

# 폰트 데이터 기반의 동양적 붓글씨 필적 생성

안정호, 이인권  
연세대학교 컴퓨터과학과  
macro516@cs.yonsei.ac.kr, iklee@yonsei.ac.kr

## Font Data-driven Oriental Brush-Art Calligraphy Generation

Jeong-Ho Ahn, In-Kwon Lee  
Dept. of Computer Science, Yonsei university

### 요 약

이 논문에서는, 기존에 존재하는 글자체의 커브 데이터를 분석하여 같은 글자를 붓글씨로 서예를 하듯이 다시 써낸 듯한 효과를 낼 수 있는 방법을 제안한다. 글자를 형성하는 위상적인 뼈대를 커브로 쪼개어, 글자 하나를 여러 획으로 분리하여 표현한 후에, 각 획에 해당하는 커브의 차원 수와, 길이와 곡률을 이용하여 붓의 궤적을 자동적으로 생성해 내는 방법이다. 붓의 궤적이 표현될 방법을 기존 글자 데이터를 이용해서 어떻게 조작 경로를 자동적으로 만들어 붓글씨 필적을 생성해낼 것인지가 풀어야 할 문제이다.

### 1. 서 론

서예는 본래 고대 중국에서 발달하여 한자를 사용하는 한국, 일본, 베트남 등 여러 나라에 계승·발달하였고, 예술의 한 영역으로 자리잡고 있다. 서예는 다음과 같은 특징이 있다.

- 먼저 글자를 쓰는 것으로써 서예술이 성립된다.
- 점과 선의 구성과 비례 균형에 따라 공간미가 이루어진다.
- 필순, 즉 시간의 흐름에 따라 형성된다.
- 필순에 따른 운필의 강약 등으로 율동미가 전개된다.
- 자연의 구체적인 사물을 그리는 것이 아니라 글자라는 추상적인 것을 소재로 한다.
- 먹은 옛날부터 오채(五彩)를 겸하였다고 하며 검정색이지만 농담(濃淡)·윤갈(潤渴)·선염(渲染)·비백(飛白) 등이 운필에 따라 여러 색을 사용하는 것과 같은 영묘(靈妙)한 결과를 낳는다.

그리고 다음은 중국 고금의 첫째가는 서예가로 존경받고 있는 진(晉)나라 왕희지의 말이다.

“글씨를 쓰려고 하는 자는 먼저 먹을 갈면서 정신을 집중시키고 마음을 가라앉힌 다음, 글씨의 자형을 미리 생각하여 크고 작은 획과 아래나 위로 향한 획과 평평하고 바른 획들을 움직여 맥이 서로 연결되게 구상한다. 이것은 뜻이 붓보다 먼저 있게 한 연후에 글씨를 쓰는 방법이다. 만약 평평하고 바른 획이 마치 주판알 같이 위아래가 반듯하고 앞뒤가 고르면 글씨가 되지 않고, 단지 점과 선만 연결시켜 놓은 꼴이 된다.”

이를 통해 서예의 미는 일관적인 규칙을 통해 나타내어진다는 사실에 주목할 수 있다. 글자는 점, 가로획, 세

로획, 삐침, 갈고리 등의 일정한 형태를 갖는 획이 균형성과 방향성, 그리고 규칙성을 갖고 구성된다. 그리고 붓을 사용할 때, 획을 그리기 시작하고 보내고 끝내는 데에도 일정한 규칙이 존재하고 있다. 우리는 이러한 규칙들을 계산적인 방법으로 자동으로 생성하는 방법에 대해 주목하였다.

기존의 동양적 붓글씨를 생성하는 연구들은 3차원 붓 모델 기반 방법과 서예 이미지 기반 방법으로 구분지을 수 있다. 3차원 붓 모델 기반 방법은 실제 붓을 3차원으로 모델링하여 그리고자 하는 면과 만났을 때, 붓털이 렌더링 평면과 만나는 영역을 칠하는 방법들이고, 서예 이미지 기반 방법은 실제 서예 이미지의 데이터를 이용해 글자를 만들어 내는 연구들로 분류할 수 있었다. 하지만 우리 연구에서는 3차원 붓 모델이나 기반 이미지 없이, 단순한 경로만을 이용해 더 간단하고 효과적으로 붓글씨를 생성해내고 있다.

섹션 2에서는 관련 연구에 관한 설명을 하면서 본 연구와의 차이점을 발견하고, 섹션 3에서는 자세한 알고리즘을 순서대로 설명할 것이다. 그리고 섹션 4에서는 실험결과를 보여주고, 섹션 5에서는 그에 따른 결론을 도출한다.

### 2. 관련 연구

본 논문과 관련된 연구는 크게 붓글씨 표현에 관한 연구와 붓글씨 분석에 관한 연구로 나눌 수 있다.

붓글씨 표현에 관한 연구는 주로 붓 모델을 3차원 상에서 생성한다. 붓이 힘을 받아 종이에 눌러져 젖을 때, 부드러운 붓의 털이 어떻게 휘는지 생성하고, 그 궤적의 부자연스러운 부분은 제거하는 방법[1]이 있었다. 그리

고 붓의 털들을 매스스프링(mass-spring) 시스템으로 연결하여 실제와 흡사한 붓털의 움직임을 표현한 방법 [2]이 있었다.

붓글씨 분석에 관한 연구는 주로 기존 붓글씨들을 보고, 어떠한 형태로 3차원 붓을 작동시켜야 기존 붓글씨와 같이 써질 것인가를 분석[4]하고 있다.

[3]에서는 서예붓으로 나무를 그려내는 수묵화 표현법을 제안하고 있으며, [5]에서는 본 연구에 필요한 트루타입폰트에서의 획 검출 알고리즘을 제안하고 있다.

우리 연구는 [5] 방법과 같이 획을 검출해내어, 그 궤적을 따라 3차원 붓 모델의 시뮬레이션 없이 붓 효과를 내게 된다.

### 3. 알고리즘

#### 3.1. 획 검출

아스키(ASCII)코드에 해당하는 글자들의 폴리라인(poly-line)들의 리스트를 B-spline 커브들의 리스트로 바꾸는 작업을 통해 획을 검출해 내었다. 한 폴리라인 내부에서 90도 이상 급격히 꺾이는 부분에서는 폴리라인을 나누는 작업을 해주었다. 직선으로 검출되는 획은 오더-2의 B-spline으로 만들었고, 곡선으로 검출되는 획은 오더-3의 B-spline으로 만들었다.

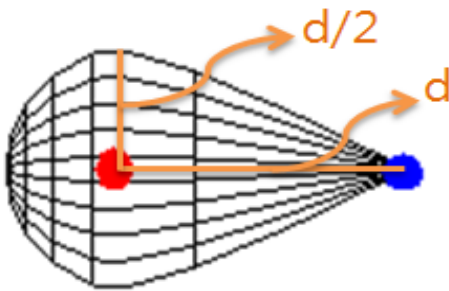


그림 1 붓이 종이에 남기는 흔적을 붓의 중심과 붓끝을 이용해 표현한 면이다. 빨간 점이 붓의 중심을 의미하며, 파란 점은 붓끝을 의미한다. 빨간 점이 붓의 궤적을 따라 이동하면, 파란 점은 빨간 점에 매달려 쫓아간다.

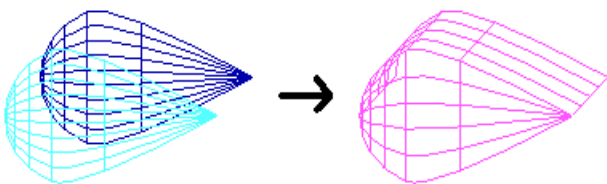


그림 2 붓이 이동하게 되면 찍히는 면들이 통합된다.

#### 3.2 궤적면 생성

붓이 경로를 따라 이동하면서 남기는 궤적은 [그림1]과 같은 면들을 모두 irit[6]의 면통합 알고리즘을 통해 [그림2]처럼 합쳐서 만들게 된다. 너비는 붓중심-붓끝 거리  $d$ 의 0.5배로 하였다.

#### 3.3 궤적면 변환 알고리즘

##### 3.3.1 붓중심의 움직임

우리는 획의 시작에는 붓을 종이에 갖다 대는 과정이 있어 속도가 느리고, 획의 진행에는 붓을 옮기면서 속도가 나고 획이 얇아지며, 획의 끝에는 붓을 떼는 과정에서 속도가 늦어진다는 가정하에 붓의 궤적을 모델링 하였다. 붓은 속도에 반비례하게 [그림3]과 같이 더 넓은 흔적을 남긴다.

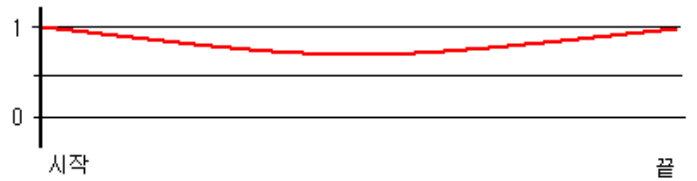


그림 3 획의 시작 -> 흐름 -> 끝에 있어 붓의 흔적의 너비는 그림과 같이 변한다.

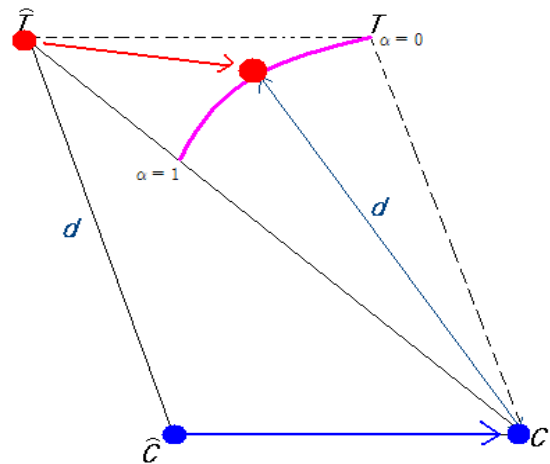


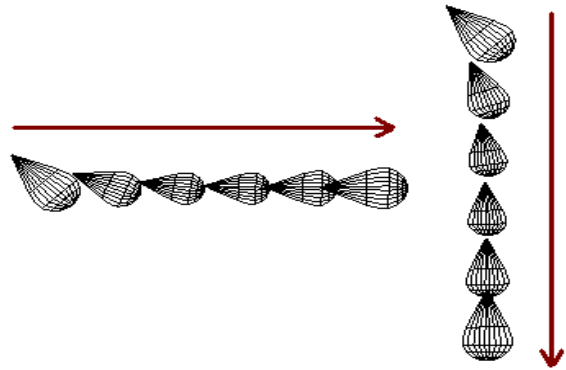
그림 4 파란색은 붓중심의 이동을 나타내고, 빨간색은 붓중심의 이동에 따르는 붓끝의 움직임을 나타낸다.

##### 3.3.2 붓끝의 움직임

붓끝은 직선을 그을 경우 좌측 상단에서 시작하며, 곡선을 그을 경우엔 곡선진행방향의 반대에서 시작한다. 붓중심이 이동하면 붓끝은 붓중심과의 간격을 유지하기 위해 따라서 이동하게 된다. 수식은 다음과 같다.

$$T(\alpha) = (d * \text{normalize}(\hat{T} - (1 - \alpha)\hat{C} - \alpha C)) + C \quad (1)$$

위의 식에서  $T$ 는 갱신될 붓끝의 위치,  $\hat{T}$ 은 이전 붓끝의 위치,  $C$ 는 갱신된 붓중심의 위치,  $\hat{C}$ 은 이전 붓중심의 위치,  $d$ 는 붓끝-붓중심 사이의 고정된 거리이고,  $\alpha = [0,1]$ 는 붓끝과 붓중심이 이루는 각도를 유지하는 계수이다.  $\alpha$ 가 커질수록 붓끝이 더 큰 관성을 갖게 된다. [그림4]에서 (1)식에 대한 그림을 볼 수 있다. 붓 중심이  $\hat{C}$ 에서  $C$ 로 이동하게 되면, 붓 끝은 붓 중심과의 거리  $d$ 를 유지하기 위해,  $\hat{T}$ 에서  $T$ 로 따라오게 된다.



#### 4. 실험결과

실험에는 Gershon Elber의 irit[6]이 이용되었다. irit은 사용하기 간단하고 유용한 기능을 가진 모델링 툴이다.

우리는 가장 좋은 실험 결과를 내기 위해 폰트에서 추출한 커브는 100개의 구간으로 나누어 계산하였고, 붓끝의 관성을 나타내는  $\alpha$ 값은 0.7으로 고정하였다.

먼저 가로획과 세로획을 긋는 경우, [그림5]와 같은 결과가 나타난다. 시작 부분에서 붓의 끝부분이 좌측 상단을 향하고 있으며, 붓중심의 이동 궤적을 따라 붓끝이 따라가는 것을 관찰할 수 있다. 또한 획의 시작과 끝 부분의 붓의 면적이 궤적 중간부분의 붓의 면적보다 넓은 것을 볼 수 있다.

우리의 연구 결과는 어떠한 문자라도, 획을 구성하는 커브를 추출해 낼 수 있다면, 적용이 가능하다. 영문 텍스트 "GeometricModeling"의 폰트 데이터에서 획을 추출하여 붓글씨를 생성한 결과가 [그림6]이다.



그림 5 본 연구에서 제시한 알고리즘을 이용해 만든 가로획과 세로획.

# Geometric Modeling

그림 6 "GeometricModeling"이라는 텍스트로 알고리즘을 돌린 결과 얻어지는 붓글씨 모양의 면들.

## 5. 결 론

우리는 폰트 데이터의 위상적인 뼈대를 커브로 표현하여, 그 경로만을 이용해 2차원으로 동양적인 붓글씨를 효과적으로 생성해내는 방법을 제안하였다.

이 논문에는 3가지의 제한점이 존재한다. 첫 번째, 텍스트 커브가 호의 길이에 따른 파라미터화(arc-length parametrization)가 되지 않아서, 브러쉬의 너비가 커브 길이의 다양함에 대해서 자유로울 수 없었다. 두 번째, 면을 통합할 때, irit[6]의 면통합 알고리즘을 이용해서 붓글씨를 만들게 되면, 필요없는 고정점이 너무 많이 포함되어 버려서, 실행속도가 현저히 느려졌다. 궤적의 테두리에 해당하는 고정점만을 이용하여 붓글씨를 생성하는 방법이 필요하다. 세 번째, 연결된 획 간에 붓끝이 튀어나가는 현상이 발생하였는데, 획간의 연결관계를 파악해서, 나중에 굵는 획의 붓끝은 이전 획 안에 위치하도록 설정한다면 그런 현상을 방지할 수 있을 것이다.

이번 실험에서는 irit에서 제공하는 알파벳에 관한 poly line정보를 가지고, 커브를 만들어 실험에 임하였는데, [5]등의 내용을 이용하여, 실제 트루타입폰트 커브데이터에서 획을 분할, 검출해내는 과정을 추가할 수 있을 것이다.

## 6. 참고문헌

- [1] Jintae Lee, Simulating oriental black-ink painting, IEEE Computer Graphics and Applications, Vol. 19, Issue 3, pp. 71-81, 1999.
- [2] Chu N.S.H and Chiew-Lan Tai, Real-time painting with an expressive virtual Chinese brush, IEEE Computer Graphics and Applications, Vol. 24, Issue 5, pp. 76-85, 2004.
- [3] Qing Zhang, Youetsu Sato, Jun-ya Takahashi, Kazunobu Muraoka and Norishige Chiba, Simple cellular automation-based simulation of ink behaviour and its application to Suibokuga-like 3D rendering of trees, THE JOURNAL OF VISUALIZATION AND COMPUTER ANIMATION, J. Visual. Comput. Animat. 10, pp. 27-37, 1999.
- [4] Helena T.F.Wong and Horace H.S.Ip, Virtual brush: a model-based synthesis of Chinese calligraphy, Computers & Graphics, Volume 24, Issue 1, pp. 99-133, 2000.
- [5] 광윤석, 구상욱, 정순기, 한글 트루타입폰트 및 손글씨의 자동 획 분할 알고리즘, 한국정보과학회 학술발표 논문집, pp. 275-280, 2008.
- [6] <http://www.cs.technion.ac.il/~irit/>