

## 063강 Curves(3)

Curves를 통하여 깃털을 만들기

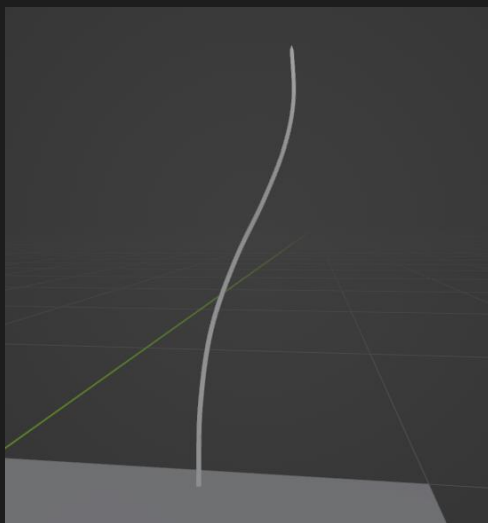


# Feather

깃털은 커브에 또다른 커브가 붙어있는 형태입니다.  
Instance on Points를 통하여 만들 수 있습니다.



Instance



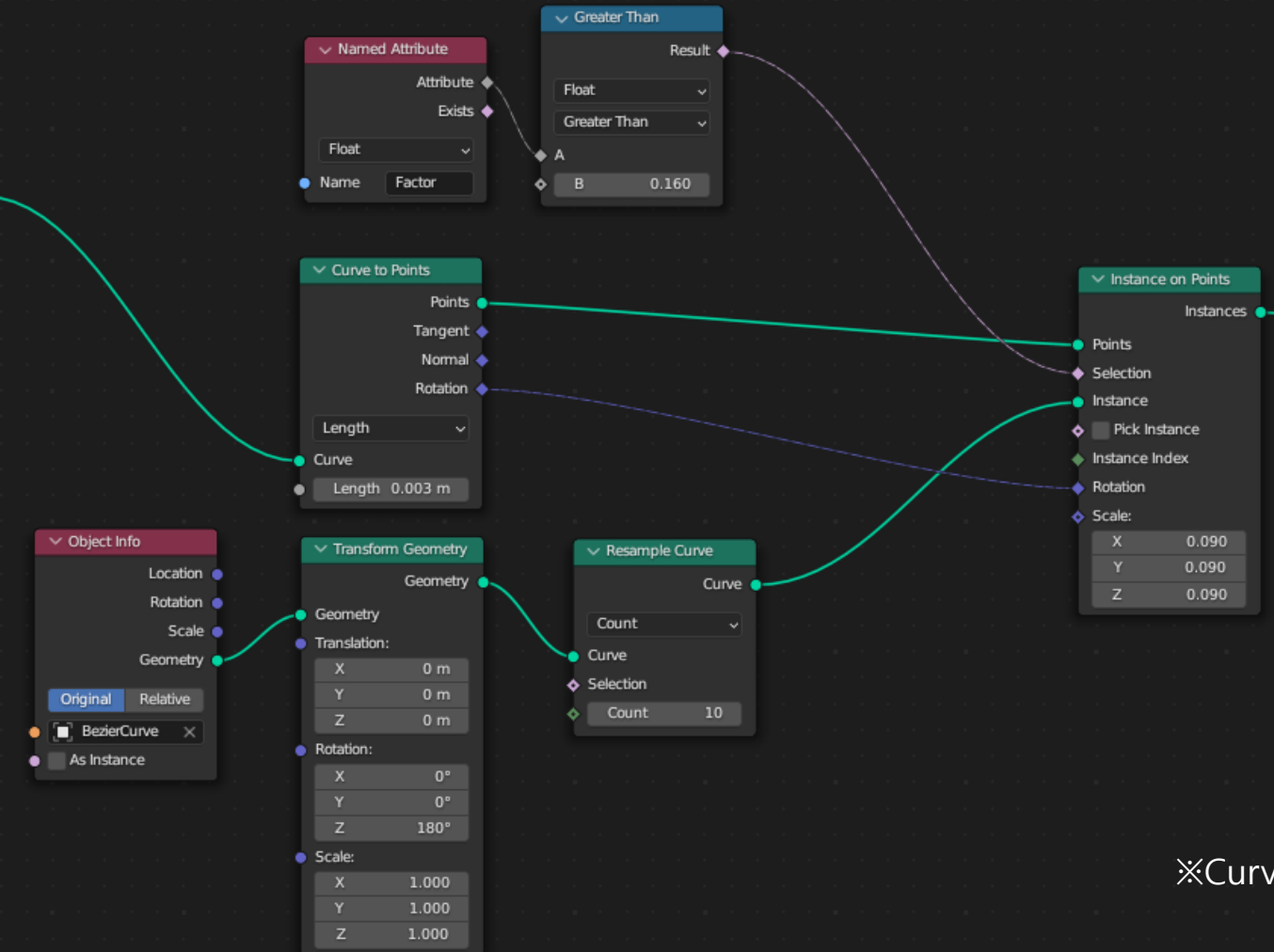
Points



Instance on Points

# Instance on points

베지에 커브로 깃털 한 올을 만들어 커브에 붙입니다.  
Curve to Points를 이용하면 점들의 간격과 깃털의 회전값을 쉽게 얻을 수 있습니다.  
다만 Spline Parameter가 전달되지 않으므로 Factor 등의 정보를 이전에 저장해 두어야 합니다.

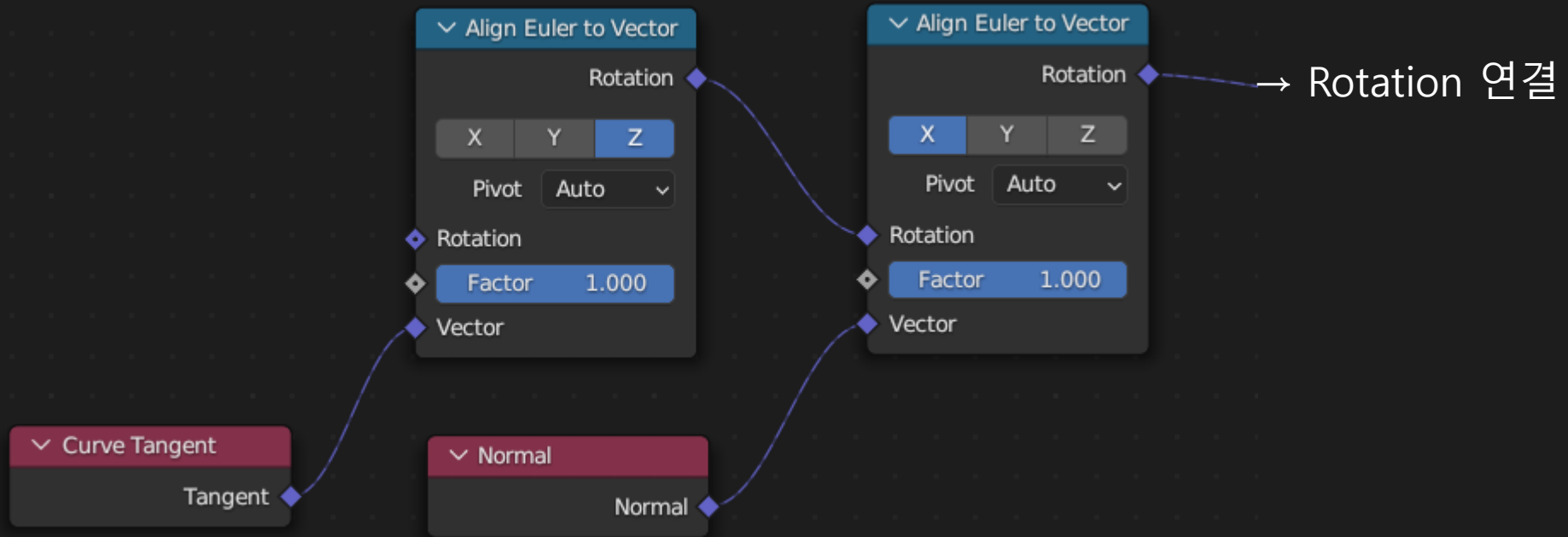


※Curves는 인스턴스를 제대로 표현하지 못하므로 나중에 Realize해 줍니다.

# Curve to Points를 사용하지 않는 경우(Optional)

Curve to Points를 사용하지 않는 경우 Rotation을 직접 만들어야 합니다.

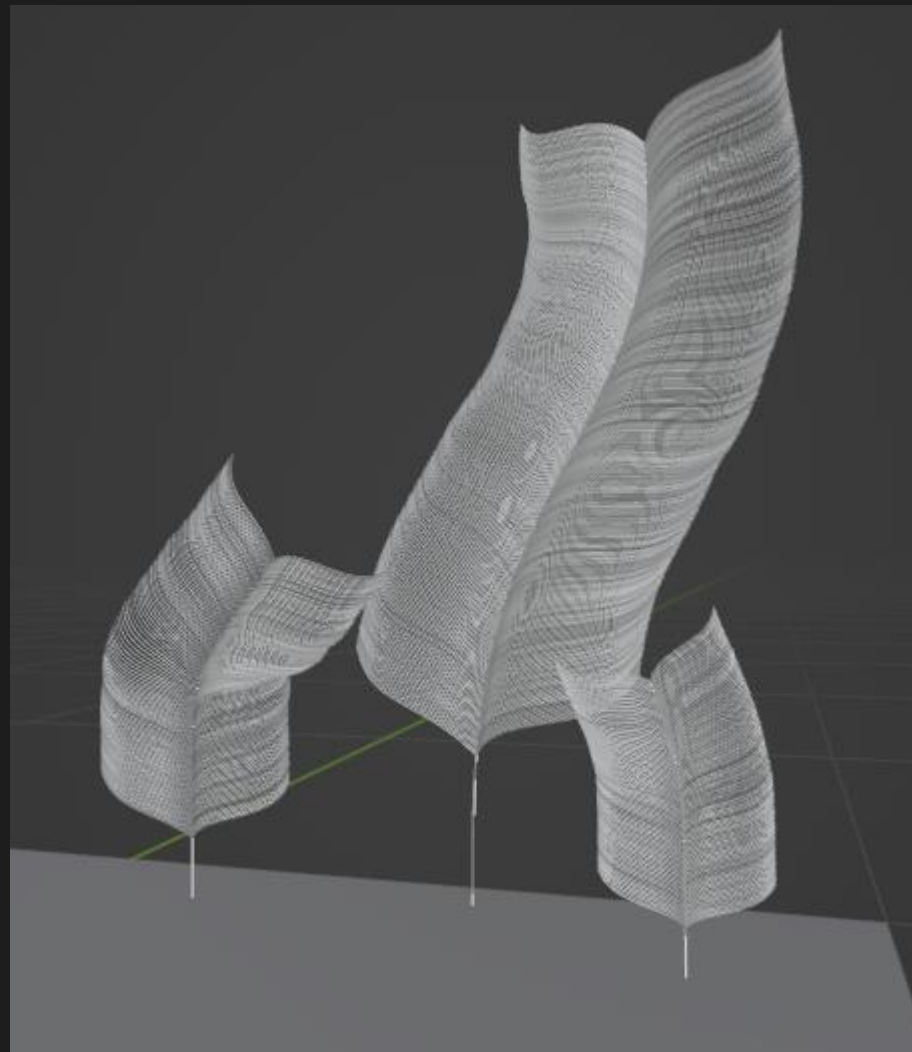
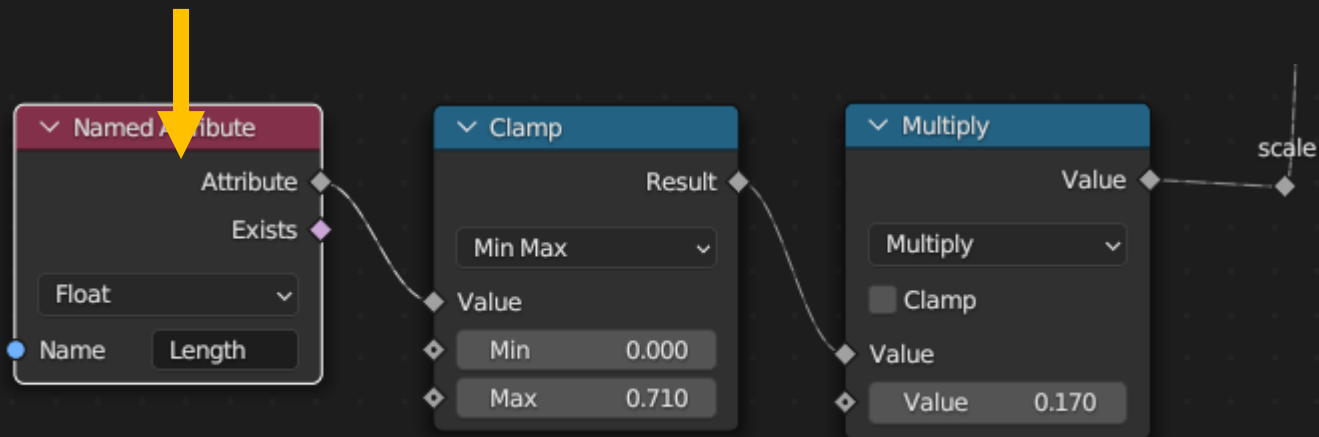
깃털은 커브의 두개의 축, Tangent와 Normal에 모두 정렬되어야 합니다.  
Align Euler to Vector를 두 번 사용하여 회전값을 얻을 수 있습니다.



# 깃털 크기

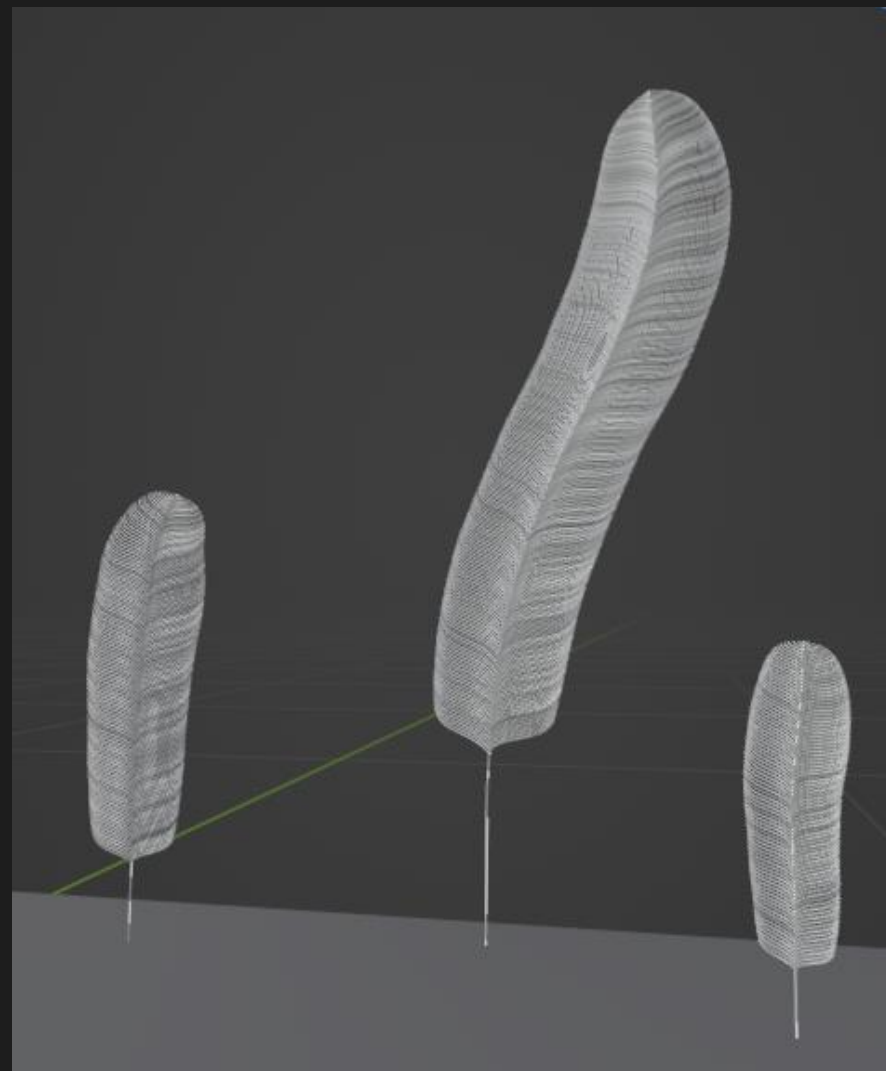
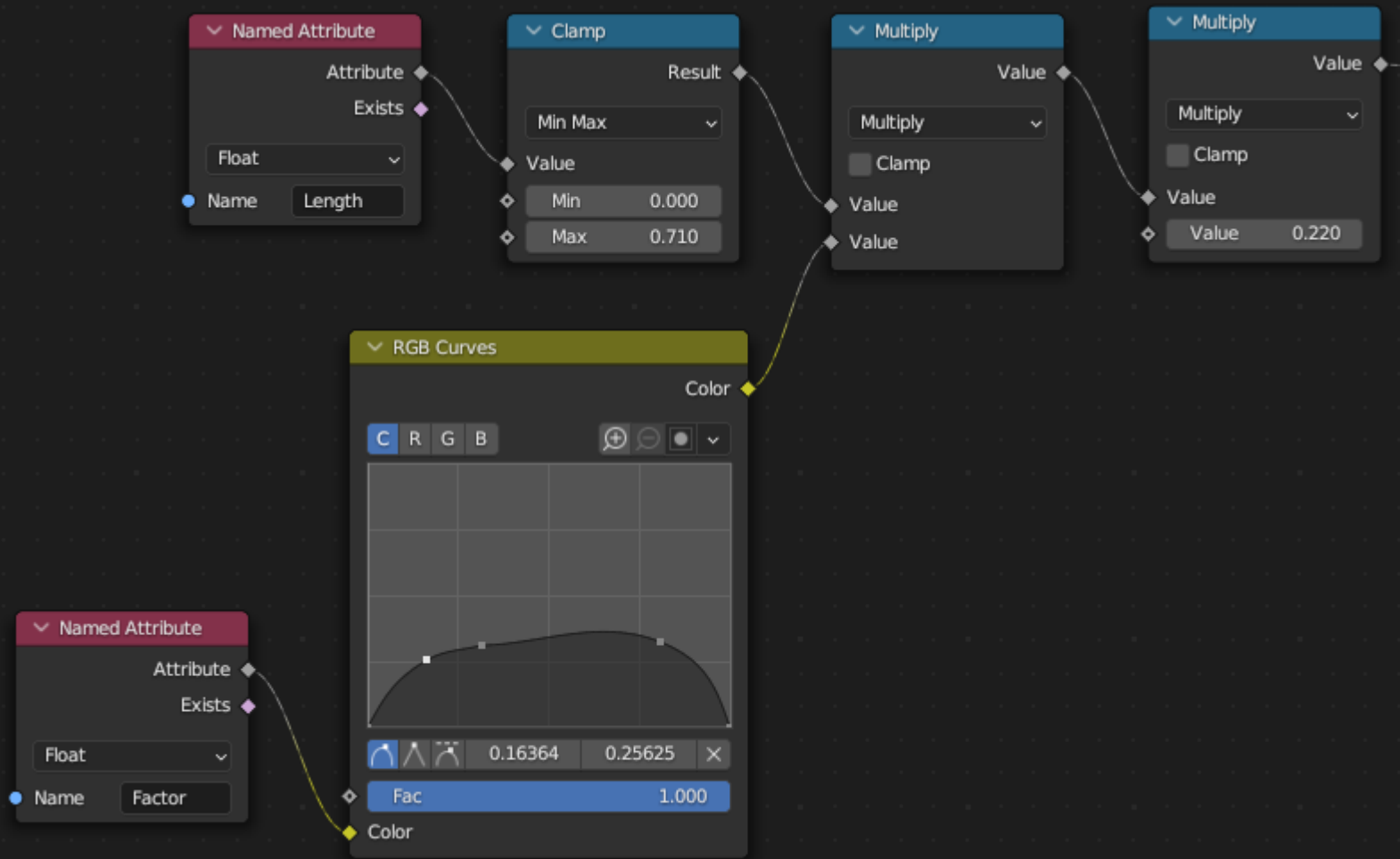
원본 커브의 길이에 따라 크기를 조절합니다.  
이 때, 너무 커지지 않게 Clamp로 크기 상한을 정할 수 있습니다.

Spline Length 는  
Curve to Points 이전에 저장해둡니다.



# 깃털 모양

원본 커브의 Factor를 따라 스케일을 추가로 조절하여 최종적인 깃털 모양을 만듭니다.



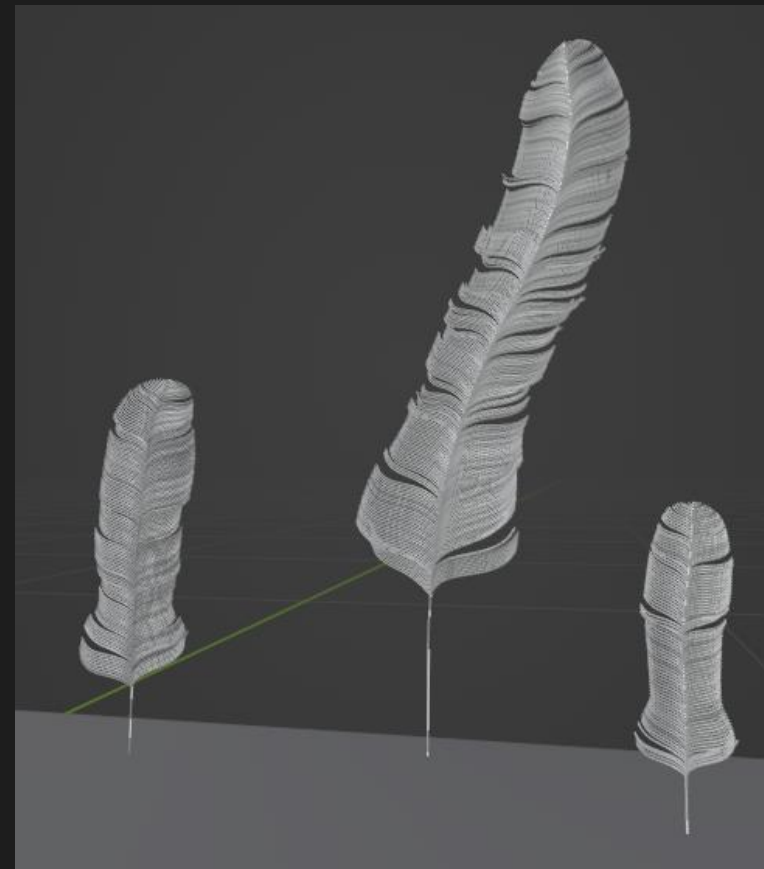
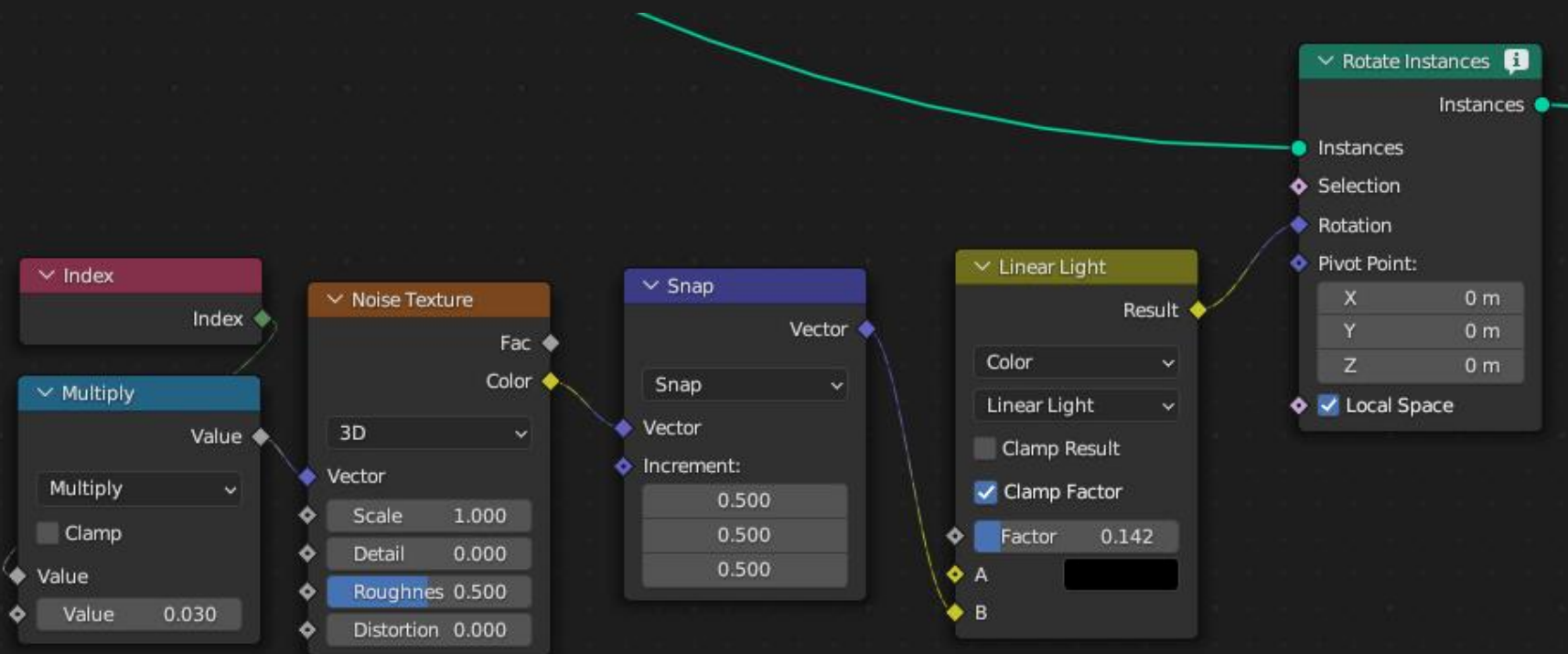


# 디테일 표현

깃털의 갈라진 부분을 표현해봅시다.

Curve의 index는 항상 정렬되어 있으므로, 노이즈의 좌표로 활용하면 원본 커브를 따라 노이즈를 생성할 수 있습니다.

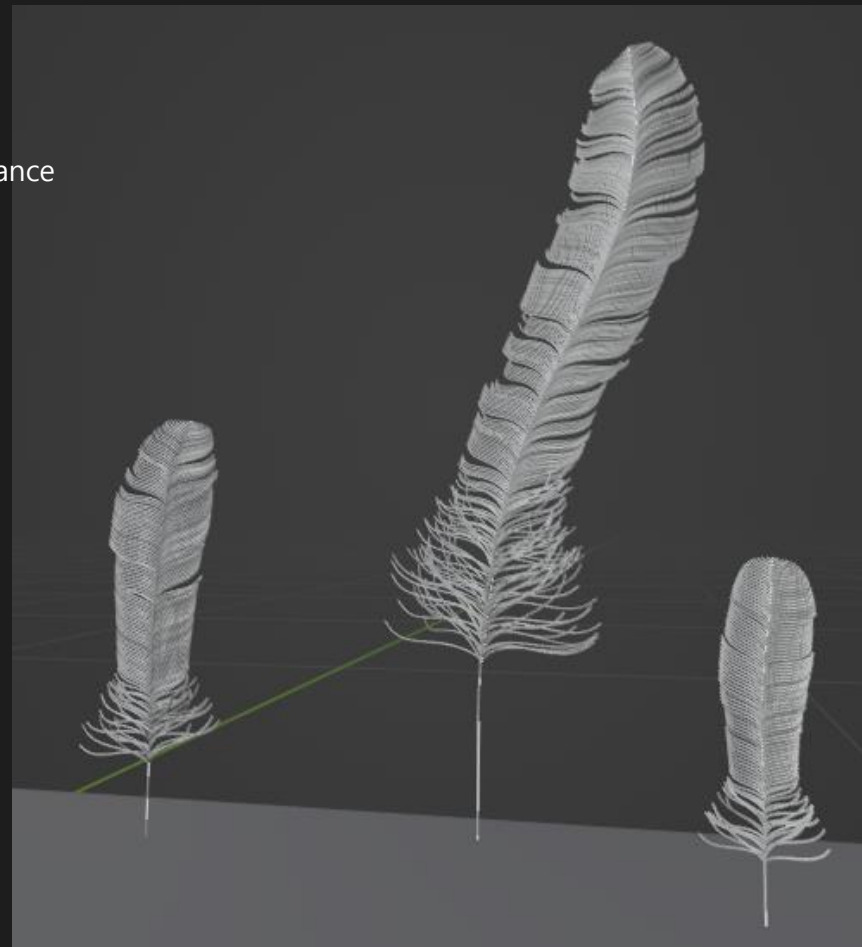
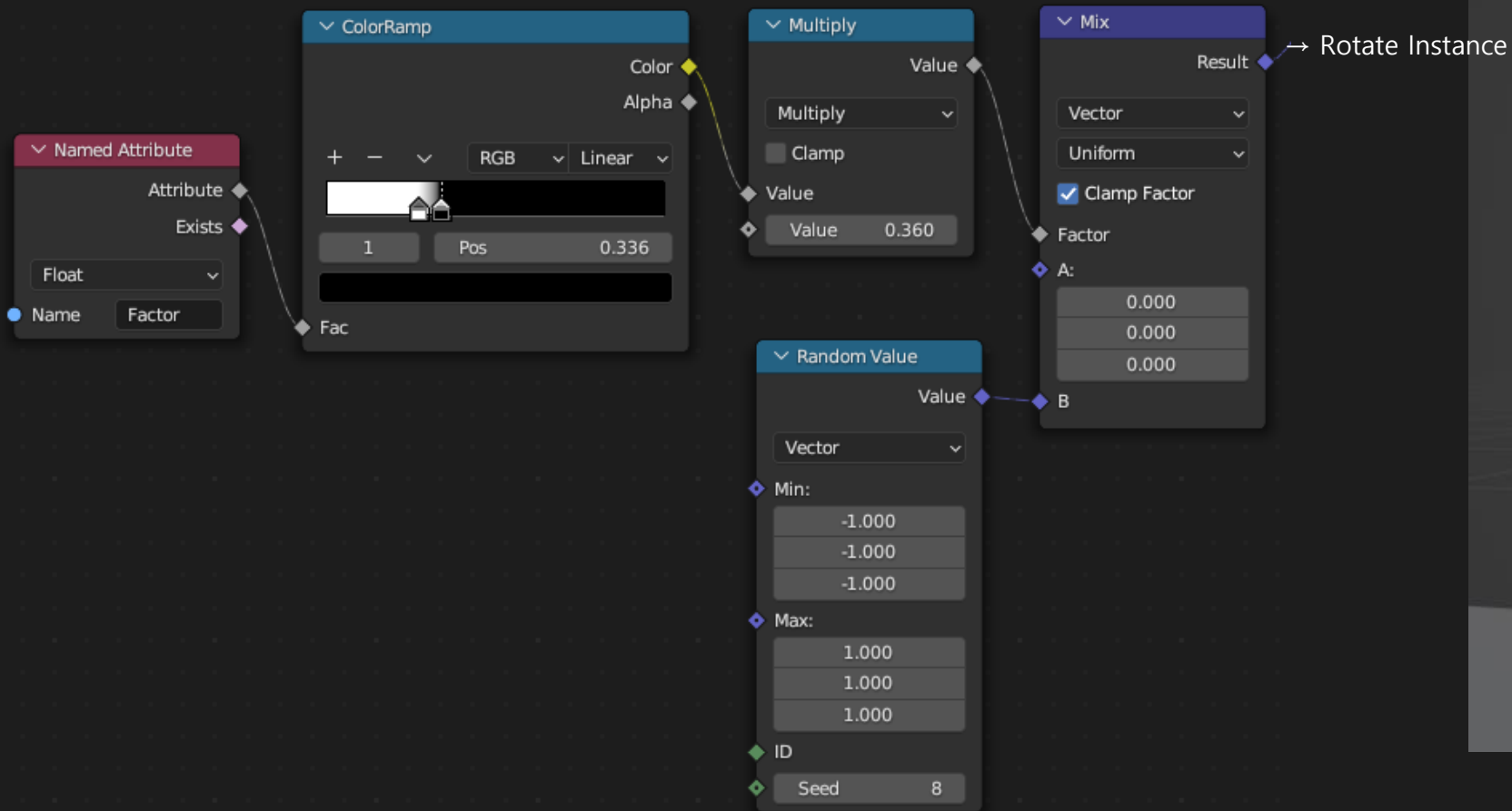
이 상태에서 Snap으로 비슷한 값을 묶으면, 깃털의 덩어리짐과 갈라짐을 표현할 수 있습니다.



## 디테일 표현(2)

Rotate Instances를 한번 더 사용하여, 깃털의 형클어짐을 표현합니다.

이 때는 붙어있는 깃털 한올한올 사이의 관계가 없으므로, 노이즈가 아니라 그냥 랜덤을 사용합니다.

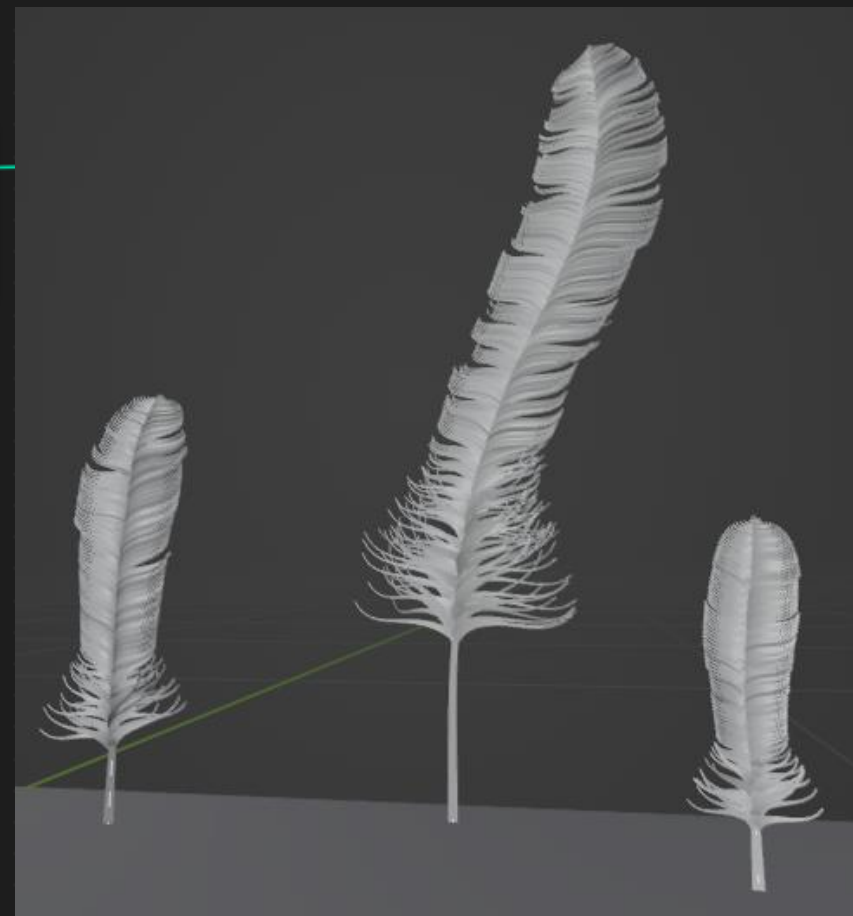
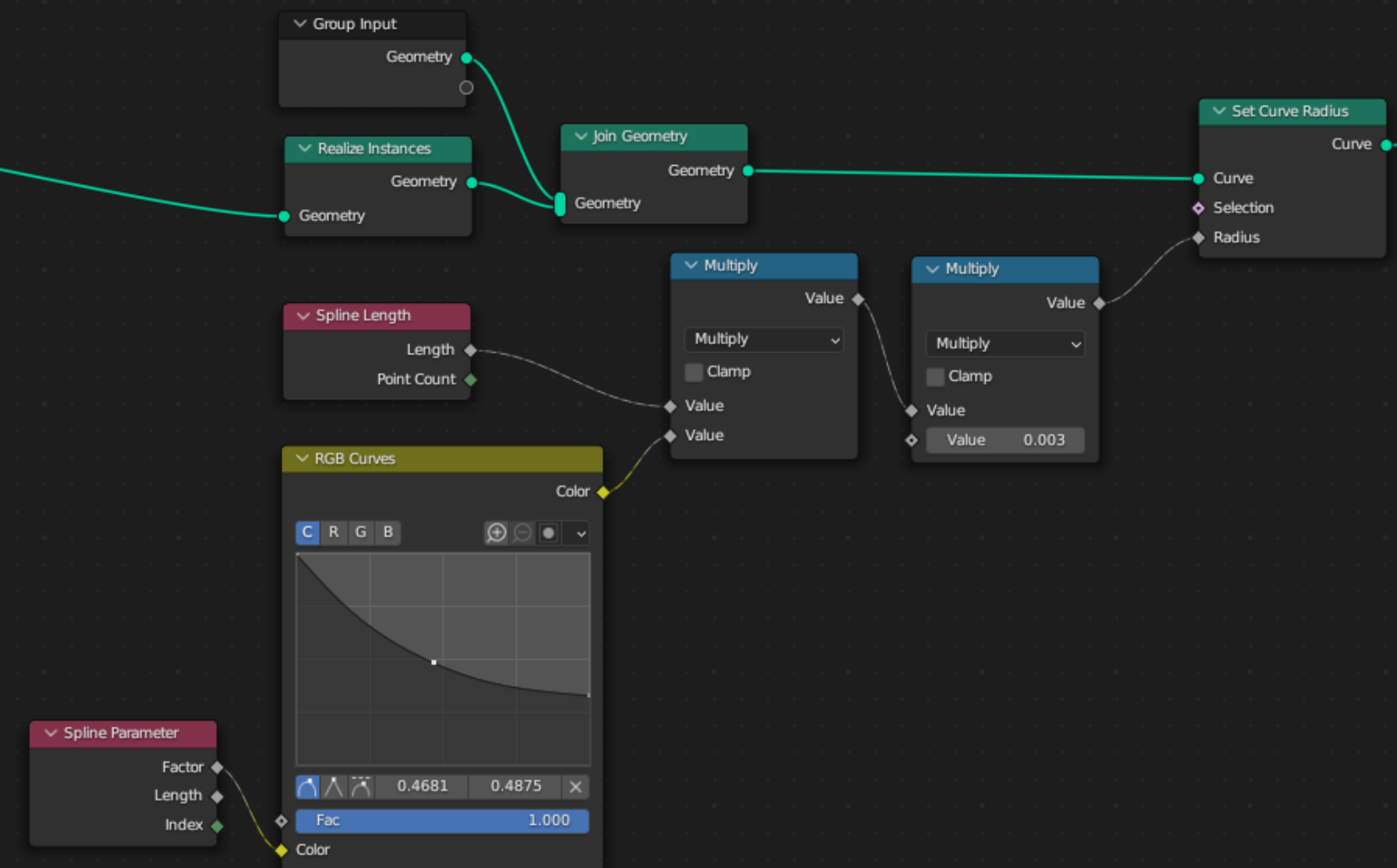




## 두께 표현

Realize 후, 원본 커브와 합친 뒤 두께를 조절해줍니다.

\*Eevee와 Cycles의 두께가 미묘하게 다르므로 유의하세요.



# 변형을 따라가기 위해

움직이지 않는 모델이라면 상관없지만, Armature deform 등 변형이 일어났을 때 커브가 따라가기 위해서는 `surface_uv_coordinate`가 제대로 입력되어 있어야 합니다.

Instance가 원본 커브의 `surface_uv_coordinate`를 받아오지만, 도메인이 control point로 잘못되어 이동합니다.

이 문제는 도메인을 Spline으로 교체하는 것으로 간단히 해결됩니다.

이후에 deform curves on surface를 사용하면 변형을 따라가게 됩니다.

