ 을 선택한다. 이 때,  $x_0 = 1, x_n = N$ 을 만족한다.

$x_0$ 부터 차례로 관측값을 조사하여 다음을 만족하는 인덱스  $L$ 을 구한다.

- (1)  $f(x_0) \gg f(x_1)$ 이면  $S = x_0, E = x_1$ ,
- (2)  $1 < i < N$  인 적당한 자연수  $i$ 에 대하여  $f(x_{i-1}) \ll f(x_i) \gg f(x_{i+1})$  이면  $S = x_{i-1}, M = x_i, E = x_{i+1}$ ,
- (3)  $f(x_{n-1}) \ll f(x_n)$ 이면  $S = x_{n-1}, E = x_n$

여기서 기호  $f(x) \ll f(y)$ 는 다음의 부등식을 만족하는 것으로 정의한다.

$$\frac{f(y) - f(x)}{y - x} > d$$

여기서  $d$ 는 작은 양의 정수(예: 0.1)로 이 부등식을 사용하면 노이즈에 덜 민감하게 되어 노이즈가 많이 포함되어 경우 제시하는 알고리즘의 강인성(robustness)을 보장해 준다.

다시 말해, 함수  $f$ 가 첫 피벗구간  $[x_0, x_1]$ 에서 감소하는 경우 그 구간을  $[S, E]$ 로 취하고, 그렇지 않은 경우 그래프 개형이 증가에서 감소로 바뀌는 구간을 피벗치를 기준으로 찾아내어  $M$ 을 포함하는 구간  $[S, E]$ 을 정한다. 만약  $f(x_0) \ll f(x_1) \ll \dots \ll f(x_n)$ 과 같이 모두 증가하는 경우  $[x_{n-1}, x_n]$ 을 구간  $[S, E]$ 로 정한다.

### 2.2. 기울기 탐색

주기적 탐색에 의해 우리는 정상점 탐색 구간을  $[1, N]$ 에서  $[S, E]$ 로 줄일 수 있었다. 초기 탐색구간을  $n$ 등분하여 탐색할 경우 구간  $[S, E]$ 의 길이  $L = E - S$ 의 기대치는 다음과 같다.

$$\begin{aligned} E[L] &= \frac{1}{n+1} \left( \frac{N}{n} + \frac{2N}{n}(n-1) + \frac{N}{n} \right) \\ &= \frac{2N}{n+1} \end{aligned}$$

$N=50, n=4$ 인 경우, 기울기 탐색을 위한 구간은 대략 20이다. 이는 3.2 절의 실험 결과를 통해서도 확인할 수 있었다.

이 구간의 길이를 정상점 위치를 추정함으로써 더 줄이는 것이 두 번째 단계인 기울기 기반 탐색 단계에서의 목표이다.

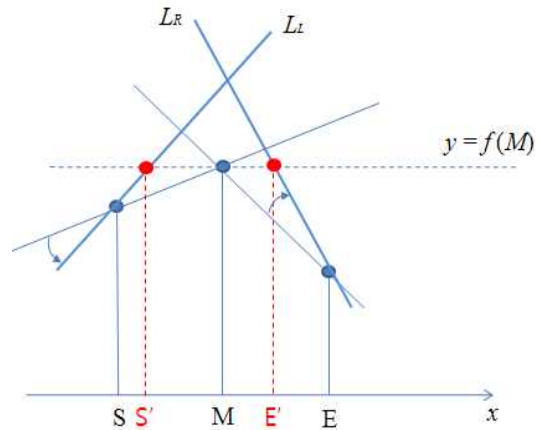
$S$ 와  $E$ 의 위치에 따라 세 가지 경우로 나누어 기울기 기반 탐색을 실시한다.

#### 2.2.1. $S > 1, E < N$ 인 경우

이는 정상점이 그래프의 양 끝에 가깝게 위치하지 않는 경우로  $f(S) \leq f(M) \geq f(E)$ 의 관계를 만족하며 위 2.1절의 (2)에 해당한다.

이 때 우리는 그림 1과 같이  $(S, f(S))$ 와  $(M, f(M))$ 을 잇는 직선의 기울기 보다 더 가파른 기울기를 가지는 직선  $L_L$ 과 직선

$y = f(M)$ 과의 교점에서 수직으로 내린 직선이  $x$ 축과 만나는 점을  $S'$ 이라 정의한다. 그리고  $(M, f(M))$ 과  $(E, f(E))$ 을 잇는 직선의 기울기 보다 더 가파른 기울기를 가지는 직선  $L_R$ 과 직선  $y = f(M)$ 과의 교점에서 수직으로 내린 직선이  $x$ 축과 만나는 점을  $E'$ 이라 정의한다. 이렇게 결정한 구간  $[S', E']$ 을 상세 탐색을 위한 구간으로 설정한다.



(그림 1)  $S > 1, E < N$ 인 경우, 기울기 기반 정상점 탐색

(그림 1)의 예에서  $f(S)$ 값이  $f(E)$ 보다 크므로 구간  $[M, E]$ 보다  $[S, M]$ 에서 정상점이 발생할 확률이 높아 보인다. 따라서 새로 구하는 구간의 길이는  $\overline{S'M}$ 이  $\overline{ME'}$ 보다 길게 나타난다. 또, 정상점은  $f(M)$  보다 높거나 같은 위치에 존재하기 때문에 직선  $L_L, L_R$ 과의 교점을 구하여 절단한다.

$S'$ 을 구하는 절차는 다음과 같다.  $(S, f(S))$ 와  $(M, f(M))$ 을 잇는 직선의 기울기  $m$ 은

$$m = \frac{f(M) - f(S)}{M - S}$$

이고 직선  $L_L$ 의 방정식은

$$y = (m + \alpha)(x - S) + f(S)$$

로 주어질 수 있다. 여기서  $\alpha$ 는 주어지는 양

의 상수로 실험에서는 1로 주어졌다. 즉 기울기를  $m$ 에서  $m+1$ 로 1만큼 증가시켜  $y = f(M)$ 과의 교점을 구한다. 따라서 교점  $S'$ 은

$$f(M) = (m + \alpha)(S' - S) + f(S)$$

로부터

$$S' = S + \frac{f(M) - f(S)}{m + \alpha}$$

로 주어진다. 마찬가지로 방법으로 교점  $E'$ 을 구할 수 있다. 다음 2.2.2절과 2.2.3절에서 교점을 구하는 방식도 이와 같다.

2.2.2.  $[S, E] = [x_0, x_1]$ 인 경우

이 경우는 2.1절의 (1)에 해당하는 경우로  $f(S) \geq f(E)$ 인 관계를 만족한다. 이 경우  $x_0$ 와  $x_1$ 의 중점  $M$ 을 추가하여 세 점  $S, M, E$ 에 대하여 다음 두 가지 경우에 대해 작업을 수행한다.

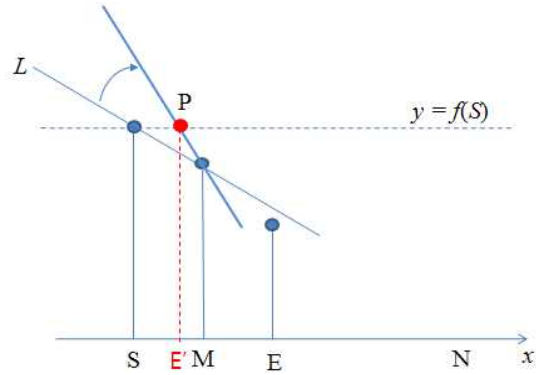
$f(S) \geq f(M)$ 인 경우,  $S$ 와  $M$ 사이에 정상점이 위치한다고 볼 수 있다. 정상점을 포함하는 구간을 더 작게 추정하기 위해 그림 2에서와 같이  $(S, f(S))$ 와  $(M, f(M))$ 을 잇는 직선  $L$ 보다 기울기가 좀 더 가파른 선과  $y = f(S)$ 와의 교점  $P$ 에서 수직으로 내린 직선이  $x$ 축과 만나는 점을  $E'$ 이라 할 때 상세 탐색을 위한 구간  $[S, E']$ 을 얻는다.

$f(S) \leq f(M) \leq f(E)$ 인 경우,  $M$  주변에 정상점이 위치한다고 볼 수 있다. 이 경우는 앞의 2.2.1와 같은 경우로 동일한 방법으로 좁혀진 구간  $[S', E']$ 을 설정할 수 있다.

2.2.3.  $[S, E] = [x_{n-1}, x_n]$ 인 경우

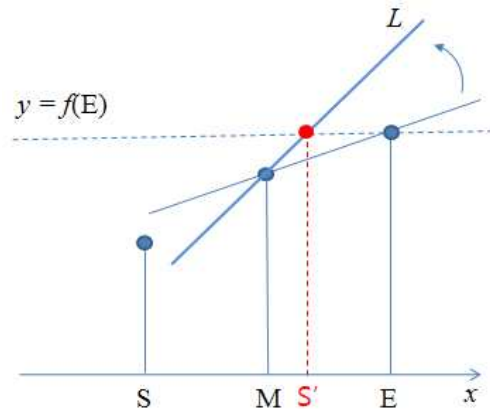
이 경우는 2.1절의 경우 (3)에 해당하는 경우로  $E = N$ 이고  $f(S) \leq f(E)$ 인 관계를 만족한다.

이 경우  $S$ 와  $E$ 의 중점  $M$ 을 추가하여 세 점  $S, M, E$ 에 대하여 다음 두 가지 경우에 대해 작업을 수행한다.



(그림 2)  $S=1, f(S) \geq f(M)$ 인 경우, 기울기 기반 정상점 탐색

$f(M) \leq f(E)$ 인 경우는 그림 3과 같은 경우로 단조성으로 인해  $M$ 과  $E$ 사이에서 정상점이 발생한다고 할 수 있고  $E$ 에서의 함수값이  $M$ 에서의 함수값보다 크므로  $E$ 를 좌 끝점으로 하고  $M$ 보다 큰 값을 시작점으로 하는 구간을 추정한다.



(그림 3)  $E=N, f(M) \leq f(E)$ 인 경우 기울기 기반 정상점 탐색

우리는  $(M, f(M))$ 과  $(E, f(E))$ 을 잇는 직선의 기울기보다 더 가파른 기울기를 가지는 직선  $L$ 에 대하여 직선  $y = f(E)$ 와의 교점의  $x$ 좌표를  $S'$ 을 정하여 상세 탐색을 위한 구간  $[S', E]$ 를 결정한다.

2.3. 상세 탐색(Detail Search)

주기적 탐색과 기울기 기반 탐색이 모드를 포함하는 작은 구간을 추정(Interval estimation)하는 것이었다면, 상세 탐색은 구간 내에서 모드를 결정하는 점추정(Point estimation) 작업이다.

주기적 탐색과 기울기 기반 탐색을 통해 얻어진 구간  $I_D$ 를  $[S', E']$ 라 할 때, 이 구간이 충분히 작다는 가정 하에 많은 참조(탐색)를 통해 모드를 결정할 수 있다. 하지만 탐색횟수가 증가하는 것은 바람직하지 않기 때문에 모드와 충분히 가까운 점을 추정하는 방식으로 상세탐색을 수행한다.

상세 탐색은  $S'$ 과  $E'$  사이에 기 관측점(observed point)의 유무에 따라 다음과 같이 두 가지로 나누어 실시한다.

2.3.1.  $[S', E']$ 에 기 관측점이 존재하지 않는 경우

이는 2.2.2절과 2.2.3절의 절차에 따라 상세 탐색구간이 정해진 경우로 원 탐색구간  $[1, N]$ 의 양 끝점을 포함하는 경우이다. 참고로, 이 경우는 다음 2.3.2절의 경우 보다 약 절반정도로 탐색구간이 작다. 따라서 많은 탐색을 하지 않아도 근사 솔루션을 구할 수 있다. 우리는  $S'$ 과  $E'$ 의 중점  $M'$ 의 함수값  $f(M')$ 과  $f(S')$ ,  $f(E')$ 의 크기를 비교하여 다음을 수행한다.

- (1)  $f(S')$ 이 최대일 때,  $S'$ 과  $M'$ 의 중점  $M''$ 을 취하여  $f(S')$ 와  $f(M'')$ 중 최대값을 정상점으로 추정한다.
- (2)  $f(E')$ 이 최대일 때,  $E'$ 과  $M'$ 의 중점  $M''$ 을 취하여  $f(E')$ 와  $f(M'')$ 중 최대값을 정상점으로 추정한다.
- (3)  $f(M')$ 이 최대일 때,  $f(S') \geq f(E')$ 이면  $f((S' + M')/2)$ 와  $f(M')$ 중 큰 값을 정상점으로 추정하고,  $f(S') \leq f(E')$ 이면  $f((E' + M')/2)$ 와  $f(M')$ 중 큰 값을 정상점으로 추정한다.

2.3.2.  $[S', E']$ 에 기 관측점이 존재하는 경우

이는 2.2.1절의 절차에 따라 상세탐색 구간이 결정된 경우로 2.2.1절이나 2.2.2절의 절차에 의해 생성된 구간보다 구간의 길이가 더 길다.

이 경우 우리는 다음 규칙에 의해 탐색점  $M''$ 을 결정한다.

$$\begin{aligned} \overline{S'M'} > \overline{M'E'} \text{ 이면 } M'' &= (S' + M')/2, \\ \overline{S'M'} < \overline{M'E'} \text{ 이면 } M'' &= (M' + E')/2 \\ \overline{S'M'} &= \overline{M'E'} \text{ 일 경우,} \\ f(S') \geq f(E') \text{ 이면 } M'' &= (S' + M')/2, \\ f(S') < f(E') \text{ 이면 } M'' &= (E' + M')/2. \end{aligned}$$

이후,  $S', M', M'', E'$ 에서의 함수값을 비교하여 최대값을 취하는 점을 정상점으로 추정한다. 다음 표 1은 제안한 세 가지 탐색방법과 그 특징을 요약한 것이다.

<표 1> 탐색 방법과 특징

탐색 방법	특징
주기적 탐색	<ul style="list-style-type: none"> <li>· 국소 최소값을 피하기 위해 주어진 구간 <math>[1, N]</math>을 <math>n</math> 등분하여 탐색</li> <li>· 크기 비교시 노이즈에 강인한 기울기 방법 사용</li> </ul>
기울기 탐색	<ul style="list-style-type: none"> <li>· 주기적 탐색의 결과 주어진 구간의 위치에 따라 세 가지 경우로 나누어 처리</li> <li>· 기울기에 기반한 추론으로 작은 구간을 생성</li> </ul>
상세 탐색	<ul style="list-style-type: none"> <li>· 확률적 추정</li> <li>· 빠른 근사 솔루션 제공</li> </ul>

### 3. 시뮬레이션 및 결과

우리는 주어진 단봉 함수의 모드(mode)를 최소의 참조를 통해 (가장 가깝게) 찾아내는 것을 목적으로 하는 알고리즘을 제안하였다. 우리는 시뮬레이션을 통해 제안한 알고리즘의 성능을 정확성과 신속성 면에서 입증하였다.

#### 3.1. 시뮬레이션 환경

제안한 기울기 기반 정상점 탐색 알고리즘의 성능을 측정하기 위해 다음과 같은 환경에서 시뮬레이션을 수행하였다.

총 스텝 수  $N$ 은 자동초점을 사용하는 실제 모바일 핸드폰의 렌즈 스텝 수와 동일하게 50으로 설정하였다. 따라서 우리는 구간  $[1, 50]$ 에

서 탐색을 수행한다. 위 구간을  $n=4$  등분하여 주기적 탐색을 수행하였다. 따라서 우리는 1, 13, 25, 38, 50의 순으로 탐색을 한다.  $n$ 값이 너무 클 경우 노이즈에 민감하고 탐색 횟수가 증가하여 성능이 저하될 수 있다.

정상점 탐색을 위한 함수  $f$ 는 다음과 같이 정규분포 함수에 화이트 노이즈를 첨가하여 발생시켰다.

$$f(x) = \frac{A}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} + e_x$$

여기서  $A$ 는 양의 상수,  $e_x \sim N(0, 1)$ 이다.

시뮬레이션을 위해 상수  $\mu$ 는 1과  $N$ 사이에서 랜덤으로 발생시켰고  $A$ 는 500,  $\sigma$ 는 10으로 고정시켰다. 참고로 상수  $A$ 와  $\sigma$ 는 실제와 비슷한 그래프를 재현하기 위한 것으로 성능에는 거의 영향을 미치지 않는다. 백색잡음  $e_x$ 는 1과 50사이의 각각의  $x$ 에 대하여 표준 정규분포로부터 발생시켰다. 이 백색잡음 때문에 랜덤 함수  $f$ 는 약한 의미에서의 단조성을 가지는 단봉 함수가 된다. 노이즈가 첨가되었으므로  $\mu$ 가 함수  $f$ 의 모드가 아닐 수 있다, 따라서 우리는  $f$ 의 최대값을 가지는 점을 조사하여 이 때  $x$ 값의  $x_0$ 를 모드로 설정하고  $x_0$ 를 탐색하는 작업을 수행하였다.

### 3.2. 시뮬레이션 결과 분석

주기적 탐색을 위하여 피봇값 1, 13, 25, 38, 50의 순으로 함수값을 조사하여 실질적으로 함수값이 감소하는 지점을 찾는 과정을 2.1절의 절차에 따라 수행하였다. 노이즈에 강인하게 기울기의 실질적 감소를 측정하기 위해  $x > y$ 에 대하여 다음과 같을 때 함수값  $f(x)$ 가  $f(y)$ 이 실질적으로 크다고 설정하였다.

$$f(x) - f(y) > -0.1(x - y)$$

이러한 고려없이 단순히 두 함수 값만을 비교할 경우 노이즈에 의해 잘못된 결과가 많이 발생함을 실험적으로 확인할 수 있었다. 실험결과 주기적 탐색에서 구한 구간의 99.93%가 정상점을 포함함을 확인할 수 있었다. 여기서 실패하는

0.007%는 모두 노이즈로 인해 발생한다. 즉 그래프가 증가 추세에 있으나 노이즈로 인해 감소 추세로 오인한 것이다.

주기적 탐색을 통해 얻어진 구간  $[S, E]$ 에 대해 기울기 탐색을 수행할 경우 탐색을 위해 새롭게 함수값을 참조하는 일은 양끝점 이외에는 일어나지 않는다. 따라서 이 때의 평균 탐색횟수는 0.26번으로 나타났다.

<표 2>는 10000번 실험에 대한 주기적 탐색과 기울기 탐색의 결과를 보여주고 있다. 성공률은 탐색을 통해 생성된 구간이 실제 모드값을 포함하는 비율을 보여준다. 우리는 주기적 탐색을 통해 평균 3.88번의 탐색(함수값 참조)으로 구간의 길이를 40.78%(=20.39/50)로 줄일 수 있었다. 추가로 기울기 탐색을 수행함으로써 4.14번의 탐색으로 구간의 길이를 25.34%(=12.67/50)로 줄일 수 있었다.

<표 2> 주기적 탐색 및 기울기 탐색 결과

	성공률	구간길이	탐색횟수
주기적 탐색	99.94%	20.39	3.88
기울기 탐색	99.87%	12.67	0.26

주기적 탐색과 기울기 탐색 이후 상세 탐색을 통해 모드를 추정한다. 우리는 상세 탐색에서 정확한 모드를 구하기 위해 탐색횟수를 늘리는 정책(strategy)을 사용하지 않았다. 연속함수라는 가정하에 모드 주변의 값은 모드의 함수값과 유사하기 때문에 모드를 대신하여 사용할 수 있다고 판단되었다. 따라서 우리는 약 2.97번의 탐색을 통해 모드를 추정했고 이 때의 오차는 추정값과 모드값의 차의 절대값으로 계산하였다. 즉 오차  $E$ 는

$$E = \frac{1}{n} \sum_{i=1}^n |(m_o^i - \tilde{m}^i)|$$

여기서  $m_o^i$ 와  $\tilde{m}^i$ 는  $i$  번째 시뮬레이션에서 실제 모드 값과 추정된 모드 값이다. 아래 <표 3>에서 보여지듯이 10000번의 시뮬레이션에 대한 평균 오차는 약 한 단위 정도(1.04)로 나타났다.

<표 3> 상세검색 결과

	오차	전이오차	탐색횟수
상세 검색	1.04	18.54	2.97

전이오차(Induced error)는 주기적 탐색과 기울기 탐색에서 모드를 포함하는 구간을 제시하지 못한 경우에 상세탐색으로 추정된 모드와 실제 모드와의 거리의 평균이다. 이 전이오차가 오차에 포함되어 있는 것을 생각할 때 상세 탐색은 평균적으로 한 단위 이하의 오차를 보인다고 할 수 있다.

### 4. 결론 및 토의

본 논문에서는 기울기를 이용한 빠른 정상점 탐색 방법론을 제안하였다. 제안하는 방법론은 주기적 탐색, 기울기 탐색, 상세탐색 등 세 가지 탐색과정을 포함한다. 주기적 탐색을 통해 국소 최대값을 피하면서 노이즈에 강인하게 모드를 포함하는 구간을 추정할 수 있었고 이렇게 추정된 구간은 기울기 기반 탐색을 통해 더 작은 구간으로 추정되었다. 이후 상세탐색을 통해 확률적으로 모드를 추정할 수 있었다. 시뮬레이션을 통해 제안한 방법론은 속도와 정확도 면에서 모두 훌륭한 결과를 보임을 보일 수 있었다. 구간의 길이가 모바일 핸드폰의 스텝 수 50에 대해 평균 7.1번의 탐색을 수행하여 1.04 정도의 정확도를 보였다. 탐색횟수가 데이터에 의존적이지 않고, 스텝 수  $N$ 과 피벗횟수  $n$ 에 대한 함수로 결정되는 것은 제안한 방법론의 큰 장점이라 할 수 있다. 우리는 이 알고리즘을 핸드폰의 자동초점 기술에 적용하여 개발하고 있다.

### 참고 문헌

[1] C.Y. Chen, R.-C. Hwang, "An Embedded Lens Controller for Passive Auto-Focusing Camera Device Based on SOM Neural Network", TENCON 2006, IEEE Region 10 Conference, pp. 1-4, Nov. 2006.  
 [2] C.-M. Chen, C.-M. Hong, and H.-C. Chung, "Efficient Auto-Focus Algorithm Utilizing Discrete Difference Equation Prediction Model for Digital Still Camera",

as", IEEE Transactions on Consumer Electronics, Vol.52(4), pp. 1135-1143, 2006.  
 [3] K.-S. Choi, J.-S. Lee, and S.-J. Ko, "New Autofocusing Technique Using the Frequency Selective Weighted Median Filter for Video Cameras", IEEE Transactions on Consumer Electronics, Vol. 45(3), pp. 820-827, 1999.  
 [4] M. Gamadia, N. Kehtarnavaz, K. R.-Hoffman, "Low-Light AutoFocus Enhancement for Digital and Cell-Phone Camera Image Pipelines", IEEE Transactions on Consumer Electronics, Vol. 53(2), pp. 249-257, 2007.  
 [5] M. Gamadia and N. Kehtarnavaz, "A Real-time Continuous Automatic Focus Algorithm for Digital Cameras", IEEE Southwest Symposium on Image Analysis and Interpretation, pp. 163-167, 2006.  
 [6] N. Kehtarnavaz, H.-J. Oh, "Development and real-time implementation of a rule-based auto-focus algorithm", Real-Time Imaging, Vol. 9, pp. 197-203, June 2003.  
 [7] W. H. P. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, "Numerical Recipes in Fortran, second edition", Cambridge University Press, pp. 390-395, 1992.  
 [8] J. He, R. Zhou, and Z. Hong, "Modified Fast Climbing Search Auto-Focus Algorithm with Adaptive Step Size Searching Technique for Digital Camera", IEEE Transactions on Consumer Electronics, Vol. 49(2), pp. 257-262, 2003.  
 [9] 안정호, "두 점의 관계를 이용한 정상점 검색 알고리즘에 관한 연구", 한국디지털콘텐츠학회 학술대회 논문집, pp. 273-275, Dec. 2009.



### 안 정 호

1998년 : 연세대학교 수학과(이학석사)  
 2001년 : Texas A&M University 통계학과(이학석사)  
 2006년 : 연세대학교 컴퓨터과학과 (공학박사)  
 2006년~2006년 : 연세대학교 BK21 지능형모바일서비스를 위한 차세대 단말소프트웨어 사업단 박사후 연구원  
 2007년~현재 : 강남대학교 컴퓨터미디어정보공부 교수  
 관심분야 : 패턴인식, 얼굴인식, 컴퓨터비전, 로봇비전, 영상처리 등