

저사양 임베디드 시스템에서의 실시간 응답이 가능한 터치 기능 연구

이용민^{1,2*}, 한창호³

¹선문대학교 디스플레이반도체학과, ²선문대학교 나노과학연구소, ³선문대학교 정보통신공학과

Research on Touch Function capable of Real-time Response in Low-end Embedded System

Yong-Min Lee^{1,2*}, Chang Ho Han³

¹Dept. of Display & Semiconductor Engineering, Sunmoon University

²Research Center for Nano-Bio Science, Sunmoon University

³Dept. of Information Communication Engineering, Sunmoon University

요약 본 논문은 낮은 사양의 임베디드 시스템에서 터치 입력시에 실시간 응답특성을 나타내기 위해 보간법을 이용한 알고리즘을 도입하여 실시간 응답 처리가 가능한 터치 스크린을 구현하는 연구에 관한 것이다. 본 실험에서는 2점 데이터에서 1차 다항식을 도출하여 임의의 데이터를 추정하는 선형 보간 알고리즘과 3점 데이터에서 2차 다항식을 도출하여 임의의 데이터를 추정하는 라그랑지(Lagrange) 보간 알고리즘이 적용되었다. 실험결과로써, 라그랑지 보간법이 선형보간법보다 수식이 복잡하여 처리속도가 느려서 글씨도 매끄럽지 못함을 알게 되었다. 선형 보간법을 사용시 화면에 표시되는 속도가 라그랑지 보간법 사용시보다 2.4배 빠름을 확인하였다. 실시간 응답특성을 위해서는 알고리즘 자체의 우수성보다는 실행파일 크기가 더 작은 알고리즘이 더 유리하다는 점을 확인하였다. 결론적으로, 저사양 임베디드 시스템에서 실시간 응답특성을 확보하기 위해서는 상대적으로 간단한 선형보간법 알고리즘 채용이 라그랑지 보간법을 사용하는 것보다 더 우수한 실시간 응답특성의 터치동작을 수행함을 확인하였다.

Abstract This paper presents a study to implement a touch screen capable of real-time response processing in a low-end embedded system. This was done by introducing an algorithm using an interpolation method to represent real-time response characteristics when a touch input is performed. In this experiment, we applied a linear interpolation algorithm that estimates random data by deriving a first-order polynomial from 2-point data. We also applied a Lagrange interpolation algorithm that estimates random data by deriving a quadratic polynomial from 3-point data. As a result of the experiment, it was found that the Lagrange interpolation method was more complicated than the linear interpolation method, and the processing speed was slow, so the text was not smooth. When using the linear interpolation method, it was confirmed that the speed displayed on a screen is 2.4 times faster than when using the Lagrange interpolation method. For real-time response characteristics, it was confirmed that smaller size of the executable file of the algorithm is more advantageous than the superiority of the algorithm itself. In conclusion, in order to secure real-time response characteristics in a low-end embedded system, it was confirmed that a relatively simple linear interpolation algorithm performs touch operations with better real-time response characteristics than the Lagrange interpolation method.

Keywords : Touch Interface, Touch Panel, Linear Interpolation, Lagrange Interpolation, Embedded System

*Corresponding Author : Yong-Min Lee(Sunmoon Univ.)

email: ymlee@sunmoon.ac.kr

Received January 21, 2021

Accepted April 2, 2021

Revised March 22, 2021

Published April 30, 2021

1. 서론

터치인터페이스 기술은 각종 컴퓨터 및 스마트폰 등 모바일기기의 입력장치로서 핵심기술로 자리 잡고 있다. 터치펜이나 손가락으로 글씨를 입력할 경우 실시간 처리 기능이 매우 중요하다. 일반적인 모바일 기기에서는 프로세서의 성능이 우수하여 실시간 터치 입력기능에 문제가 없지만 낮은 사양의 프로세서를 채용한 임베디드 시스템에서 터치 입력 기능을 사용할 경우 일반적으로 임베디드 프로세서의 성능이 일반 컴퓨터나 스마트폰등의 프로세서에 비해 떨어지므로 실시간 터치 응답속도를 나타내는데 제약이 있어 개선하려는 시도가 이루어지고 있다[1-3].

예를 들어 터치 스크린에서 펜으로 선을 그릴 때 터치 응답속도가 느릴 경우 스크린에 원하는 선이 그려지지 않고 마치 점선을 표시한 것처럼 불연속적으로 나타나게 된다. 따라서 터치 펜을 느리게 동작시키거나 사용하는 프로세서를 고속동작이 가능한 고가의 제품을 사용하지 않으면 문제를 해결할 수가 없다. 또 다른 방법으로 다수의 프로세서를 사용하거나 시스템내의 알고리즘을 보강하여 반응속도를 증가시키는 방법이 있다. 하지만 저사양의 시스템에서 고가의 프로세서나 여러개의 프로세서를 사용하는 방법은 좋은 해결방법이라고 볼 수 없다.

본 논문에서는 낮은 사양의 임베디드 시스템에서 터치 입력시에 실시간 응답특성을 나타내기 위해 보간법을 이용한 알고리즘을 도입하여 실시간 처리가 가능한 터치 스크린을 구현하는 연구에 관한 것이다. 2장에서는 선형 보간법과 라그랑지 보간법에 대한 이론적인 설명을 진행하고 이를 구현한 터치 알고리즘을 실험하기 위한 테스트 장치 구성 및 실험 결과에 대해 토의하였다. 3장에서는 결론 및 향후 연구과제에 대해 기술하였다.

2. 본론

2.1 보간법의 개요

보간법은 주어진 데이터의 점들이 불연속적일 경우 그 점들 사이의 값을 추정하여 데이터를 보강하는 방법이다. 우선 주어진 데이터의 점들을 지나는 추정 함수를 구하고 원하는 부분의 변수값을 대입하여 추정 함수 값을 계산하면 된다. 본 논문에서는 주어진 2점의 데이터로부터 일차 다항식을 도출하여 임의의 데이터를 추정하는 선형 보간법과 3점의 데이터로부터 2차 다항식을 도출하여 임의의 데이터를 추정하는 라그랑지 (Lagrange) 2차 보간

법을 각각 터치 알고리즘에 적용하여 실험한다.

2.1.1 선형 보간법 (Linear Interpolation)

선형 보간법[4] 은 Fig. 1과 같이 2개의 주어진 데이터 점을 지나는 1차식의 추정함수를 구하고 임의의 변수 값에 대한 추정함수의 해를 구하는 것이다. 원래 함수 $f(x)$ 는 임의의 변수 n 개의 점 x_1, x_2, \dots, x_n 에 대한 함수 값을 정의할 수 있다. 함수 $f(x)$ 의 임의의 두점 $(x_i, f(x_i))$ 와 $(x_{i+1}, f(x_{i+1}))$ 를 지나는 직선의 식 $g(x)$ 는 Eq. (1)과 같이 정의할 수 있다.

$$g(x) = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}(x - x_i) + f(x_i) \quad (1)$$

Where, x_i and x_{i+1} denote random points, $f(x_i)$ and $f(x_{i+1})$ denote function value of the random points, respectively

추정 함수 $g(x)$ 의 값은 두 점 x_i 와 x_{i+1} 사이의 임의의 x 값에 대한 선형 보간 값을 나타내게 된다. 임의의 점 a 에 대해서 원래의 함수 값은 $f(a)$ 가 되겠지만 선형 보간법을 이용하여 그의 추정치인 $g(a)$ 의 값을 구할 수 있다. 따라서 선형 보간법을 이용한 알고리즘을 적용하면 터치 스크린에서 취득한 두 점 x_i 와 x_{i+1} 을 이용하여 그 두 점 사이의 데이터를 추정하여 스크린상에 두 점을 이룰 수 있는 데이터를 구하여 표시할 수 있다.

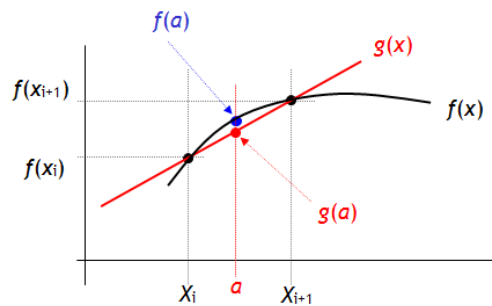


Fig. 1. Linear interpolation

2.1.2 라그랑지 보간법 (Lagrange Interpolation)

라그랑지 보간법[5]은 일반적으로 n 개의 주어진 데이터 점으로부터 $n-1$ 차의 추정함수를 구할 수 있다. 본 논문에서는 Fig. 2와 같이 3개의 주어진 데이터 점을 지나는 2차식의 추정함수를 구하고 임의의 변수 값에 대한 추정 값을 구하는 방법을 사용한다. 이 보간법은 주어진 데

이터 점과 점 사이를 단순히 직선으로 연결하는 것이 아니라 세 점을 지나는 곡선의 2차함수를 구하여 주어진 데이터 점 사이를 곡선으로 연결하게 된다. 함수 $f(x)$ 는 원래 함수를 나타내며 임의의 세점 $(x_i, f(x_i))$, $(x_{i+1}, f(x_{i+1}))$ 와 $(x_{i+2}, f(x_{i+2}))$ 를 지나는 곡선의 2차함수 근사식 $g(x)$ 는 Eq. (2)과 같이 정의할 수 있다.

$$g(x) = \frac{(x-x_{i+1})(x-x_{i+2})}{(x_i-x_{i+1})(x_i-x_{i+2})}f(x_i) + \frac{(x-x_{i+2})(x-x_i)}{(x_{i+1}-x_{i+2})(x_{i+1}-x_i)}f(x_{i+1}) + \frac{(x-x_i)(x-x_{i+1})}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})}f(x_{i+2}) \quad (2)$$

Where, x_i , x_{i+1} and x_{i+2} denote random points, $f(x_i)$, $f(x_{i+1})$ and $f(x_{i+2})$ denote function value of the random points, respectively

추정 함수 $g(x)$ 의 값은 두 점 x_i 와 x_{i+1} 사이의 임의의 x 값에 대한 곡선의 보간 값을 나타낸다. 임의의 점 a 에 대해서 원래의 함수 값은 $f(a)$ 가 되겠지만 라그랑지 보간법을 이용하여 그의 추정치인 $g(a)$ 의 값을 구할 수 있다. 이 보간법을 이용하면 곡선상의 데이터값을 취득함으로써 선형보간법보다 좀 더 원래 함수에 가까운 데이터 값을 추정 할 수 있을 것으로 예측된다.

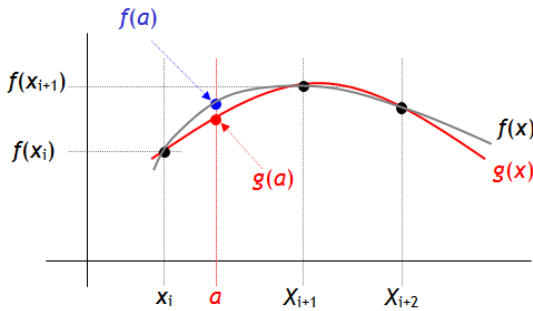


Fig. 2. Lagrange interpolation

2.2 실험 방법

위의 보간법을 적용한 터치응답속도 실험을 위해 실험 장치를 Fig. 3과 같이 터치LCD와 이를 구동하기 위한 마이크로프로세서로 구성하였다. 터치LCD는 320×240 도트의 3.5인치 컬러LCD를 사용하였다. 터치 패널에 채용된 터치 방식은 저항막 방식이며 터치 펜을 사용하여 터치동작을 인식하도록 되어 있다. 마이크로 프로세서는

Atmel사의 Atmega128 CPU를 사용한 보드로 구성하였다. 프로그램 소스코드는 AVR Studio4 통합개발환경에서 C++을 이용하여 작성하였고 AVR용 툴체인은 WinAVR과 c언어 교차 컴파일러는 avr-gcc++.exe를 사용하였다.

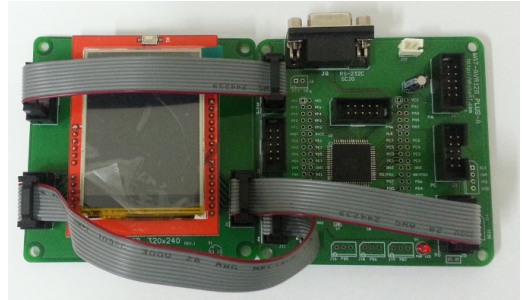


Fig. 3. Experimental setup

실험은 알고리즘별로 3가지의 터치동작 결과를 비교 진행하였는데 우선 보간법을 사용하지 않은 알고리즘 사용 시 터치동작을 관찰하고, 선형보간법과 라그랑지 보간법을 사용시의 터치동작을 비교 검토하였다. 첫 번째 실험은 터치 화면에 직선을 그리는 동작을 실행했을 때의 터치동작을 비교하였고 두 번째 실험은 터치 화면에 문자를 그릴 때의 터치동작을 비교하는 실험을 진행하였다.

2.3 실험 결과 및 토의

첫 번째 실험은 Fig. 4와 같이 터치화면에 직선을 그릴 때 보간법별로 터치동작의 결과를 비교하였다. 보간법을 사용하지 않을 경우 Fig. 4(a)와 같이 터치 화면에 직선을 그리면 점선의 형태로 나타난다. 이는 CPU의 처리속도 문제로 하드웨어적인 한계이므로 소프트웨어적으로 해결하는 수 밖에 없다. 선형 보간법과 라그랑지 보간법의 알고리즘을 적용할 경우 Fig. 4(b), (c)와 같이 직선이 그려짐을 보여준다. 처리속도의 비교를 위해 선을 그릴 때 점선이 의도적으로 짝히도록 프로그래밍 하였다. 즉 직선상에 점이 많을수록 처리속도가 빠름을 의미한다. 두 보간법이 모두 문제없이 직선을 그려냄을 알 수 있으며 선형보간법이 라그랑지 보간법보다 처리속도가 빠름을 알 수 있다. 처리속도의 비교를 위해 Table 1과 같이 보간 알고리즘 별로 직선상의 점의 수를 세어 화면에 디스플레이되는 속도를 비교하였다. 점의 수의 측정은 알고리즘별로 총 10회를 측정하여 평균값으로 나타내었고 알고리즘이 없을 경우를 기준으로 다른 알고리즘의 디스플레이 되는 속도를 비교하였다. 알고리즘을 사용하지 않을

경우의 디스플레이 되는 속도를 100%로 간주하였을 때 선형 보간법 사용시에는 표시 속도가 79%이고 라그랑지 보간법을 사용시에는 표시 속도가 33%를 나타내었다. 또한 선형 보간법의 경우가 라그랑지 보간법 사용시보다 2.4배의 빠른 표시속도를 나타내었다.

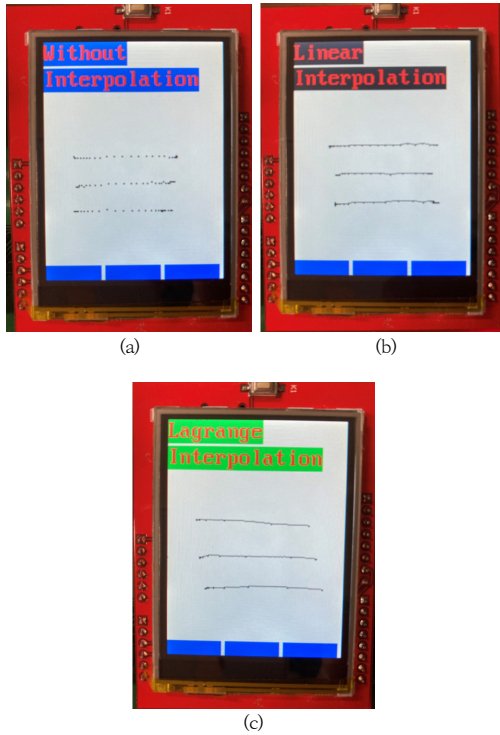


Fig. 4. Experimental result of drawing straight lines on the touch screen for each algorithm
(a) without interpolation (b) with linear interpolation
(c) with Lagrange interpolation

Table 1. Comparison of display speed by algorithm

Algorithm	Average number of the points on the straight line	Percentage comparison of display speed
without interpolation	21.5	100 %
linear interpolation	16.9	79 %
Lagrange interpolation	7.1	33 %

두 번째 실험은 Fig. 5와 같이 터치화면에 문자를 쓸 때 보간법별로 터치동작의 결과를 비교하였다. 보간법을 사용하지 않을 경우 Fig. 5(a)와 같이 터치 화면에 알파벳 문자를 그리면 첫 번째 실험의 직선을 그릴 때와 마찬가지로 점선의 형태로 나타난다. 선형 보간법과 라그랑지

보간법의 알고리즘을 적용할 경우 Fig. 5(b), (c)와 같이 알파벳 문자가 그려짐을 볼 수 있다. 선형보간법을 채용한 알고리즘이 라그랑지 알고리즘을 채용한 경우보다 더 보기 좋게 잘 쓰여짐을 알 수 있다. 당초에는 라그랑지 보간법이 더 우수한 필기성능을 보일 것으로 예상하였지만 라그랑지 보간법 수식이 복잡하여 프로그램 실행파일이 커져서 처리속도가 느리고 글씨도 매끄럽지 못함을 알게 되었다. 실시간 터치기능을 위해서는 알고리즘자체의 우수성도 중요하지만 알고리즘의 실행파일 크기가 작아야 실시간 처리에 유리하다는 점을 확인하였다. 본 실험을 통해 저사양 임베디드 시스템용으로 실시간 터치기능을 위해서는 선형보간법이 라그랑지보간법에 비해 더 우수한 터치동작을 수행함을 알 수 있었다.

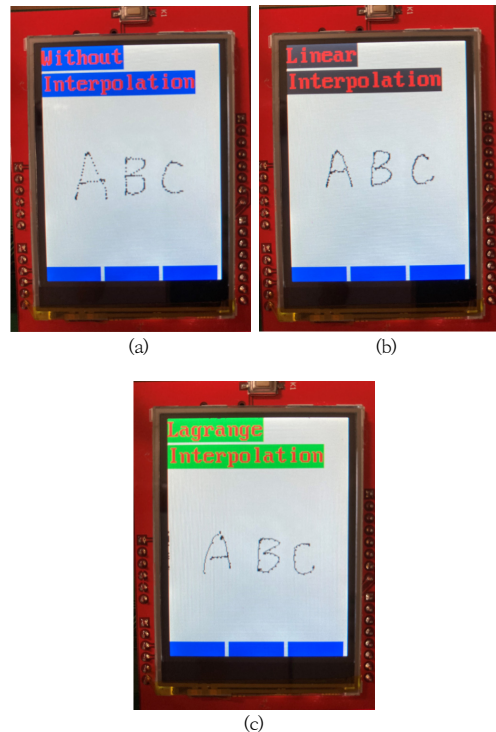


Fig. 5. Character writing experimental results for each algorithm
(a) without interpolation (b) with linear interpolation
(c) with Lagrange interpolation

3. 결론

본 연구에서는 저사양 임베디드 시스템에서 실시간 터치기능을 확보하기 위해서는 상대적으로 간단한 선형보간법 알고리즘 채용이 라그랑지 보간법을 사용하는 것보

다 더 우수한 실시간 터치동작을 수행함을 확인하였다. 본 연구에서 검토한 보간법은 주로 다항식 보간법이며 향후 연구과제로서 스프라인 보간법등 기타 보간법들을 검토하여 실시간으로 터치동작이 좀더 우수한 알고리즘을 구현하고자 한다.

References

- [1] S. K. Kwon, D. S. Lee, "Implementation of Real-time Virtual Touch Recognition System in Embedded System", *J. Korea Multimedia Society*, Vol.19, No.10, pp.1759-1766, Oct. 2016. DOI: <http://dx.doi.org/10.9717/kmms.2016.19.10.1759>
- [2] J. A. Stankovic, "Misconceptions About Real-Time Computing: A serious Problem for Next Generation Systems", *IEEE Computer*, Vol. 21, No. 10, pp10-19, 1988. DOI: <http://dx.doi.org/10.1109/2.7053>
- [3] K. G. Shin and P. Ramanathan, "Real-Time Computing: A New Discipline of Computer Science and Engineering", *Proc. of the IEEE*, Vol. 82, No. 1, pp.6-24, 1994. DOI: <https://doi.org/10.1109/5.259423>
- [4] R. L. Burden and J. D. Faires, 2001, "Numerical analysis", 7th, Prindle Weber and Schmidt, Boston.
- [5] S. C. Chapra and R. P. Canale, 2012, "Numerical methods for engineers", 6th, McGraw-Hill.

한 창 호(Chang Ho Han)

[정회원]



- 1993년 2월 : 청주대학교 전자공학과 (학사)
- 1995년 2월 : 동 대학원 전자공학과 (석사)
- 2010년 2월 : 선문대학교 전자공학과 (박사)
- 2014년 ~ 현재 : 선문대학교 정보통신공학과 부교수

<관심분야>

영상처리, 모션캡처

이 용 민(Yong-Min Lee)

[정회원]



- 1986년 2월 : 인하대학교 전자공학과 (공학사)
- 2005년 1월 : Univ. of Edinburgh 전기전자공학 (PhD)
- 1985년12월 ~ 2007년 2월 : 삼성 SDI 수석연구원
- 2008년 3월 ~ 현재 : 선문대학교 디스플레이반도체공학과 교수

<관심분야>

디스플레이, 광통신소자 응용