

Bezier curve를 이용한 3차원 곡선의 애니메이션화

본 연구에서는 3차원의 임의의 이미지를 3차원 곡선으로 근사하기 위해 ‘개선된 베지어 곡선’을 정의하였고, 이 후 곡선에 움직임을 주어 애니메이션을 구현하는 새로운 방법을 고안하였다. 우리는 공간상에서 임의의 이미지에서 색깔, 명암이 많이 변화하는 곡선을 기준으로 곡선 길이에 따라 이등분하여 소구간을 정의하였다. 각 구간에서 베지어 곡선을 그리기 위해 구간의 ‘중간점’을 구간의 양 끝점을 이은 직선과의 거리가 최대인 점으로 정의하여 중간점과 구간의 양 끝점을 이용하여 만들어진 베지어 곡선과 기존의 곡선과의 오차율을 계산하였고, 이 오차율이 기준값 이하가 되는 과정을 반복하도록 설정하였으며, 근사의 정도를 판단하는 오차율의 기준은 선행연구와 같이 하우스도르프 거리를 이용하였다. 선행 연구에서 2차원 베지어 곡선의 움직임을 주는 방법을 본 연구에서는 3차원으로 확장시켜 3차원 곡선에 대해서도 움직임을 주는 것이 가능함을 알 수 있었다. 이를 processing으로 구현하였고, 근사 결과 4번의 구간 분할만에 오차가 0.001보다 작게 근사된 결과를 보였다. 이로써 본 연구에서 정의한 베지어 곡선으로 근사시키는 방법이 선행연구보다 효율이 매우 높다는 점과 충분히 좋은 근사가 가능하다는 것을 보여주었다. 본 연구의 방법을 통해 3차원 곡선을 표현할 수 있기 때문에 홀로그램이나 VR, 자율주행 등 활용 가능성이 있으며 베지어 곡선을 구성하는데는 세 개의 점만 필요하기 때문에 저장용량이 효율적으로 감소될 것이라 기대된다.

1. 서론

영화 또는 애니메이션을 보면 부드럽고 자연스러운 곡선이 많이 등장한다. 애니메이션의 많은 부드러운 곡선들은 베지어 곡선을 통해 구현된 것이다.[1] 어떤 곡선을 수학적 도구를 이용해 나타낼 수 있다면 이를 수학적으로 조작함으로써 다양한 컴퓨터 그래픽으로의 활용이 가능하다. 하지만 선행 연구에서는 2차원에서의 곡선을 근사하는 연구만 이루어졌다. 만약에 2차원 곡선을 베지어 곡선을 통해 근사하는 방법을 3차원 곡선으로 확장 시킨다면 3차원에서의 곡선의 움직임을 구현할 수 있다. 우리는 이 3차원 곡선을 통해 우리가 2차원으로서만 보고 이해하기 어려운 곡선이나 그림들을 3차원으로 표현함으로써 홀로그램 등 최근 주목받는 IT 기기들의 혁명적인 발전에 기여할 수 있을 것이라 기대된다.

임의의 공간상의 곡선을 각 구간별로 분할 한 뒤, 각 구간에서 한 개의 2차 베지어 곡선을 정하기 위해서는 세 개의 점이 필요하다. 그리고 이 세 개의 조절점을 어디에 잡는지에 따라 무수히 많은 곡선이 가능하다. 하지만 그 중에서도 본 연구에서는 공간상의 임의의 곡선을 베지어 곡선으로 근사하면서 원래 곡선과 오차가 적은 곡선을 적절히 고르는 것이 목표이다. 그리고 곡선을 그린다고 하여도 그런 곡선들의 모임이 매끄럽게 이어지지 않는다면 오차가 작더라도 원래 어색하게 보일 수 있으므로 곡선끼리 이을 때도 어떻게 해야 부드러운 곡선이 나올지에 대해 충분히 생각해 봐야 한다. 그러므로 우리는 세 개의 점을 어떻게 잡아야 임의의 연속인 곡선과의 오차를 최소화할지, 또한 그 곡선들이 매끄럽게 이어지기 위해서 어떤 기준으로 점을 잡아야 할지 동시에 생각해 보아야 한다.

본 연구에서는 이 질문에 답하기 위해 기존 선행

연구에서 사용되던 고차 베지어 곡선 대신 2차 베지어 곡선을 선택하였다. 2차 베지어 곡선은 고차 베지어 곡선에 비해 곡선을 구성함에 있어 필요한 조절점의 개수가 가장 적은 장점을 가지고 있다. 따라서 2차 베지어 곡선을 이용하여 임의의 3차원 곡선을 효율적으로 근사할 수 있는 방법을 제안한다.

2. 본문 내용

2.1. 이론적 배경

2.1.1. 2차 베지어 곡선

다음의 다항식을 2차 베지어 다항식이라고 한다.

$$B(t) = (1-t)^2P_0 + 2t(1-t)P_1 + t^2P_2$$

그리고 $t \in [0, 1]$ 에서 2차 베지어 다항식이 나타내는 곡선을 2차 베지어 곡선이라 한다.[1]

이때 세 점 P_0, P_1, P_2 을 이 2차 베지어 곡선의 조절점이라 한다.

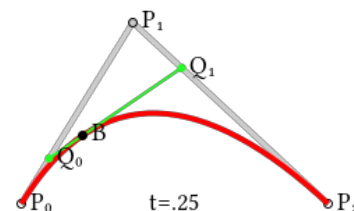


Fig 1. 2차 베지어 곡선($t = 0.25$)

2.1.3. 하우스도르프 거리

거리공간 (M, d) 의 공집합이 아닌 부분집합 X, Y 에 대해 하우스도르프 거리는

$$d_H(X, Y) = \max(\sup_{x \in X} \inf_{y \in Y} d(x, y), \sup_{y \in Y} \inf_{x \in X} d(x, y))$$

와 같이 정의한다. [2]

2.2. 곡선을 조각별로 나누기

주어진 곡선을 하나의 2차 베지어 곡선으로 근사하면 원래 곡선과의 차이가 커질 수 있다. 곡선을 근사하기 위해 선택한 방법은 곡선을 조각별로 나누는 것이다. 연속함수 $\gamma: [0, 1] \rightarrow \mathbb{R}^3$ 에 대해서 곡선이 $C = \gamma([0, 1])$ 로 정의될 때 $\gamma(t)$ 의 정의역을 2^n 등분한다. 즉 다음과 같다.

$$\gamma_i(t) = \gamma\left(\frac{i+t}{2^n}\right) \quad (i = 0, 1, \dots, 2^n - 1)$$

2.3. 곡선을 근사하기

각 조각을 근사하기 위한 방법은 다음과 같다.

$$P_0 = \gamma_i(0), P_2 = \gamma_i(1)$$

이유는 곡선과의 차이를 줄이기 위해, 그리고 각 조각을 근사한 곡선들이 연속적으로 이어지기 위함이다.

$\gamma_i(t)$ 에서 $\gamma_i(0)$ 과 $\gamma_i(1)$ 을 잇는 직선에 내린 수선의 길이가 최대가 되게 하는 t 의 값을 $t_{1/2}$ 라 하자. 이제,

$$P_1 = 2\gamma_i(t_{1/2}) - \frac{\gamma_i(0) + \gamma_i(1)}{2}$$

$$\beta_i(t) = (1-t)^2 P_0 + 2t(1-t) P_1 + t^2 P_2, \quad B_i = \beta_i([0, 1])$$

이라 하자.

2.4. 곡선의 근사 정도 확인하기

각 조각에 대한 근사 거리와 전체 곡선에 대한 근사 거리를 다음과 같이 정의한다.

$$\mu_i = d_H(C_i, B_i), \quad \mu = \max\{\mu_i \mid i = 0, 1, \dots, 2^n - 1\}$$

이제 주어진 양수 ε 에 대해 $\mu < \varepsilon$ 이면 충분한 근사가 되었다고 판단하고, 과정을 종료한다. [3]

만약 $\mu \geq \varepsilon$ 이라면 n 대신 $n+1$ 에 대해 2.2.2로 돌아간다.

2.5. 곡선의 움직임

곡선의 움직임을 나타내기 위해서는 두 곡선이 필요하다. 두 곡선 C_0, C_1 를 각각 충분히 조각별로 나누고, 각각의 서로 대응되는 점 b_0, b_1 에 대해 점 $b_t = b_0 + t(b_1 - b_0)$ 을 얻는다. 그리고 $t \in [0, 1]$ 에 대해 b_t 들을 이용해 베지어 곡선을 그린다. [4]

이 방법을 사용하면 곡선에 부드러운 움직임을 줄 수 있다. 그리고 $t = \frac{k}{n}$ 꼴로 선택하면 초당 프레임을 정해 직접 애니메이션으로 구현할 수 있다.

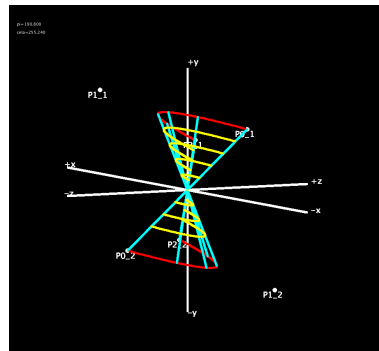


Fig 2. 처음과 끝 곡선(빨간색), 중간 곡선(노란색)

3. 결론

우리는 이와 같은 방법으로 2차 베지어 곡선을 구성하는 데 필요한 조절점을 얻고, 3차원 곡선을 근사할 수 있다. 또한 근사한 곡선을 이용하여 3차원 곡선의 움직임을 줄 수 있게 되었다. 함수식이 주어지지 않은 3차원 곡선의 경우 2차 베지어 곡선을 이용하면 조각별로 3개의 점만으로 구현할 수 있다는 점에서 가공할 수 있는 곡선의 대상이 무한하다는 점과 함께 이에 따라 곡선을 저장하는데 필요한 용량은 효율적으로 줄어든다.

그리고 이를 2차원 곡선뿐만이 아닌 3차원 곡선을 근사시키는 과정에서 더 활용 가능성을 높일 수 있다. 3차원 곡선에 움직임을 주는 방법은 홀로그램이나 VR 기술 등 미래에 상용화될 것이라고 기대되는 많은 기술들을 구현함에 있어 효과적으로 도움이 될 것이라고 예상된다.

참고문헌

- [1] Hazewinkel, Michiel (1997). Encyclopaedia of Mathematics: Supplement. 1. Springer Science & Business Media. p. 119.
- [2] Munkres, James (1999). Topology (2nd ed.). Prentice Hall. pp. 280–281.
- [3] Seon-Hong Kim & Young Joon Ahn (2007) "An approximation of circular arcs by quartic Bezier curves", Computer-Aided Design 39, pp.490-493.
- [4] 문성룡, 문홍진, 송의남, 김종교 (1985). Quadratic Bezier Polynomial 과 B-Spline Function을 이용한 Curve Fitting 및 Computer Graphic Animation 에 관한 기초연구. 대한전자공학회 학술대회, 321-327