

# 컴퓨터 자수용 누비기 알고리즘에 관한 연구

이양선, 이성욱  
서경대학교 컴퓨터공학과  
e-mail:yslee@seokyeong.ac.kr

## A Study on Tatami Algorithm for Computer Embroidery

Yang-Sun Lee, Sung-Oog Lee  
Dept of Computer Engineering, Seokyeong University

### 요약

의류제품에 자수를 놓기 위해서는 자수기와 이를 구동해 주는 자수 프로그램이 반드시 필요하다. 이 자수 프로그램은 밑박음질(Running), 박음질(Satin), 누비기(Tatami) 기능을 기반으로 구성되므로 가장 중요한 핵심기능이라고 할 수 있다. 본 논문에서는 베지에르 곡선식을 사용하여 구현한 누비기(Tatami) 알고리즘을 기술한다. 누비기는 범위가 넓어 박음질(Satin)로 처리를 못할 때 사용되며 다양한 무늬를 표현하거나 박음질과는 달리 넓은 범위를 채울 때 사용된다.

### 1. 서론

70년대 우리나라의 수출 주종목 가운데 하나였던 섬유업은 수출에서 차지하는 높은 비중에도 불구하고 자수기 및 자수 프로그램은 거의 외국의 기술에 의존해 왔다. 모든 의류제품에는 브랜드 상표나 마크가 부착되어 있으며 이러한 상표나 마크에는 반드시 자수가 들어간다.

의류제품에 자수(Embroidery)를 표시하기 위해서는 자수를 만들어 내는 자수기(Embroidery Machine)가 있어야 하며 이 자수기를 구동해 주는 자수 프로그램인 펀칭 프로그램(Punching Program)이 반드시 있어야 한다. 자수기는 근간에 한국 Sunstar라는 곳에서 자체 개발하여 외국 제품과 경쟁해왔으나 자수 프로그램은 독일의 윈스(WinGs), 일본의 타지마(Tajima) 제품을 사용해 왔다. 이런 이유로 국내 고급 자수 제작기술을 바탕으로 경쟁력있는 펀칭 프로그램의 개발이 필요하다.

컴퓨터 자수 프로그램인 펀칭 프로그램은 크게 밑박음질(런닝-Running), 박음질(사틴-Satin), 누비기(다다미-Tatami)의 세가지 기능으로 구성된다. 컴퓨터로 자수를 놓기 위해서는 먼저 밑박음질로 원단의 틀어짐을 방지하고 박음질을 통해 테두리부분과 거리가 짧은 부분을 디자인 내용에 따라 밀도와 수폭을 조절하여 채우고 다음에 범위가 넓은 부분을 채우기로 마무리해야 한다[10].

본 논문에서는 윈도우즈용 펀칭 프로그램의 개발과정에서 연구되어진 누비기 알고리즘을 기술한다. 펀칭 프로그램의 기능 중 기본이 되는 누비기 기능은 크게 직선 누비기와 곡선 누비기로 구분할 수 있는데 곡선 누비기는 실제 구현하는 과정에서 내부적으로 직선 누비기를 사용하고 있기 때문에 곡선 누비기에 대한 설명을 통하여 전체 자수용 누비기 알고리즘을 기술하기로 한다.

### 2. 베지에르 곡선식

자수용 프로그램에서 사용되는 박음질 및 누비기 기능은 대부분 곡선 혹은 곡면을 기준으로 작업이 이루어지기 때문에 박음질 및 누비기 기능을 구현하려면 사용자가 원하는 모양의 곡선을 자유롭게 표현할 수 있도록 하는 연구가 반드시 선행되어야 한다.

컴퓨터에서 이러한 곡선의 표현은 주로 많은 수의 작은 선분을 서로 연결하여 작성하게 되는데 이때 각각의 선분 요소를 모두 표현하는 것은 표현에 필요한 시간은 많이 소요되는 것에 비하여 표현된 모양의 자연스러움은 그에 비례하지 않는다. 때문에 단지 몇 개의 점만을 표시한 후에 매끈한 곡선을 그리는 베지에르 곡선(Bezier Curve)과 B-Spline 곡선의 두가지 방법이 널리 사용되고 있는 실정이다.

## 2.1. 베지에르 곡선

그래픽에서 곡선을 처리하는 대표적인 기법인 베지에르 곡선(Bezier Curve)[11,14,15]은 1970년대 초반에 자동차의 자동 설계에 사용할 목적으로 프랑스의 수학자 P.Bezier라는 사람에 의해 개발되었다. 이 기법은 계산이 쉽고 안정적이기 때문에 계산시간이 많이 요구되는 그래픽 프로그램에서 많이 사용된다.

베지에르 곡선은 두 끝점을 연결하며 여러개의 컨트롤 포인트(Control Point)에 의해 결정되는 곡선을 이루는 부분을 가지고 있다. 베지에르 곡선의 모든 부분에 대해 컨트롤 포인트가 영향을 주지만 B-Spline 곡선은 곡선상의 현재 위치의 가장 가까이에 있는 네 개의 점에 의해 영향을 받는다. 베지에르 곡선은 두 점간의 매우 부드러운 곡선을 제공해주는 반면 B-Spline 곡선은 컨트롤 포인트사이에 그려진 불규칙적인 라인을 보다 가깝게 따라가는 부드러운 곡선을 제공해 준다. 때문에 본 연구에서는 베지에르 곡선을 사용하기로 결정하였다.

## 2.2. 베지에르 곡선의 방정식

베지에르 곡선의 방정식은 3좌표에서 N개의 컨트롤 포인트를 주었을 때 구해지는 베지에르 곡선의 함수식이다. 여기서, 컨트롤 포인트는 베지에르 곡선을 그리기 위해 임의로 선택해준 좌표점을 말한다.

$$P(u) = \sum_{k=0}^n p_k B_{kn}(u), \quad 0 \leq u \leq 1$$

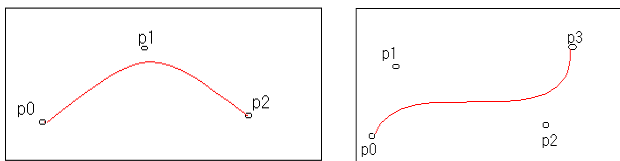
$n+1$  개의 컨트롤 포인트,  $p_k = (x_k, y_k, z_k), k=0, 1, \dots, n$

$$B_{kn}(u) = \frac{n!}{k!(n-k)!} u^k (1-u)^{n-k}$$

[그림 1] 베지에르 곡선식

3좌표  $x, y, z$ 는  $P_x(u), P_y(u), P_z(u)$  베지에르 곡선의 함수식에 의해 각각 계산된 3좌표에 의해 그려진다. 그러나 여기서는  $z$ 좌표는 고려하지 않고 단지  $x, y$ 좌표를 이용하여 곡선을 그린다. 이 함수식에서 매개변수  $u$ 는 각각의 컨트롤 포인트가 곡선을 만들 때 영향을 주는 요소이다.

다음은 위 방정식에 의해 그린 베지에르 곡선의 예이다.

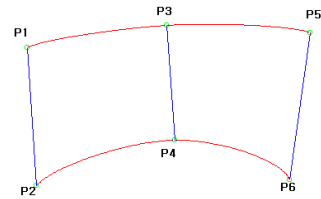


[그림 2] N=1와 N=2일 때의 베지에르 곡선

위 그림을 통해서 베지에르 곡선은 시작점과 끝점을 항상 지나지만 가운데의 컨트롤 포인트들은 지나지 않고 단지 그 위치가 곡선의 모양에 영향을 주는 것을 알 수 있다. 따라서 어떤 곡선을 표현하기 위해서는 곡선의 두 끝점을 시작점과 끝점으로 하고, 그리려는 곡선에 맞게 가운데에 컨트롤 포인트를 주면 된다.

이를 기초로 점의 좌표가 6개 일 때 베지에르 곡선을 그리는 방법은 다음과 같다. 점 6개를 찍으면 곡선이므로 첫째, 우선  $p_1, p_3$ 사이의 길이와  $p_2, p_4$  사이의 길이를 구한다. 둘째, 두 직선 중에서 큰 쪽을 기준을 잡아 큰 쪽을 밀도만큼 나눈다. 셋째, 큰 직선을 나눈 수만큼 다른 직선도 나눌 수 있도록 계산해 둔다. 넷째, 큰 직선과 작은 직선 사이를 연결하는 직선을 미리 계산해 두고 베지에르 곡선을 그려 나간다. 다섯째, 두 개의 베지에르 곡선을 그려가면서 베지에르 곡선의 포인트끼리 가상적으로 연결한 다음 네번째에서 구한 두 직선의 교점을 구하여 지금 구한 두 좌표와 전에 찾았던 좌표를 읽어서 그 사이를 영문자 N 모양으로 연결해 준다.

다음은 이와같은 방법으로 6개의 점의 좌표를 가지고 그린 베지에르 곡선의 예이다.

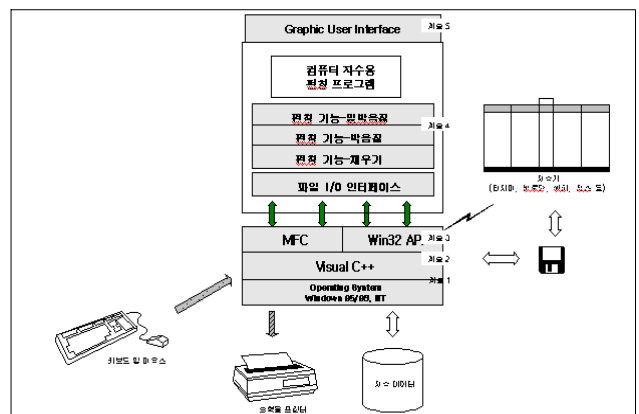


[그림 3] 베지에르 곡선의 예

## 3. 누비기 알고리즘

### 3.1 시스템 구성도

편집 프로그램의 시스템 구성도는 [그림 4]에서 보는바와 같이 PC의 윈도우즈 환경위에서 객체지향 프로그래밍 언어인 Visual C++ 언어[1,4,13]를 사용



[그림 4] 편집 프로그램의 시스템 구성도

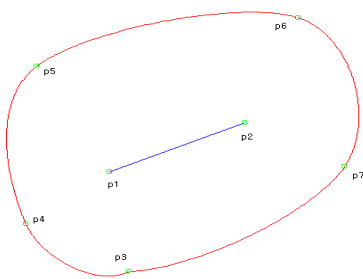
하여 개발하였으며 이 과정에서 MFC 라이브러리와 Win32 API[3,9]를 사용하였다.

### 3.2 누비기 알고리즘

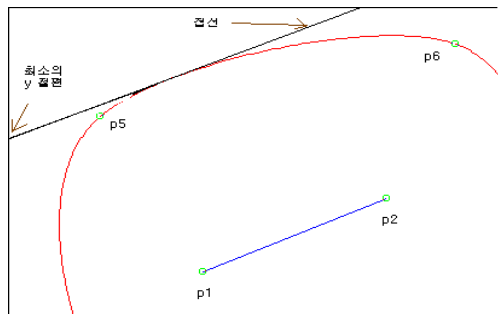
누비기(Tatami)는 범위가 넓어 박음질로 처리를 못할 때 사용되며 다양한 무늬를 표현할 때도 사용된다. 박음질과는 달리 넓은 범위를 채울 때 사용되며 실이 끊기지 않고 한번에 지정된 영역을 채울 수 있도록 처리된다.

누비기는 기본점들과 밀도를 입력받아 외곽선과 결의 기울기 및 밀도로 모양을 결정한다. 즉, 누비기는 그림을 밀도로 나눈 직선들과 외곽선과의 교점을 구한후 각점들을 순서에 따라 연결시키는 작업을 수행한다.

먼저, 누비기를 하기 위해 밀도를 정하고 결의 기울기는 처음점과 두 번째 점을 있는 직선의 기울기로 구한다. 외곽선은 세 번째 점 이후의 점들을 베지에르 식에 넣어 구한다. 그리고 외곽선을 지나는 직선들중 최소의 y 절편을 구하고 외곽선에 접하는 시작점을 구한다. [그림 5]는 기본점들을 가지고 구한 베지에르 곡선과 기울기의 예이고 [그림 6]은 최소의 y 절편값을 갖는 직선의 예이다.



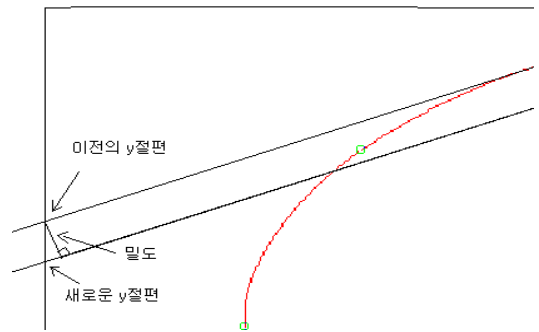
[그림 5] 기울기(p1,p2)와 외곽선



[그림 6] 최소의 y 절편 값을 갖는 직선

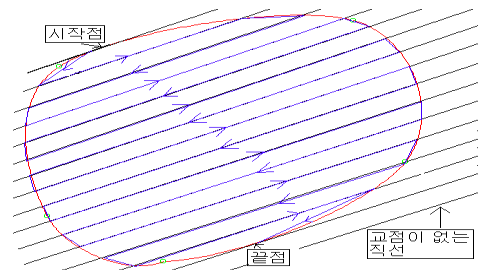
다음에 기울기와 y절편, 밀도를 이용하여 새로운 직선을 구한다. 새로운 직선은 y 절편을 지나는 접선의 법선과 접선에서 밀도만큼 떨어진 다음 직선과

의 교점을 찾아서 구한다.



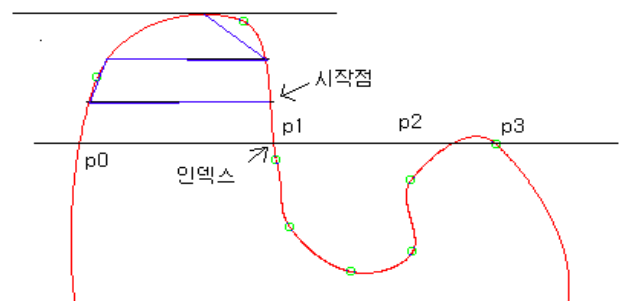
[그림 7] 법선과 다음 직선과의 교점 구하기

새로운 직선과 베지에르 곡선과의 모든 교점을 구하여 각각의 좌표를 기억한다. 그리고 이와같은 과정을 교점이 더 이상 나타나지 않을 때까지 반복한 후에 [그림 8]과 같이 누비기를 실행한다.



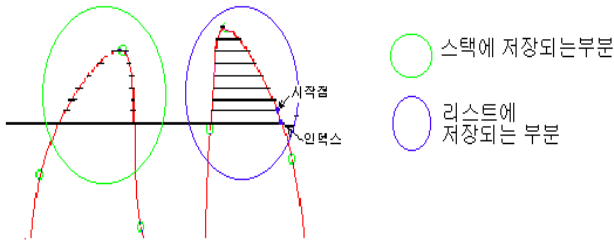
[그림 8] 교점을 이용한 누비기

그런데 베지에르 곡선과 밀도로 나눈 직선과의 교점이 4개 이상되는 경우 다음과 같은 점을 고려해야 한다. 첫째, 교점이 증가하는 경우 인덱스를 이용하여 인덱스가 속한 쌍을 선택해서 리스트에 저장하고 나머지를 임시 저장용 스택에 저장한다. 즉, [그림 9]에서 p0, p1 쌍은 리스트에 저장되고 p2, p3 쌍은 스택에 저장된다.



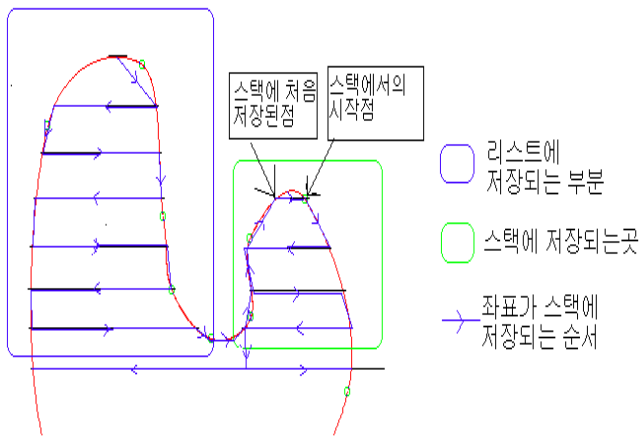
[그림 9] 교점이 증가하는 경우

둘째, 교점이 같은 경우 인덱스가 위치한 쌍은 리스트에 저장하고 다른 쌍은 임시 저장용 스택에 저장한다.



[그림 10] 교점이 같은 경우

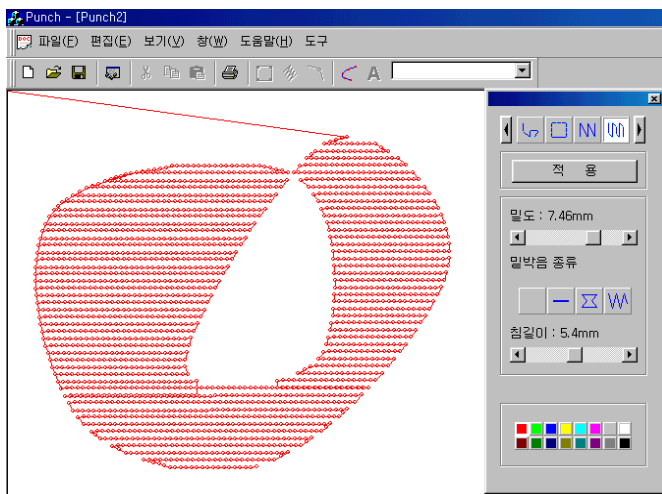
세제, 교점이 감소하는 경우 스택의 초기값을 찾는다. 스택의 자료를 이용하여 좌표를 생성하고 이것을 순서에 맞추어 리스트에 입력한다.



[그림 11] 교점이 감소하는 경우

다음에 리스트에 저장된 순서대로 누비기를 수행한다.

다음은 위의 알고리즘으로 누비기를 수행한 예이다. [그림 12]는 누비기할 곳의 외곽선을 입력시킨후에 그 사이점들을 주어진 값인 밀도로 분할하여 면을 채운 모습이다.



[그림 12] 누비기

#### 4. 결론

의류제품에 자수를 표시하기 위해서는 자수 프로그램인 펀칭 프로그램이 반드시 있어야 한다. 그러나 섬유산업의 사양화로 국내 자수 프로그램 업체가 2-3군데 밖에 안되고 원천기술이 부족하여 국내에서는 거의 독일의 wings(Wings)와 일본의 타지마(Tajima) 제품을 사용해 왔다.

이런 이유로 본 논문에서는 순수 국내 고급 자수 제작 기술을 바탕으로 윈도우용 펀칭 프로그램의 개발을 통해 원천기술을 확보하고 외국 제품의 국내시장 잠식을 막기 위해 펀칭 프로그램의 기본형을 개발하였다. 본 논문에서는 이 펀칭 프로그램의 중요한 기능중 하나인 누비기 알고리즘을 기술하였다. 앞으로 매끄럽고 섬세한 고급 자수걸을 제작하기 위한 기능 향상이 필요하다.

#### 참고문헌

- [1] B.Stroustrup, The C++ Programming Language, Prentice-Hall, 1996.
- [2] E.Horowitz,S.Sahni,D.Mehta, Fundamentals of Data Structure in C++, CSP, 1992.
- [3] J.Conger, Windows API Bible, The Waite Group, 1992.
- [4] J.Martin & J.Odell, Object-oriented Analysis & Design, Prentice-Hall, 1992.
- [5] L.Heiny, Windows Graphics with Borland C++, John Wiley & Sons, 1994.
- [6] Pohl, Object-oriented Programming using C++, Addison-Wesley, 1996.
- [7] P.S.Wang, C++ with Object-oriented Programming, PWS, 1994.
- [8] R.Simon, WIN 32 API 슈퍼 바이블 상,하, 정보문화사, 1998.
- [9] Tajima Industires Ltd., DG/ML For Windows User Manual, 1999.
- [10] Viking Sewing Machines AB., Professional Embroidery System User's Guide, 1999.
- [11] V.B.Anand, Computer Graphics & Geometric Modeling for Engineers, John Wiley & Sons, 1992.
- [12] 사재학, 시스템 분석과 설계, 옹보, 2000.
- [13] 송호중, MFC 4 바이블, 대림, 1997.
- [14] 유희용, 베지에르 곡선의 컨트롤포인트 찾기, 프로그램 세계 4월호, 1997.
- [15] 임용식, C 그래픽 연습, 크라운 출판사, 1993.