

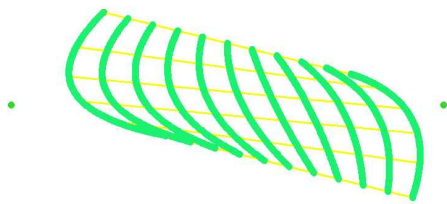
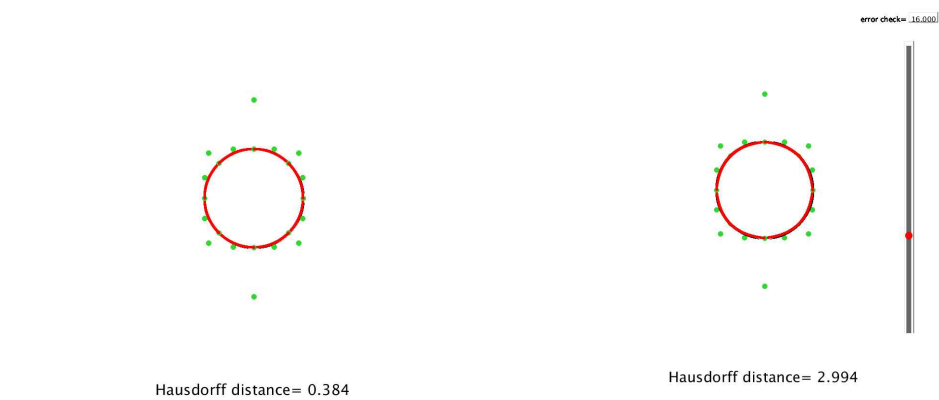
학번 : 20046

이름 : 박건호

제목 : Bezier curve를 이용한 원호근사와 모션을 구현하여 애니메이션화 시키기

만든 것에 대한 설명(결과 그림 캡처 후 삽입)

- 관련 수학/과학 교과명 : 수학(R&E)
- 관련 내용 또는 주제 : Bezier 곡선, Hausdorff distance
- 만든 processing 시뮬레이션 결과 캡처 사진 (이 기능들 모두 있습니다)



만든 것의 과학/수학/정보 관점의 의미

베지어 곡선은 픽사나 다른 여러 회사에서 영화를 만들 때 쓰는 곡선으로 곡선을 되게 부드럽게 보이게 해주는 곡선이다.

이 코드는

1. 어떤 원에 대하여 이차함수의 일종인 베지어 곡선을 이용하여 근사시키는 것이다
이때 근사시키는 척도는 하우스도르프 거리이다.
2. 어떤 모션을 줄 곡선의 첫 번째 곡선과 마지막 곡선을 입력하면 내분비를 이용하여
자연스럽게 움직이는 효과를 낼 수 있으며 이를 활용하면 선의 움직임 뿐만이 아니라
면을 채우는 효과까지 만들 수 있다.
3. 2번 과정에서 채우고 지우는 함수가 따로 있는데 이에 대하여 채우는 곡선의 굵기와
지우는 곡선의 굵기를 다르게 한다면 3D 프린팅을 하는 효과까지 만들 수
있다.(명암까지 포함시켜서)

수학적 과학적 원리에 대한 설명

이차 베지어 곡선(quadratic Bezier curve)이란?

세 점 P_0, P_1, P_2 와 $0 \leq t \leq 1$ 에 대하여

$\overline{P_0P_1}$ 을 $(1-t):t$ 로 내분하는 점을 Q_0 , $\overline{P_1P_2}$ 를 $(1-t):t$ 로 내분하는 점을 Q_1 이라 할
때,

이차 베지어 곡선 위의 한 점 $B(t)$ 는 $\overline{Q_0Q_1}$ 을 $(1-t):t$ 로 내분하는 점이다.

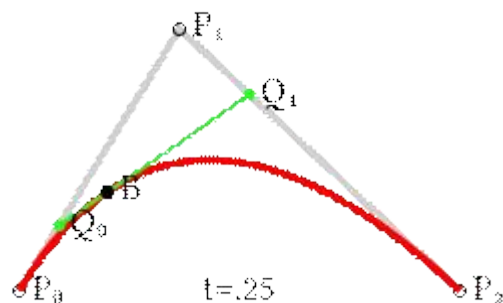
이 때 세 점 P_0, P_1, P_2 를 베지어 곡선의 조절점(control point)라 한다.

식으로는 다음과 같이 나타나진다 $B(t) = (1-t)^2P_0 + 2t(1-t)P_1 + t^2P_2$

$$Q_0 = (1-t)P_0 + tP_1$$

$$Q_1 = (1-t)P_1 + tP_2$$

$$\begin{aligned} B(t) &= (1-t)Q_0 + tQ_1 \\ &= (1-t)[(1-t)P_0 + tP_1] + t[(1-t)P_1 + tP_2] \\ &= (1-t)^2P_0 + 2t(1-t)P_1 + t^2P_2 \end{aligned}$$



$$B(t) = (1-t)^2P_0 + 2t(1-t)P_1 + t^2P_2, t = 0.25$$

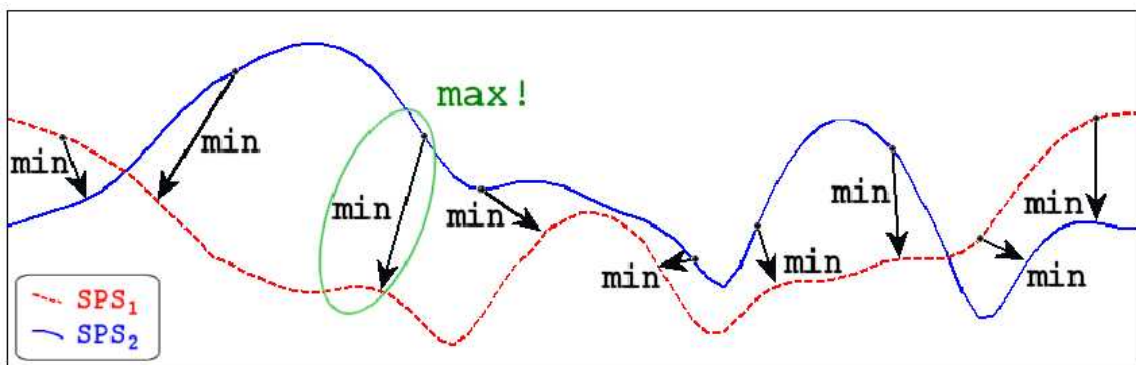
P_0, P_1, \dots, P_n 을 조절점으로 가지는 베지어 곡선은 다음 식으로

$$\text{나타내어진다. } B(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i P_i$$

하우르도르프 거리(Hausdorff distance)란?

X, Y 가 \mathbb{R}^n 의 공집합이 아닌 부분집합일 때, 하우스도르프 거리

$$d_H(X, Y) = \max \left(\max_{x \in X} (\min_{y \in Y} d(x, y)), \max_{y \in Y} (\min_{x \in X} d(x, y)) \right) \quad [2]$$



이런 이론들을 사용하여 다양한 임의의 곡선을 근사시킨다면 되게 효율적일 것이다.

베지어 곡선을 이용하여 근사하는 방법을 정리하면

1. 구간을 이등분한다.
2. 양끝점을 정의한다.
3. 중간점을 양끝점에 대해서 결정되는 좌표계로 바꾸어 원래 곡선의 중간점의 x, y, z 성분을 2배하면 구할 수 있다.

점 P_0 을 $(-a, 0)$, 점 P_2 를 $(a, 0)$ 이라고 하자. 그리고 위의 베지어 곡선 식에 대입하고 $t=0.5$ 를 넣으면? 베지어 곡선의 중간점의 각 성분의 절반을 한다면 원래의 곡선의 중간점이 나온다는 것을 확인 할 수 있다.

4. 베지어 곡선의 양끝점은 원래 곡선의 양끝점과 동일하므로 똑같이 정의한다.
5. 베지어 곡선을 그리고 원래 곡선과의 하우스 도르프 거리를 구해 임의의 값을 넘기면 다시 1번으로 돌아가고 아니면 이 과정을 끝낸다.

모션을 주기 위해서는?

양 끝 모션의 곡선을 c_1, c_2 라고 하고 그 중간의 모션의 곡선을 $c_1 + (c_2 - c_1) * t$ 라고 한다면 t 가 0부터 1까지 변할 때 자연스러운 과정으로 c_1 부터 c_2 까지 변하는 것을 관

잘 할 수 있다.

그리고 만약에 앞의 곡선과 뒤의 곡선을 일정한 비율로 내분하는 점들을 모아 선을 만든뒤 내분비를 다르게 모션으로 나타내준다면 앞의 그림 중 오른쪽 아래에 있는 사진처럼 모션이 나타난다(모션이 바뀌는걸 넣고 싶은데 사진이여서 못넣었어요 move 라는 함수를 작동시키면 작동합니다.)

circles() 3차원 예시 1

spiral_curve() 3차원 예시 2

coordinate3D() 3차원 좌표계

check3D(float x,float y,float z) (x,y,z)가 (x,y)에 대응되도록 하는 함수

point3D(float x,float y,float z) (x,y,z)가 찍힌다.

point3D_RGB(float x,float y,float z,float R,float G,float B,float st) (x,y,z)가 색깔 굵기 까지 설정되도록 그리기

Bezier_line3D(float x0,float y0,float z0,float x1,float y1,float z1,float x2,float y2,float z2,float x3,float y3,float z3,float x4,float y4,float z4,float x5,float y5,float z5)

베지어 곡선의 각 점이 움직이는 경로 표시

line3D_RGB(float x,float y,float z,float w,float m,float n,float R,float G,float B) 선의 색깔도 결정하여 3차원 좌표계에서 선을 그린다.

Bezier_move3D(float x0,float y0,float z0,float x1,float y1,float z1,float x2,float y2,float z2,float x3,float y3,float z3,float x4,float y4,float z4,float x5,float y5,float z5)

베지어 곡선이 움직이는 경로 표시

Bezier3D(float x0,float y0,float z0,float x1,float y1,float z1,float x2,float y2,float z2, float num)

3차원 베지어 곡선 그리기

makeBezier(float start,float last,float r,float n)

이 코드를 작동시키는 방법 (함수가 많아서 설명을 넣었습니다)

bar(800,100,800):(두번째 사진에 있는 막대기 한번 누르셔야지 작동합니다)

오차율을 체크하여줍니다

아니면 제일위에 있는 변수인 app의 수를 바꿔주면 더 정밀한 근사가 나옵니다

circle2D(100,0,0,360):(0도부터 360도까지 반지름이 100인 원을 그리고 0번 나눈겁니다)

bezier2D(300,100,450,50,600,100,1):(첫번째점 중간점 끝점, 곡선 번호)순서로 넣으면 곡선이 만들어 집니다

Housedorf(100,220,320,300,100,450,50,600,100):(원과 곡선의 하우스 도르프 거리

를 측정합니다)

makeBezier(0,360,100,0);(근사시키기위해서 첫 번째로 실행시켜야하는 함수입니다)

Bezier2D(300,100,150,250,400,300,1);

Bezier2D(700,200,850,250,800,400,2);

Bezier_line2D(300,100,150,250,400,300,700,200,850,250,800,400);(두 베지어 곡선이 있을 때 움직이는 경로를 표시하여 줍니다)

Bezier_move2D(300,100,150,250,400,300,700,200,850,250,800,400);(곡선의 움직임을 한번에 표현하여 줍니다)

Bezier_move_change2D(300,100,400,25,500,100,450,900,400,700,500,800,move,0,0,0,0);(앞의 곡선, 뒤의 곡선 내분비, RGB순으로 면을 채워줍니다)아래에 있는 change랑 같이 작동시키면 뒤의 부분이 동시에 지워져 선의 모양이 됩니다)

Bezier_move_change2D(300,100,500,175,500,100,450,900,500,1000,500,800,move,0,0,0,0);

Bezier_move_change2D(300,100,400,25,500,100,450,900,400,700,500,800,move-0.01,255,255,255,1);

Bezier_move_change2D(300,100,500,175,500,100,450,900,500,1000,500,800,move-0.01,255,255,255,1);

if(move<=1)

```
{
    move+=0.002;
}
else
{
    background(255);
    move=0;
}
```

내분비를 변화시켜줍니다

- 자신이 분석해 변경하거나 바꾼 것들

처음에는 뭘할지 모르고 조금씩 바꾸면 다른 사람들이랑 많이 겹칠거 같은 느낌이 들어 예시 코드의 void draw, void setup같은 기초적인 함수들을 익히고 background, size같이 배경을 바꿀수 있는 것을 배웠다. 그리고는 톱니바퀴를 만들 생각하고 블로그를 참고하며 명령어를 하나씩 알아가며 만들어서 바꾸거나 변경하지는 않았습니다 그리고 갑자기 제 알앤이가 애니메이션화랑 관련되어있어서 한번 processing과 연관지어 보았습니다

참고한 예시 또는 내용

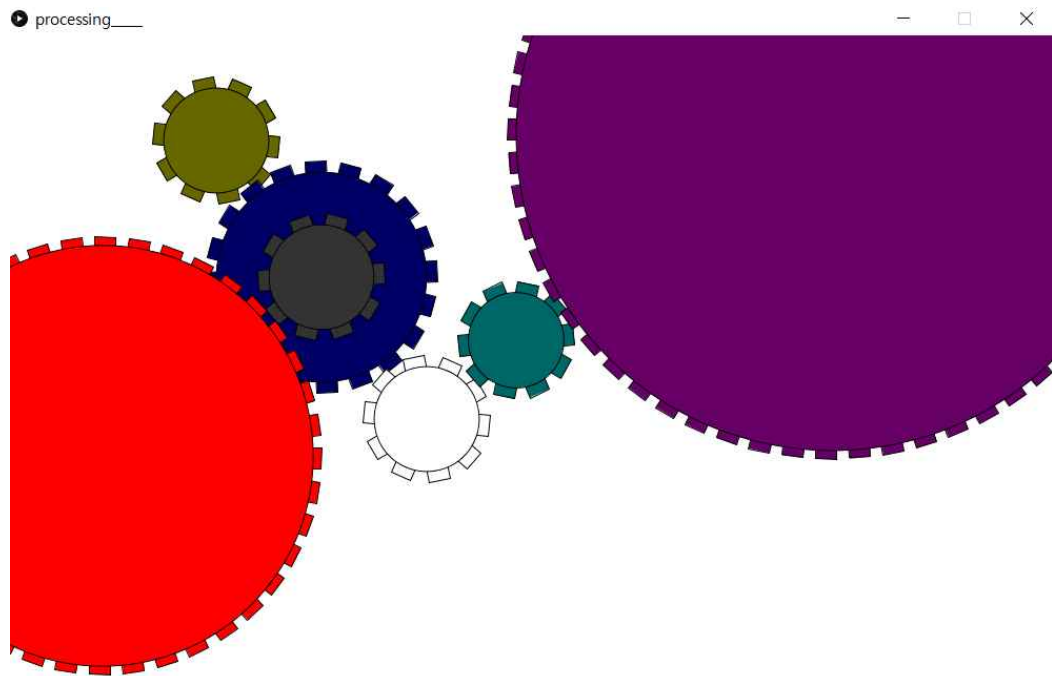
블로그에 있는 예시를 활용하여 명령어를 알아 나갔으며 가장 익숙지 못한 명령어는 배열을 선언하는 것으로 이를 가장 많이 참고하였습니다. 그리고 마우스를 연결지어 만드는 것도 있는데 이와 관련된 내용도 많이 참고하였습니다.

- 찾아보면서 재미있었던 예시와 그 이유는?

다른 예시를 찾아보니 마우스를 클릭했을 때 색깔이 바뀌는 것이 제일 재미있었던 거 같다. 다른 어려운 명령어는 이해를 못하겠어서 더 신기한 예시를 못 찾은 것일수도 있으나 실행하였을 때 그냥 이미지로 된 애니메이션이나 동작을 나타내는 것이 끝일 줄 알았는데 마우스를 이용하여 다른 실행결과가 나타난다는 것이 신기하였다

만들면서 생각해보거나 시도해 본 것들

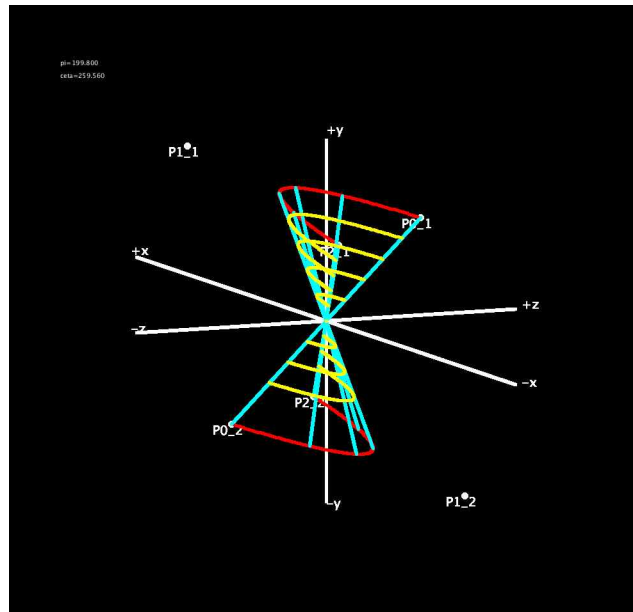
만들면서 처음에 색깔이 바뀌는거랑 마우스 클릭하는 것이 너무 신기해서 그것으로 마우스를 클릭했을 때 색깔이 바뀌는 것을 만들어 보았는데 딱히 의미가 없어서 톱니바퀴도 해보았고 여러 가지를 만들어 보았는데 확실히 알앤이에 관련된 것을 해보았더니 연구된 것이 많아 제일 완성도가 높은 것 같다. 오차율(하우스도르프거리)를 측정하는데 어떻게 코드를 만들어야하는지 잘 몰라 포문 하나로 만들어보려는 노력을 해보았지만 잘 되지 않아 포문을 두 번 돌려 수작업으로 만들었다.



톱니바퀴 processing 결과

만들면서 이해하게 되었거나 느낀 점들

처음에 다 만들고 실행해 보았는데 톱니바퀴는 돌아가긴 하지만 톱니바퀴 돌아가는 것이 처음 톱니바퀴는 제대로 돌아가는데 두 번째 톱니바퀴가 첫 번째 톱니바퀴 기준으로 돌면서 회전하는 결과가 나와 계속 당황스러웠고 이 때문에 3시간이 넘게 고민하고 찾아본 거 같다. 그랬더니 push.popmatrix로 쉽게 해결되는 것을 보고 어이가 없었던 것 때문에 push.popmatrix에 대해서 완벽히 이해한 거 같다. 그리고 톱니바퀴 말고도 제가 제출한 내용에 관련된 알앤이 에 관련된 것도 만들어보는 과정에서 오차율을 바꾸어주는 과정에서 바를 만드는데 마우스를 이용하여 컨트롤 할 수 있다는 것이 너무 신기하였다 나중에는 진짜 R&E 주제와 관련된 2D가 아닌 3D를 좌표계를 만들어 봄으로서 확장시키고 싶다



양식을 다 작성한 후에 실제로 확장시켜 만들어 보았습니다

3D 좌표계를 회전 행렬을 이용함으로서 구현해 보았는데 실제로 마우스 클릭에 대한 X,Y값으로 각도를 설정하여 만들었습니다. 각도를 3개 표현하고 싶은데 2개까지 밖에 못만들었다는 점에서 마우스 클릭으로 3개의 값을 정하게 도와주는 코드를 만들어 봄으로서 더 자연스럽게 움직이는 좌표계를 만들어 보고 싶다는 생각을 했다.