

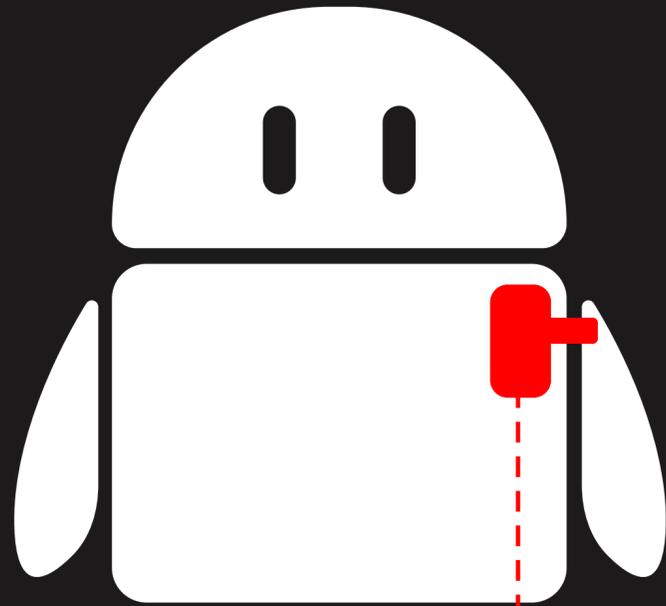
# 3D 애니메이션 환경 기반의 디자이너 중심 로봇 움직임 설계 프레임워크

A Design-Centered Framework for Robot Motion Design in a 3D Animation Environment

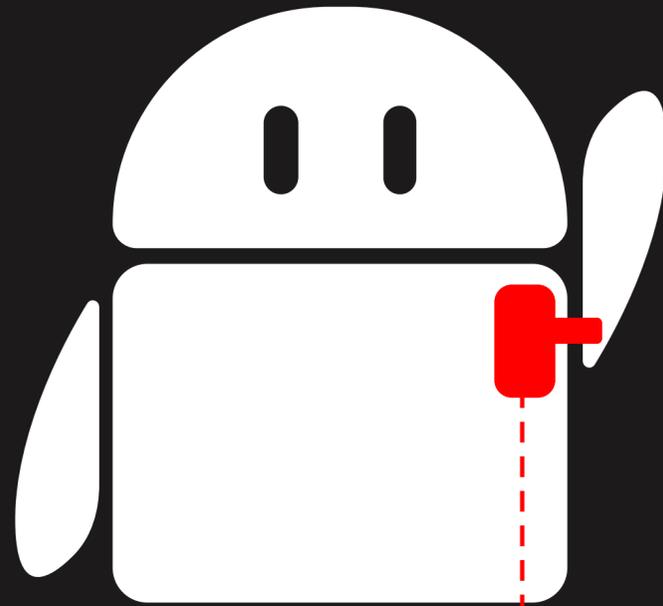
## 1. 서론

### 1.1. 로봇 움직임 구현에서의 디자이너 접근성 한계

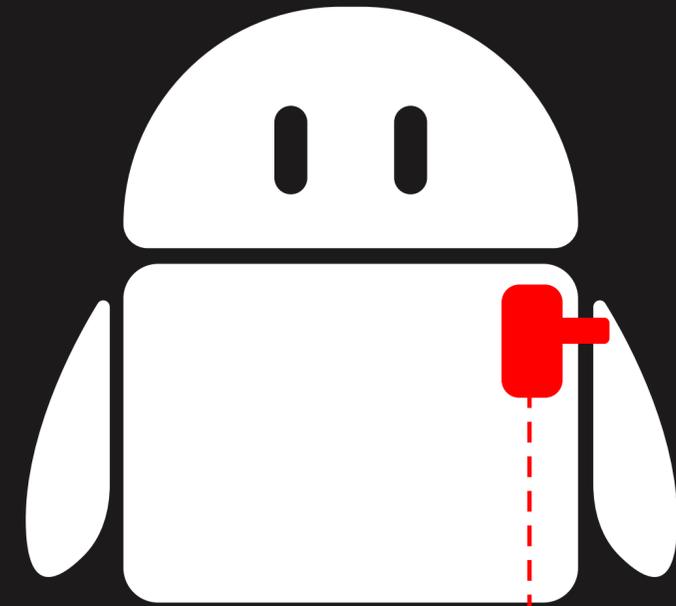
# 로봇 시나리오 1: 팔을 올려 손을 들었다 내리는 모습 표현하기



모터 회전 0°

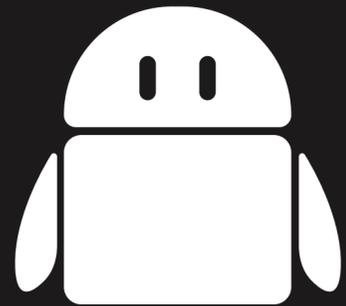


모터 회전 90°



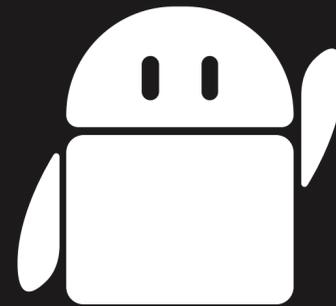
모터 회전 0°

# 로봇 시나리오 1: 팔을 올려 손을 들었다 내리는 모습 표현하기



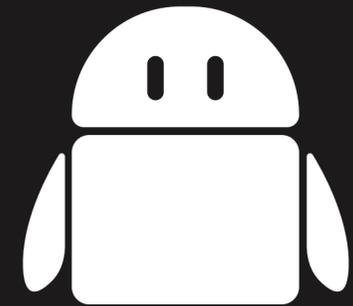
0.00sec

모터 회전 0°



0.50sec

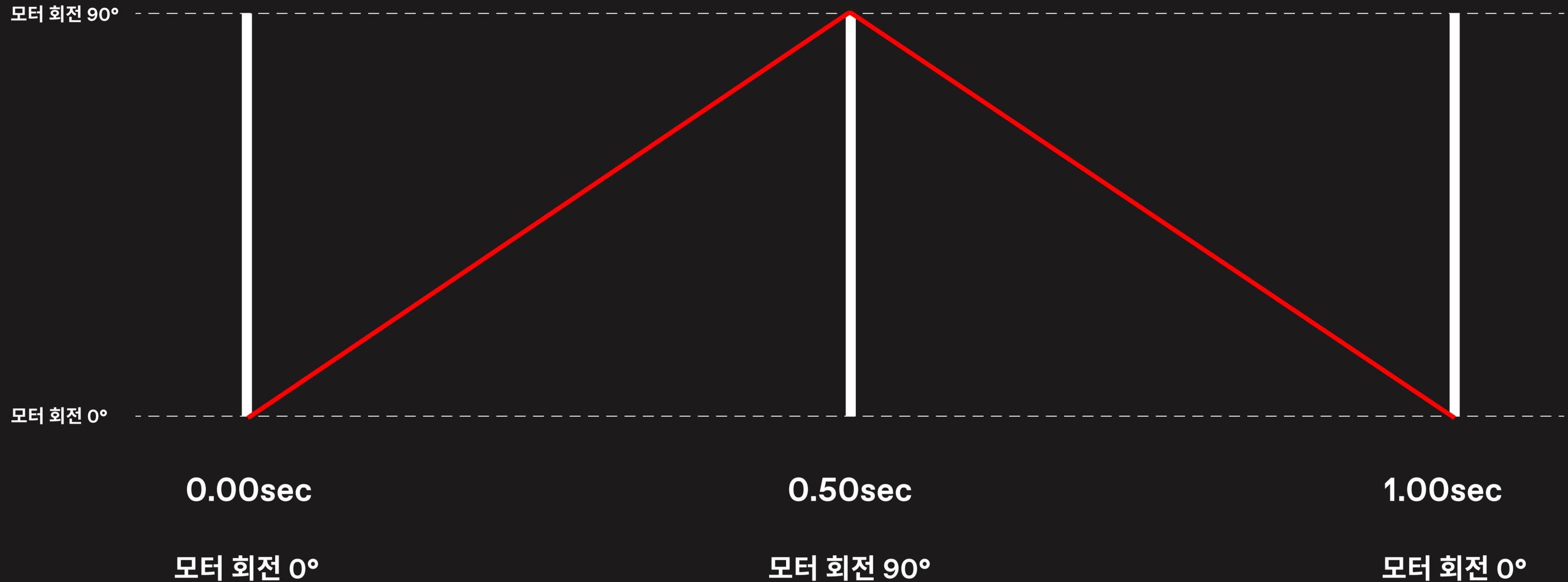
모터 회전 90°



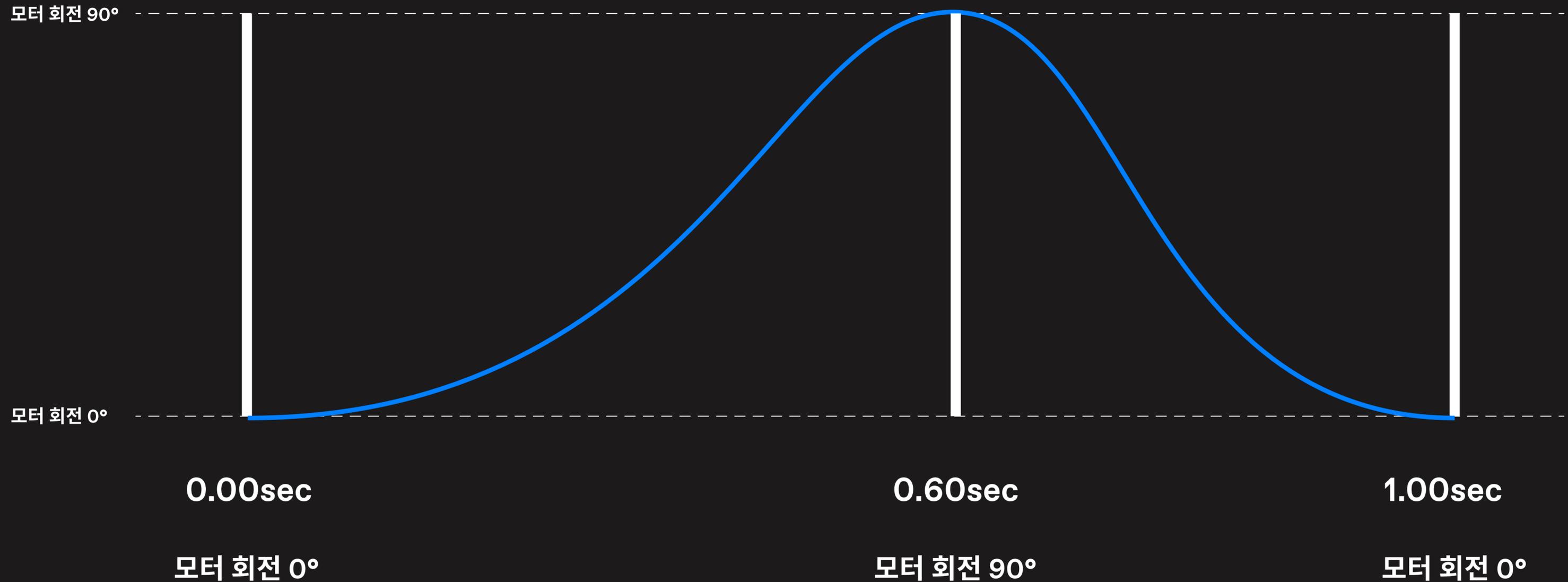
1.00sec

모터 회전 0°

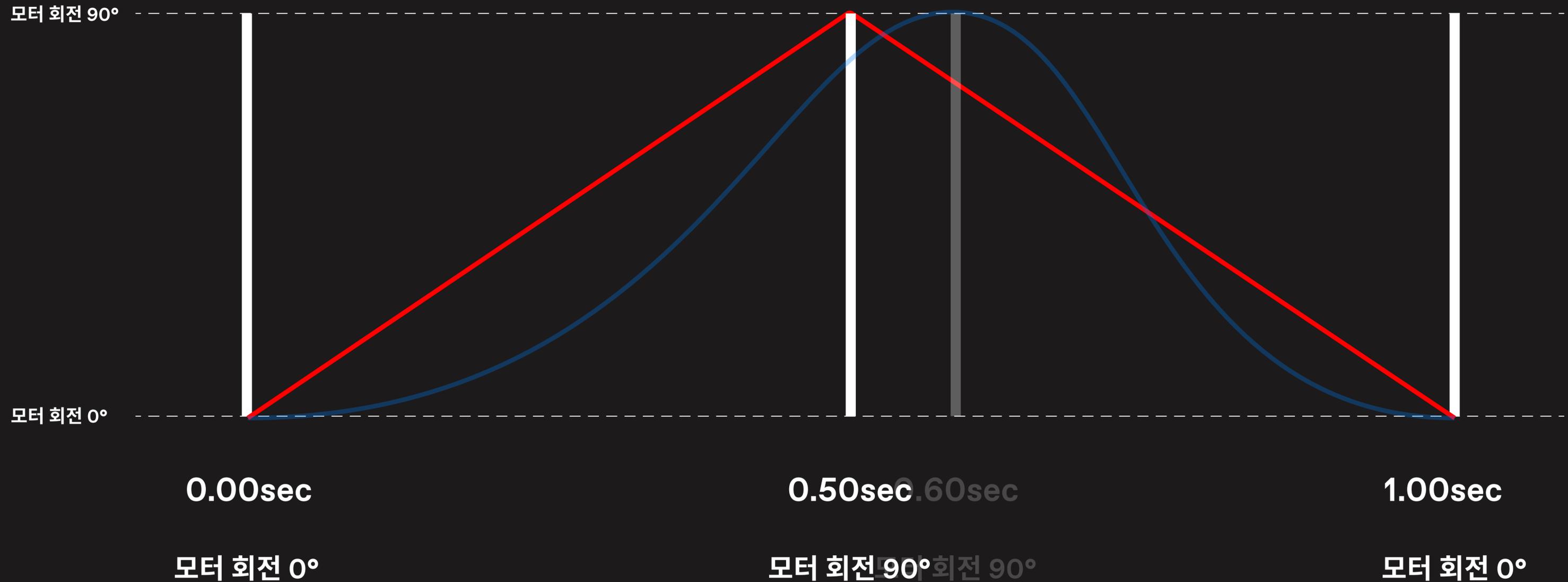
# 1차적인 정보로 짜여진 모션



# 학생 디자이너: 더 자연스럽게, 부드럽게 시작해서 끝나면 좋겠습니다!

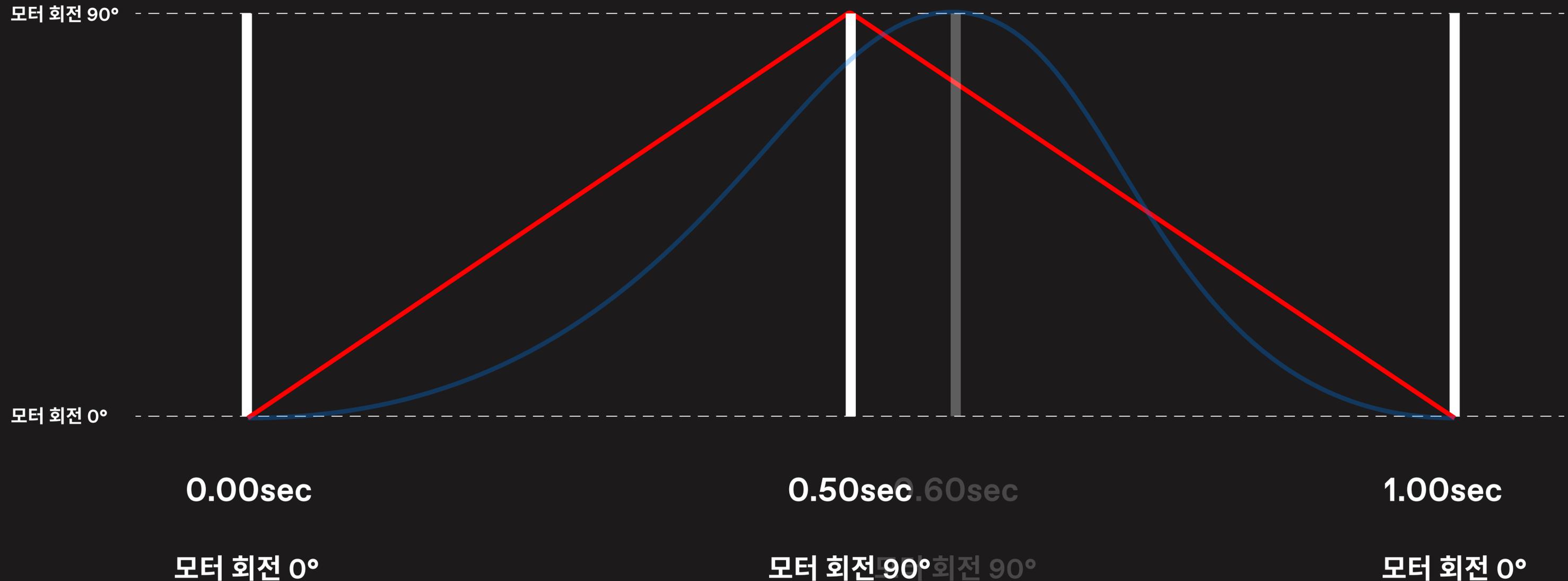


# 학생 엔지니어: 시간 - 속도 수정하고 사인 보간 넣으면 되겠네요

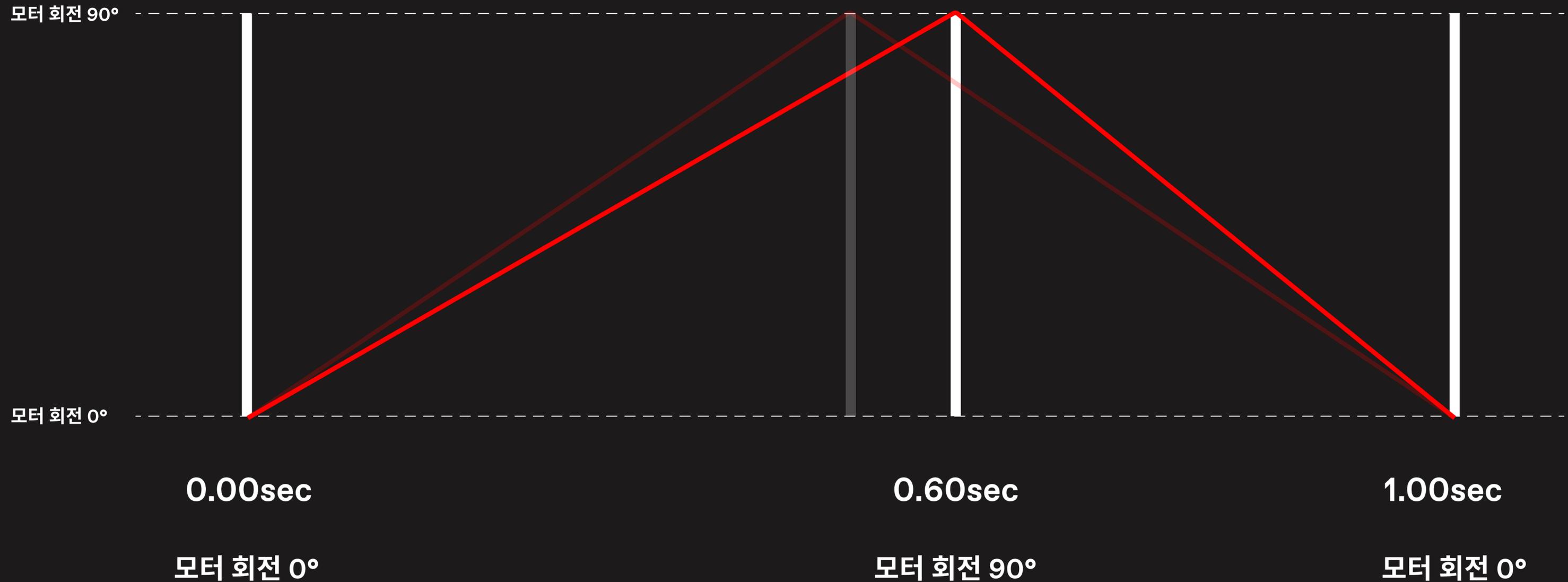


학생 엔지니어: 시간 - 속도 수정하고 사인 보간 넣으면 되겠네요

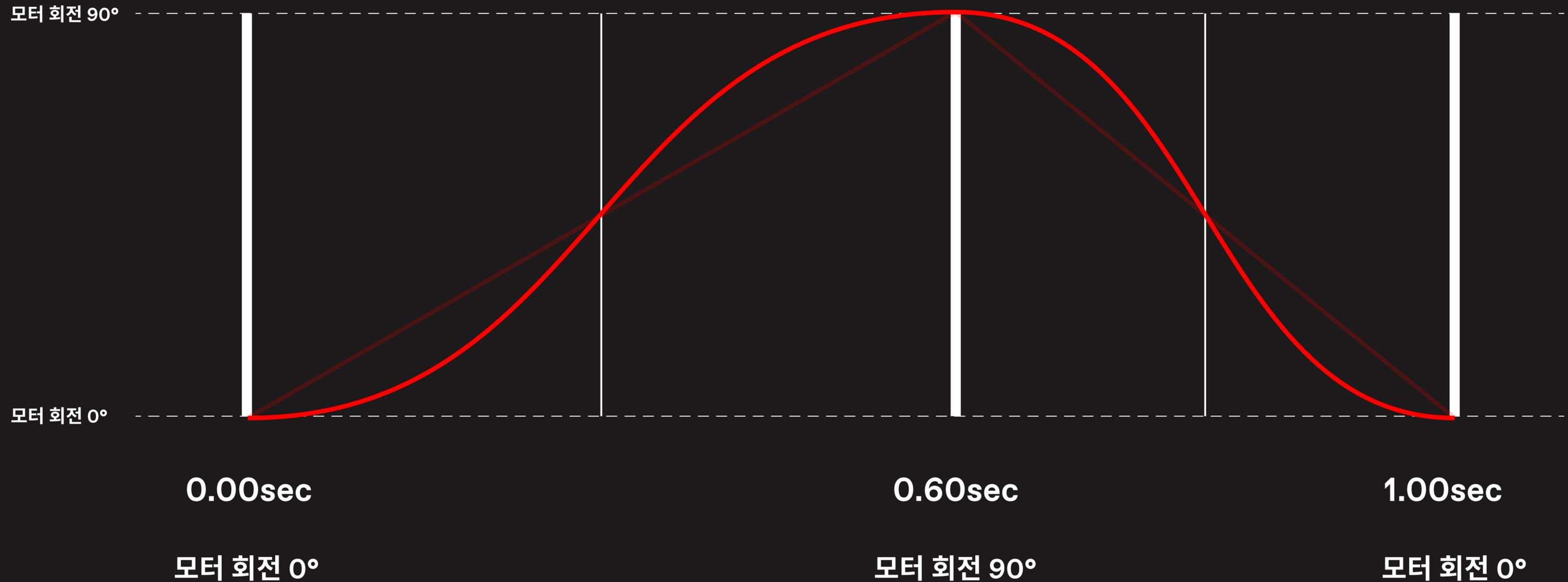
학생 디자이너: 그게 무슨 뜻일까... 어째든 가능하다는 것이군



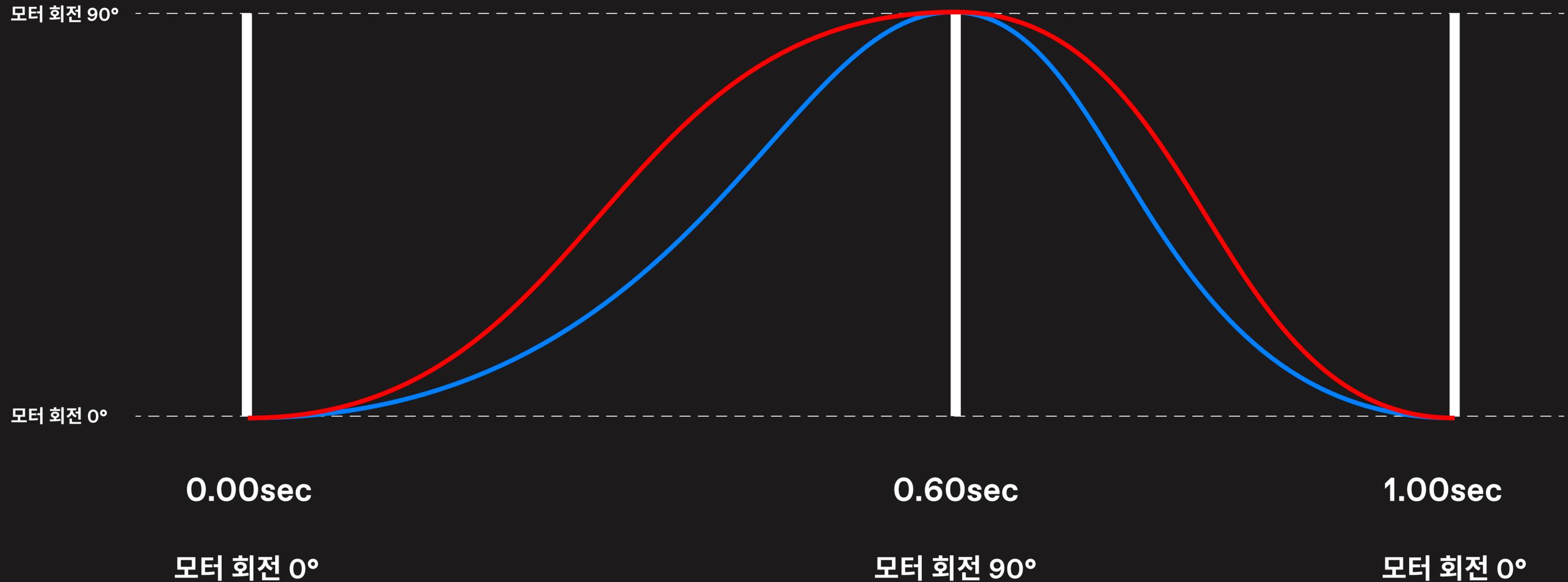
# 학생 엔지니어: 시간 - 속도 수정하고 사인 보간 넣으면 되겠네요



# 학생 엔지니어: 시간 - 속도 수정하고 사인 보간 넣으면 되겠네요

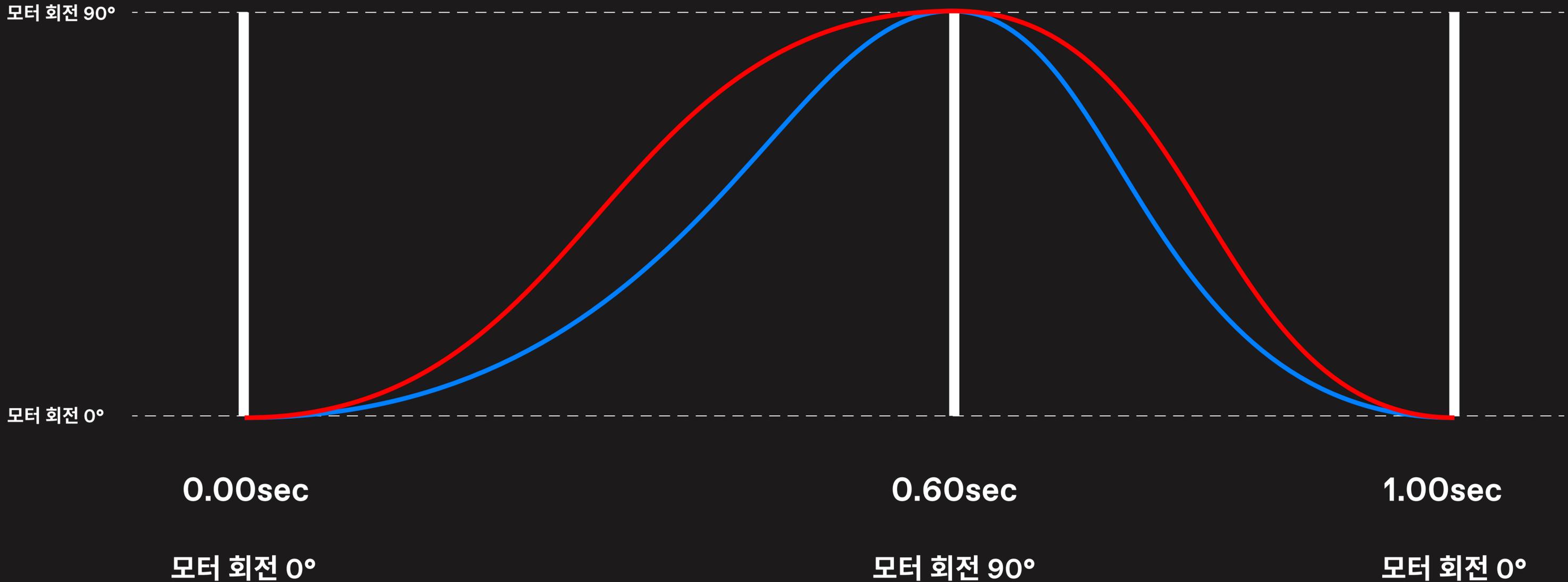


# 학생 디자이너: (비슷하긴 한데... 살짝 느낌이 다르네...)

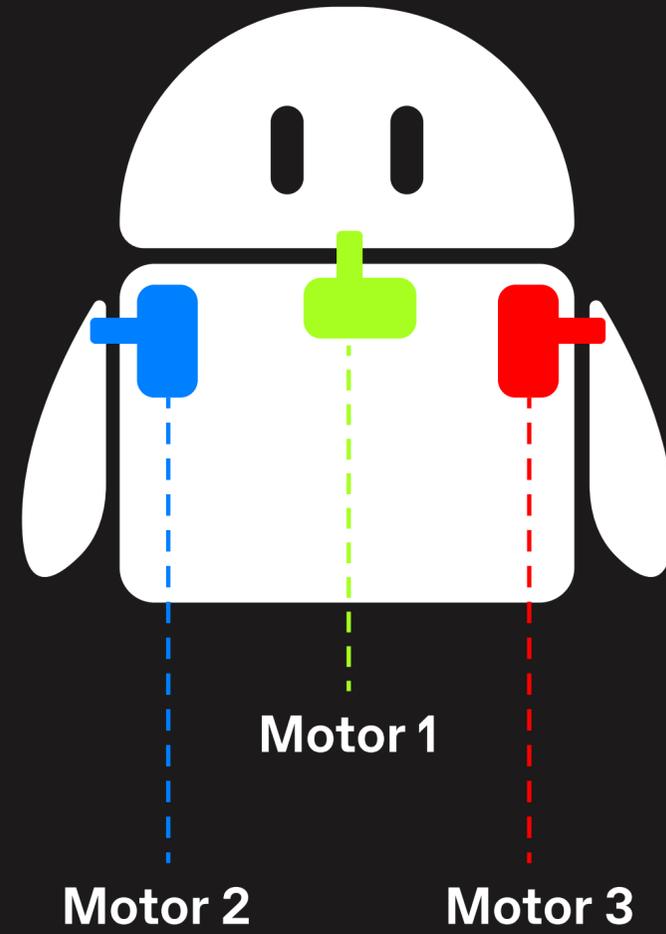


# 제안과 수식이 일치할 때까지 수정 ..... 혹은 적당한 지점에서 타협

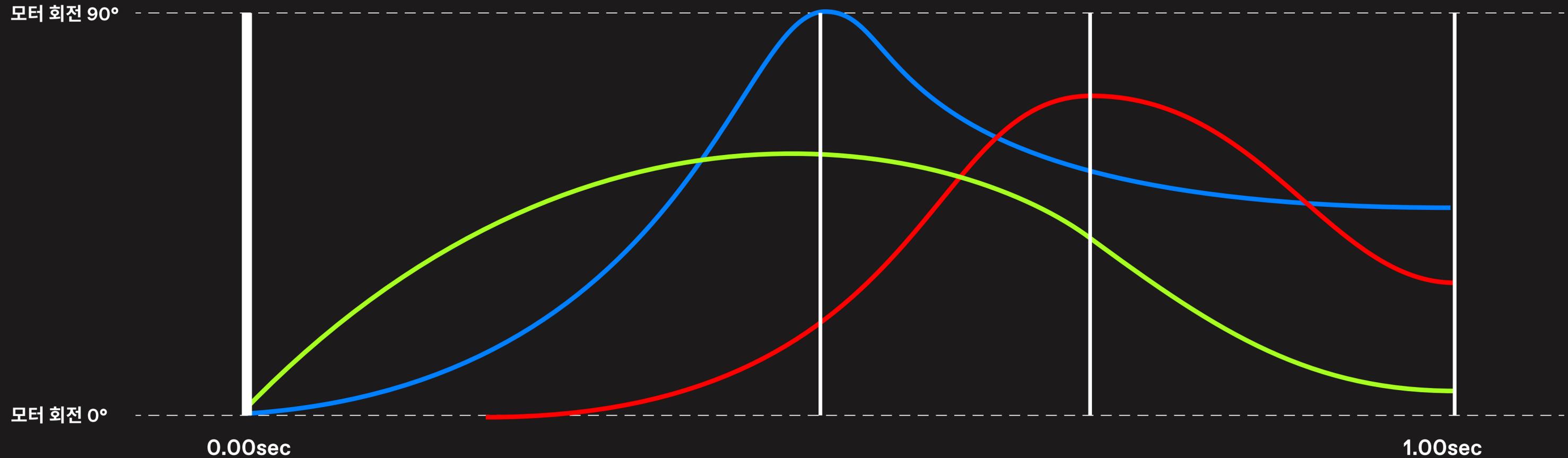
by Designer by Engineer



# 로봇 시나리오 2 : 춤추는 듯이 움직이기



# 학생 디자이너: 모터 1번 2번 3번이 다 다른 타이밍에 조금씩 다르게 움직여야...



Motor 1 Movement

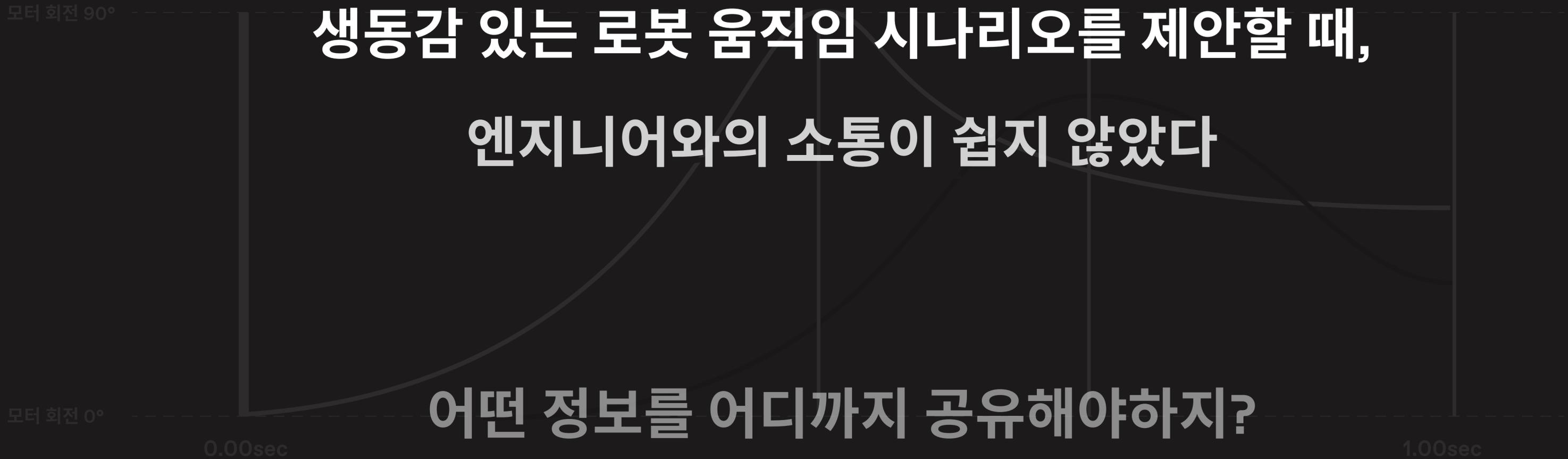
Motor 2 Movement

Moter 3 Movement

학생 디자이너: 모터 1번 2번 3번이 다 다른 타이밍에 조금씩 다르게 움직여야...

**생동감 있는 로봇 움직임 시나리오를 제안할 때,  
엔지니어와의 소통이 쉽지 않았다**

**어떤 정보를 어디까지 공유해야하지?**



Motor 1 Movement

Motor 2 Movement

Motor 3 Movement

학생 디자이너: 모터 1번 2번 3번이 다 다른 타이밍에 조금씩 다르게 움직여야...

**생동감 있는 로봇 움직임 시나리오를 제안할 때,  
구현을 직접 하는 입장이 아니니  
결과물에서 약간의 차이는 감수하게 되었다**

**조금 다르긴 하지만 비슷하니 이정도로...**

모터 회전 90°

모터 회전 0°

0.00sec

1.00sec

Motor 1 Movement

Motor 2 Movement

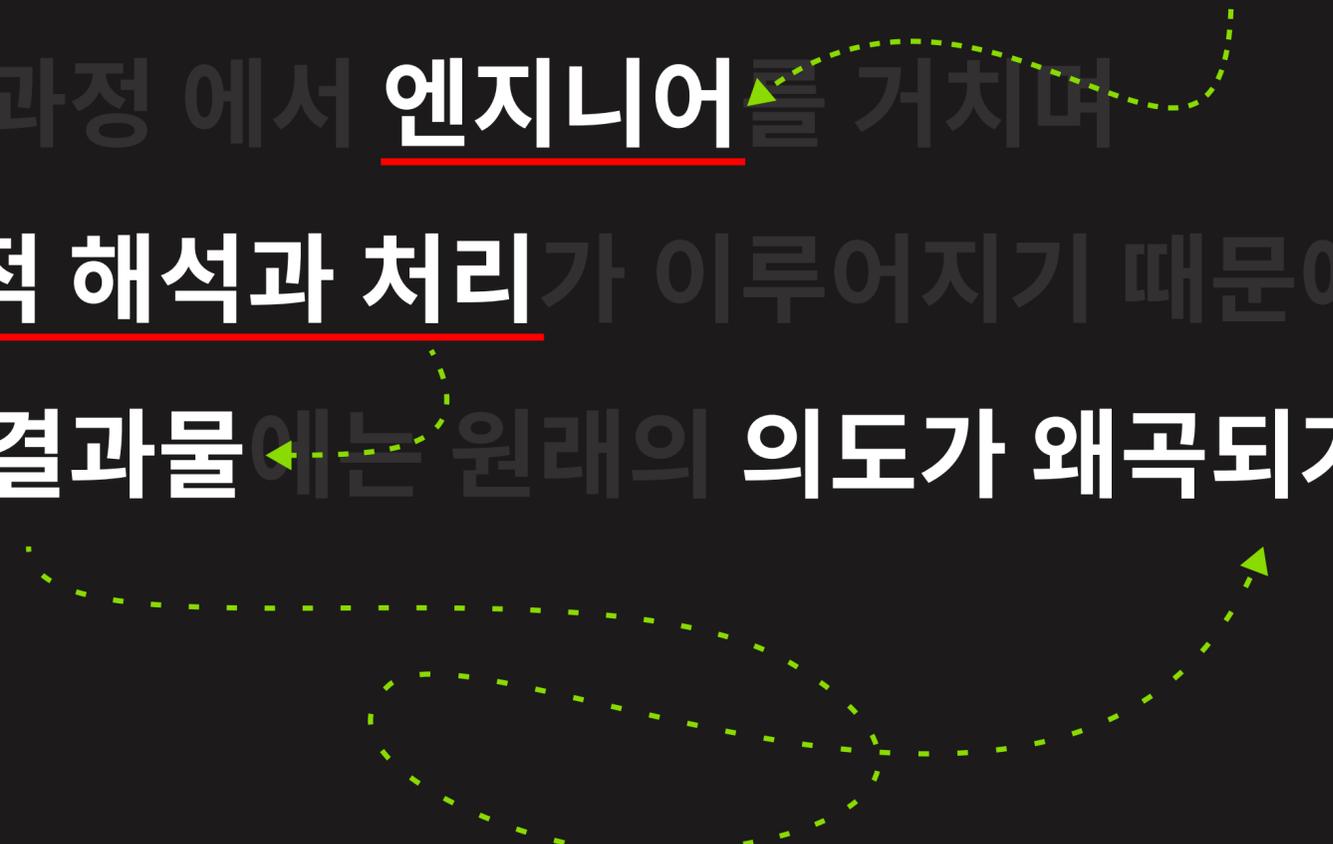
Motor 3 Movement

**생동감 있는 로봇 움직임 시나리오를 제안할 때,  
더 직접적으로, 표현적인 애니메이션 시안이  
로봇의 움직임에 그대로 반영되었으면 좋겠다**

**관절을 동시에 다른 정도로 움직여줘야 더 Lively 한데...**

## 문제점

디자이너가 HRI(Human-Robot Interaction)를 고려한  
생동감 있는 움직임의 표현 의도를 제안하더라도,  
구현 과정에서 엔지니어를 거치며  
기술적 해석과 처리가 이루어지기 때문에  
최종 결과물에는 원래의 의도가 왜곡되거나 희석되는 경우가 빈번하다.



문제점 원인 1

**디자이너-엔지니어 간 커뮤니케이션 격차**

문제점 원인 2

**프로그래밍 환경에 대한 비전공자(디자이너)의 진입장벽**

## 문제점 원인 1

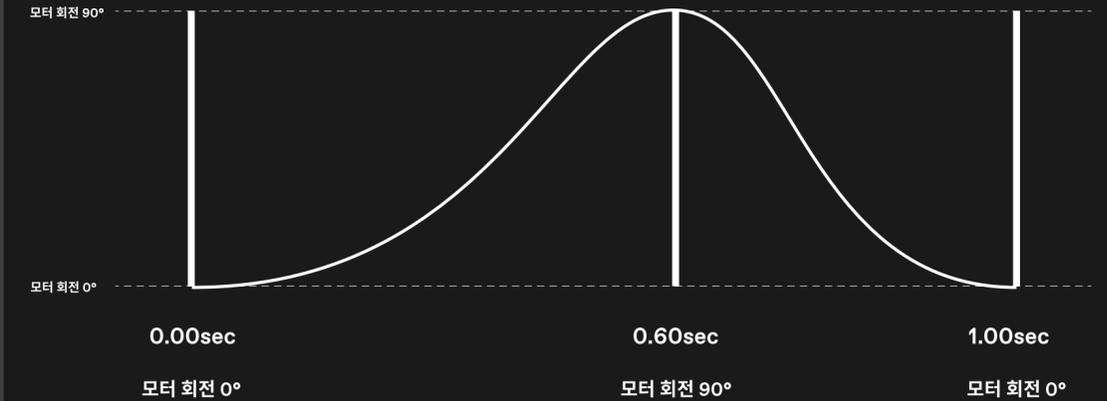
# 디자이너-엔지니어 간 커뮤니케이션 격차

Mcharek(2019)과 Nguyen(2022)은 이러한 현상이 언어, 도구 체계, 작업 방식의 차이에서 비롯된다고 지적하며, 디자이너-엔지니어 간 커뮤니케이션 격차가 협업 과정의 가장 큰 한계로 작용한다고 보고하였다.

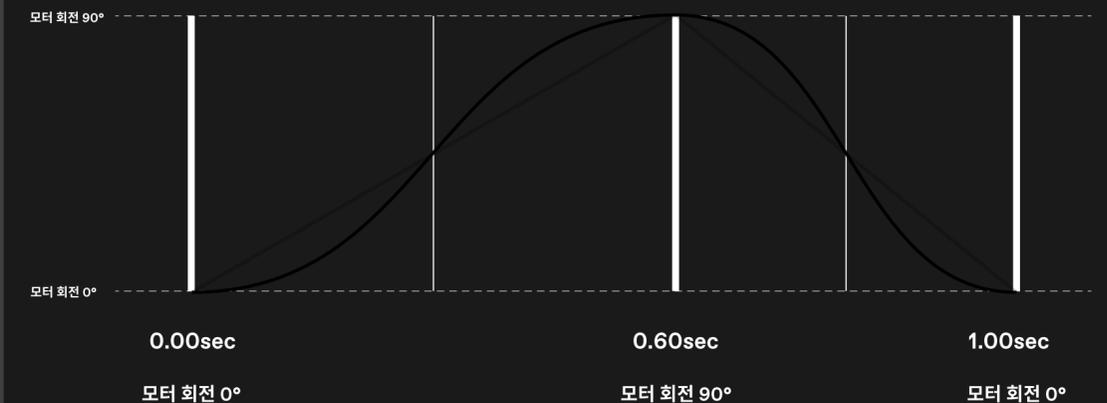
Mehdi Mcharek, Toufik Azib, Cherif Larouci, and Moncef Hammadi, Collaboration and multidisciplinary design optimization for mechatronic systems, IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society, Vol. 1, IEEE, pp. 624-629, 2019.

Mimi Nguyen and Celine Mougnot, A systematic review of empirical studies on multidisciplinary design collaboration: Findings, methods, and challenges, Design Studies, Vol. 81, p. 101120, 2022.

학생 디자이너: 더 자연스럽게, 부드럽게 시작해서 끝나면 좋겠어요



학생 엔지니어: 시간 - 속도 수정하고 사인 보간 넣으면 되겠네요

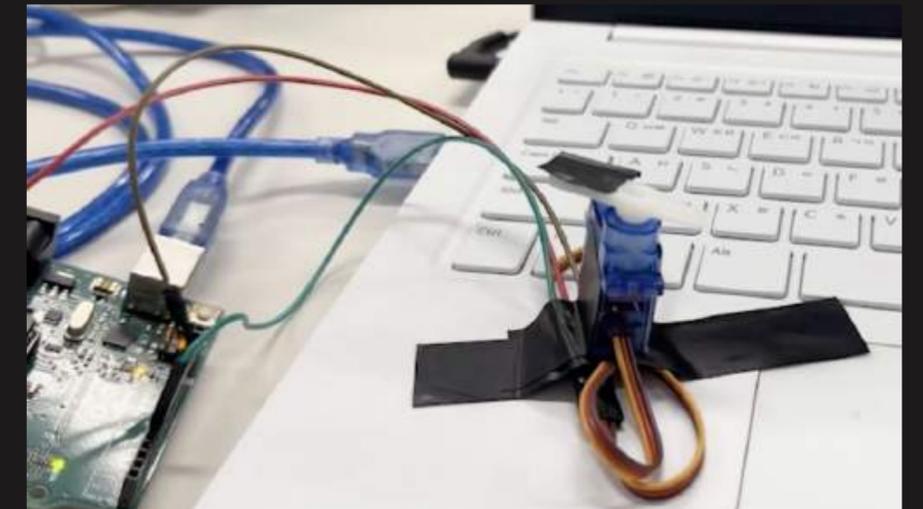
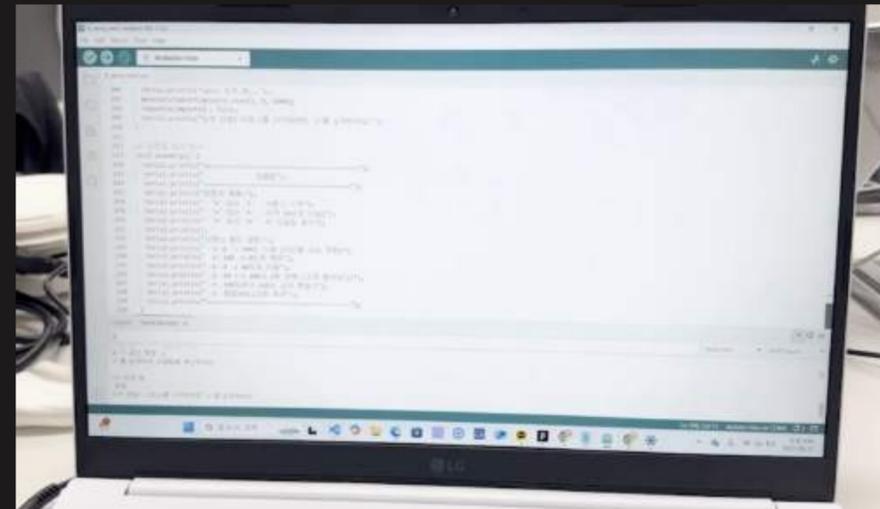


## 문제점 원인 2

# 프로그래밍 환경에 대한 비전공자(디자이너)의 진입장벽

아두이노의 등장으로 프로토타이핑에 대한 접근성이 향상되었음에도, 여전히 C++ 기반의 프로그래밍 환경은 비전공자에게 높은 진입장벽으로 작용하여 디자이너가 직접 움직임을 수정하고 검증하기 어렵게 만든다.

Hari Kishan Kondaveeti, Nandeesh Kumar Kumaravelu, Sunny Dayal Vanambathina, Sudha Ellison Mathe, and Suseela Vappangi, A systematic literature review on prototyping with Arduino: Applications, challenges, advantages, and limitations, Computer Science Review, Vol. 40, p. 100364, 2021.



## 1. 서론

### 1.2. HRI 기반 표현 의도의 반영 필요성

## 1.2. HRI 기반 표현 의도의 반영 필요성

**생동감 있는 움직임의 표현** 이 로봇에서 중요하기 때문

---

Hoffman, G., & Ju, W. (2014). Designing robots with movement in mind. *J. Hum. Robot Interact.*, 3(1), 89–122.

## 1.2. HRI 기반 표현 의도의 반영 필요성

로봇의 움직임은 단순한 기능 수행을 넘어  
로봇에 대한 사용자의 긍정적 태도와 정서적 반응에 직접적인 영향을 미치는 요소

## 1.2. HRI 기반 표현 의도의 반영 필요성

로봇의 움직임 패턴의 설계가 사용자와의 친밀감 형성에 유의미한 영향을 미친다

Nakayama (2016)

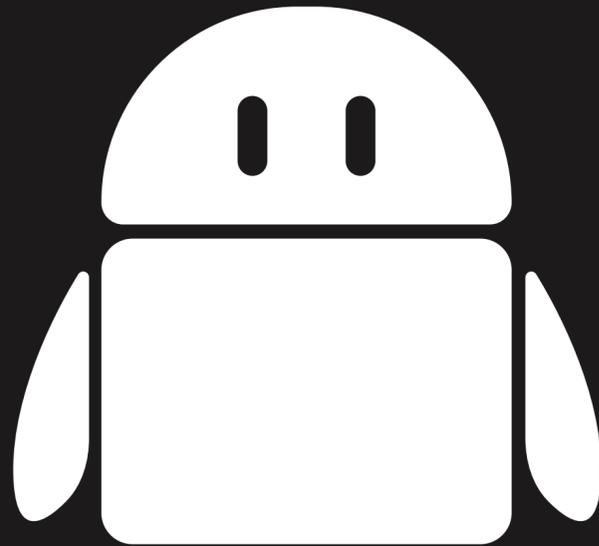
기능적 움직임 (functional movement)과 표현적 움직임 (expressive movement)

의 결합이 로봇에 대한 사용자의 긍정적인식을 유의미하게 향상시키고

상호작용 참여도 (engagement)와 유대감 (sense of connection) 증진에 기여한다

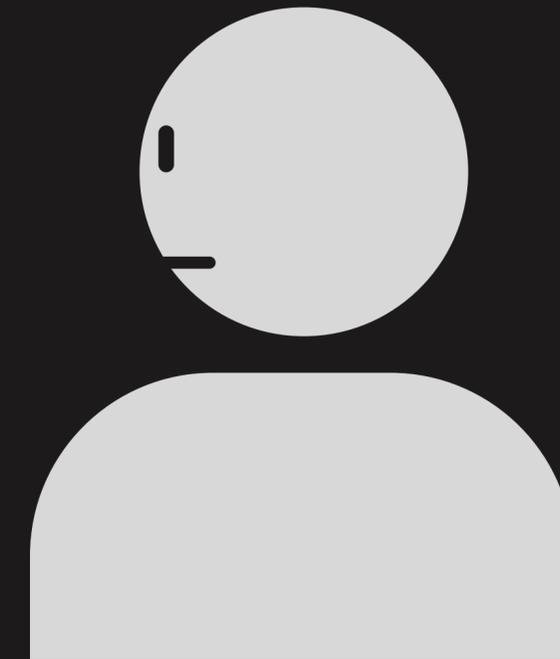
Hu et al. (2025)

# 같은 역할 수행을 하더라도 표현적인 움직임이 있는 로봇이 사용자에게 더 긍정적으로 인식



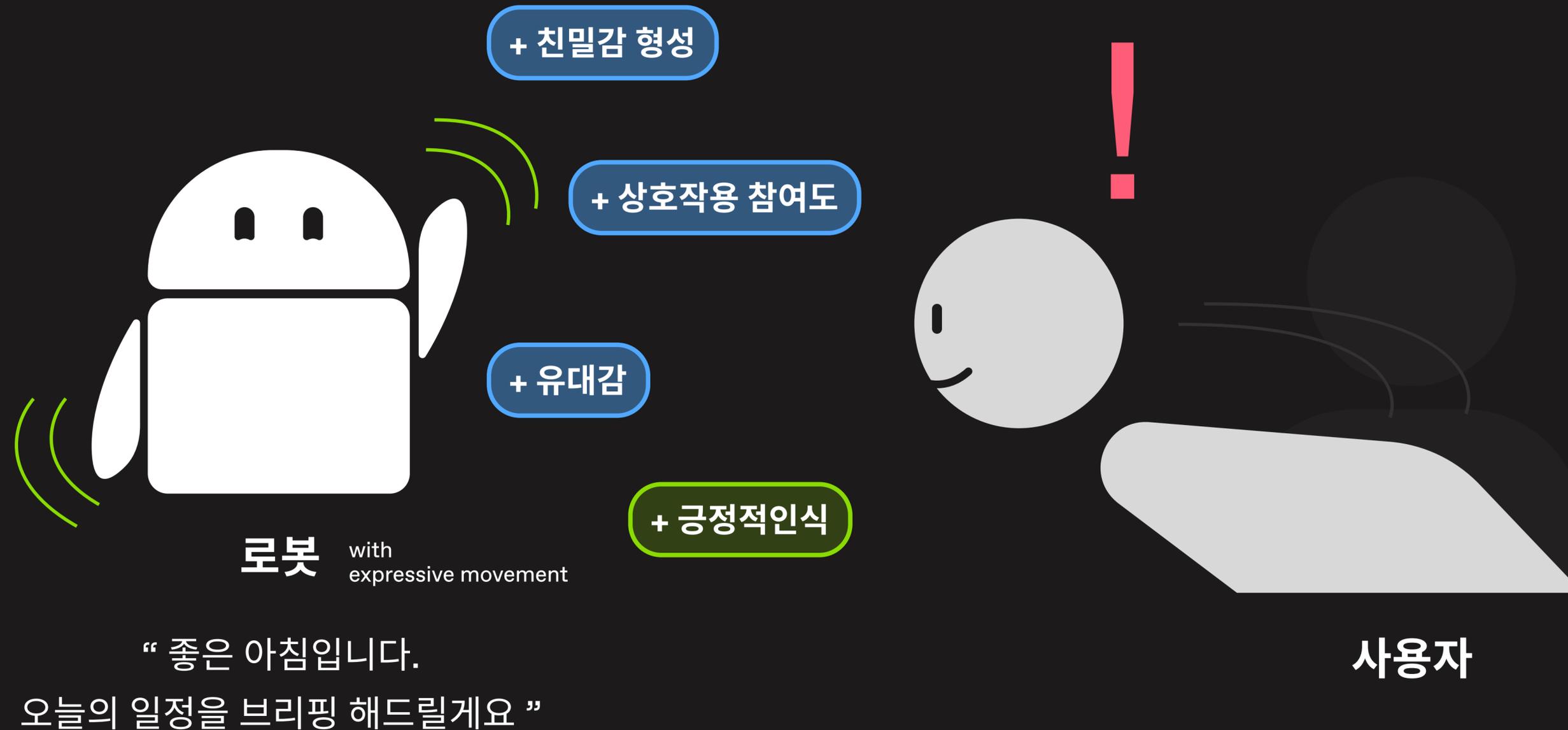
로봇

“ 좋은 아침입니다.  
오늘의 일정을 브리핑 해드릴게요 ”



사용자

# 같은 역할 수행을 하더라도 표현적인 움직임이 있는 로봇이 사용자에게 더 긍정적으로 인식



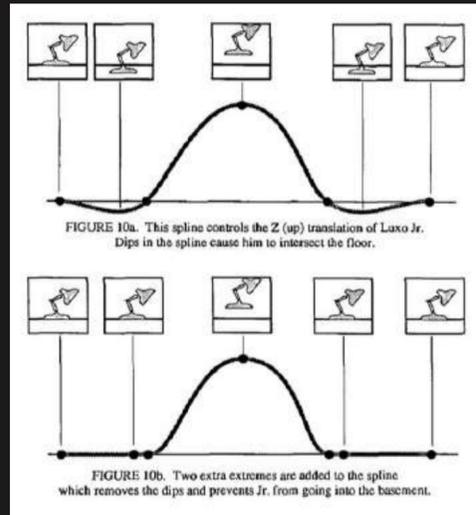
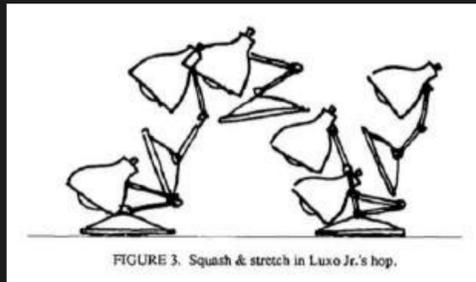
## 2. 문제 접근법

HOW

**로봇의 움직임에 생동감을 부여, 어떻게?**

# HOW

## 로봇의 움직임에 생동감을 부여, 어떻게?



## 애니메이션의 12가지 원리 활용

# 전통 애니메이션 원리를 3D 애니메이션에 활용 (퍼펫)

# 3D 애니메이션 캐릭터의 생명감 → 로봇에게 적용 가능성

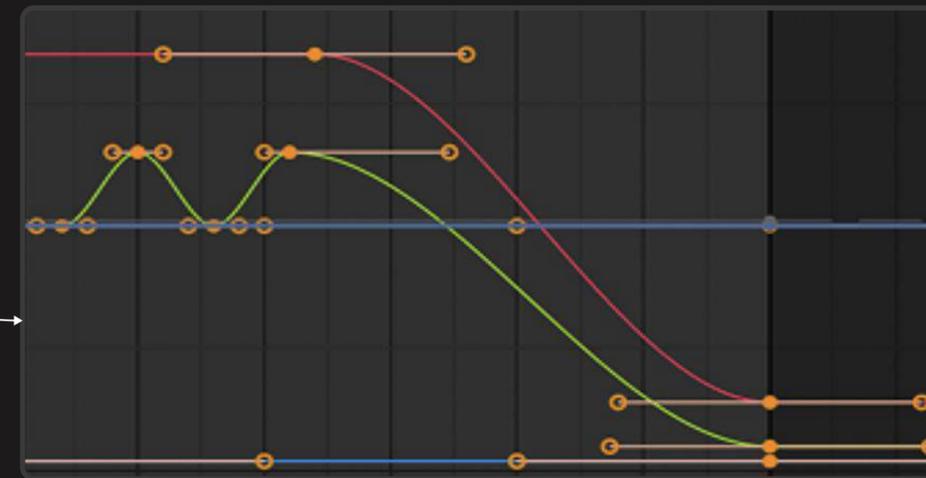
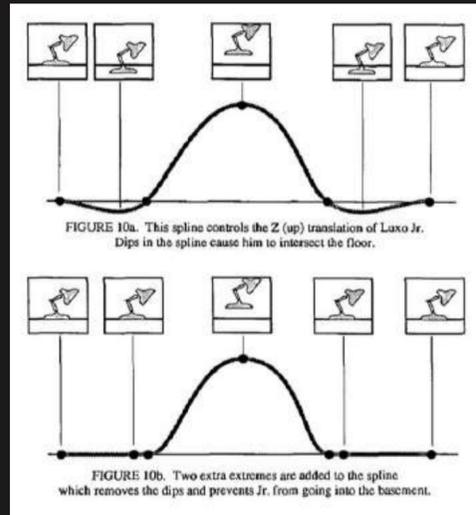
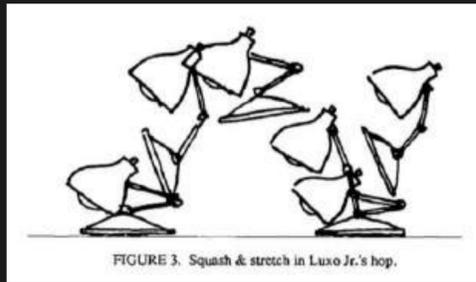
Lasseter(1987) 는 동작의 완급을 조절하는 'Slow in and out', 곡선형 궤적으로 자연스러운 흐름을 만드는 'Arcs', 주동작에 따라 보조적으로 발생하는 움직임을 통해 사실감을 부여하는 'Secondary Action' 등의 원리를 제시하였다.

이러한 원리는 3D 애니메이션 캐릭터의 생명감을 표현하기 위해 제안되었으나, 움직임의 속도 변화·궤적·보조 반응을 설계하는 개념으로서 로봇의 움직임 설계에도 적용 가능한 기준으로 볼 수 있다.

Lasseter, J. (1987). Principles of traditional animation applied to 3D computer animation. ACM SIGGRAPH Computer Graphics, 21(4), 35-44, <https://doi.org/10.1145/37402.37407>

# HOW

## 로봇의 움직임에 생동감을 부여, 어떻게?



## 애니메이션의 12가지 원리 활용

# 전통 애니메이션 원리를 3D 애니메이션에 활용 (퍼펫)

# 3D 애니메이션 캐릭터의 생명감 → 로봇에게 적용 가능성

Lasseter, J. (1987). Principles of traditional animation applied to 3D computer animation. ACM SIGGRAPH Computer Graphics, 21(4), 35-44, <https://doi.org/10.1145/37402.37407>

## 현업 사례 1

# BDX Droid

# 애니메이션 기반 접근법

# 모션디자이너와의 협업

모션 디자이너와 엔지니어가 협업하여 개발한 디즈니 리서치의 이족보행 로봇 'BDX Droid'는 시각적으로 생동감 있는 움직임을 물리적 환경에서 구현하였다. 이 연구는 3D 애니메이션 소프트웨어(Maya)에서 제작된 움직임 데이터를 활용하여, 강화학습을 통해 로봇이 해당 움직임을 재현하도록 학습시켰다.



Grandia, R., Knoop, E., Hopkins, M. A., Wiedebach, G., Bishop, J., Pickles, S., Müller, D., & Bäcker, M. (2025). Design and control of a bipedal robotic character. arXiv preprint arXiv:2501.05204, Retrieved from <https://arxiv.org/abs/2501.05204>

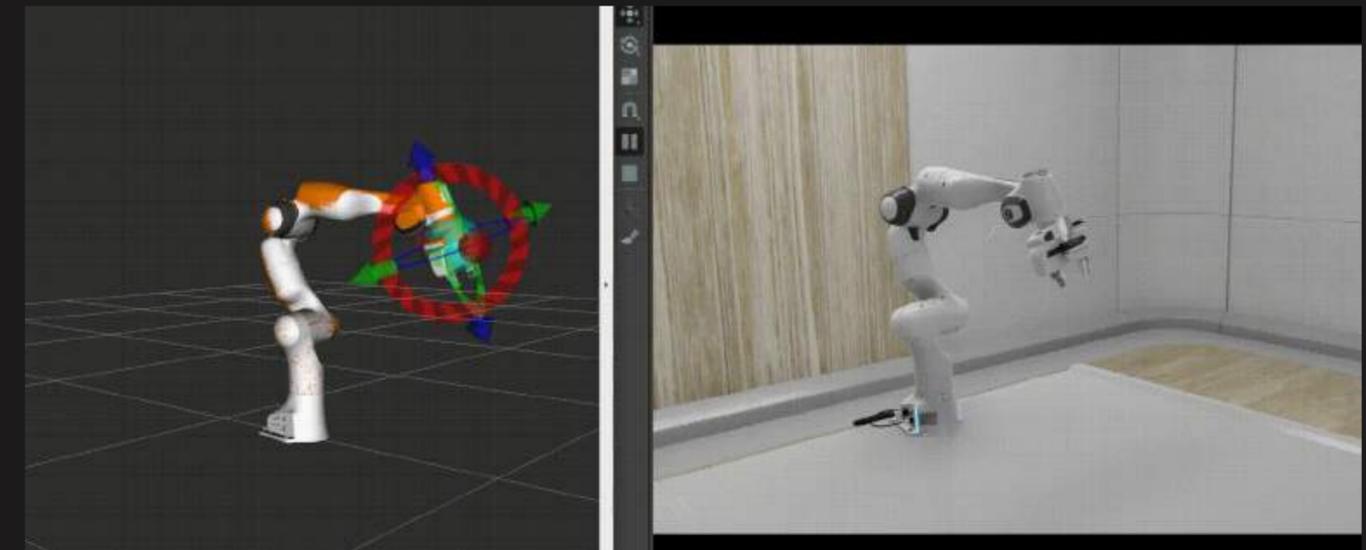
## 현업 사례 2

# Movelt

# 디자이너-엔지니어 간 협업 지원

# 간극 감축 연구 사례

디자이너-엔지니어 간 협업을 지원하거나 간극을 줄이기 위한 연구에서는 Movelt을 기반 기술로 활용하여, ROS 기반의 3D 디지털 트윈 환경에서 로봇을 설계하고 시각적으로 직관적인 인터페이스를 통해 움직임을 제작하는 방식을 제안하였다. 이러한 접근은 로봇 제어의 운동학 (Kinematic)적 수식들을 시각화 하여 사용자 접근성을 높였다는 점에서 의의를 가진다.



Yeping Wang, Alexander Peseckis, Zelong Jiang, and Michael Gleicher, Motion comparator: Visual comparison of robot motions, IEEE Robotics and Automation Letters, 2024.  
Andoni Rivera-Pinto, Johan Kildal, and Elena Lazkano, Toward programming a collaborative robot by interacting with its digital twin in a mixed reality environment, International Journal of Human-Computer Interaction, Vol. 40, No. 17, pp. 4745-4757, 2024.  
Pablo Malvido Fresnillo, Saigopal Vasudevan, Wael M. Mohammed, Jose L. Martinez Lastra, and Jose A. Perez Garcia, Extending the motion planning framework Movelt with advanced manipulation functions for industrial applications, Robotics and Computer-Integrated Manufacturing, Vol. 83, p. 102559, 2023.

## BDX Droid

# 애니메이션 기반 접근법

# 모션디자이너와의 협업

모션 디자이너와 엔지니어가 협업하여 개발한 디즈니 리서치의 이족보행 로봇 'BDX Droid'는 시각적으로 생동감 있는 움직임을 물리적 환경에서 구현하였다. 이 연구는 3D 애니메이션 소프트웨어(Maya)에서 제작된 움직임 데이터를 활용하여, 강화학습을 통해 로봇이 해당 움직임을 재현하도록 학습시켰다.

## Movelt

# 디자이너-엔지니어 간 협업 지원

# 간극 감축 연구 사례

디자이너-엔지니어 간 협업을 지원하거나 간극을 줄이기 위한 연구에서는 Movelt을 기반 기술로 활용하여, ROS 기반의 3D 디지털 트윈 환경에서 로봇을 설계하고 시각적으로 직관적인 인터페이스를 통해 움직임을 제작하는 방식을 제안하였다. 이러한 접근은 로봇 제어의 운동학(Kinematic)적 수식들을 시각화 하여 사용자 접근성을 높였다는 점에서 의의를 가진다.



그러나 두 사례 모두 로봇 엔지니어를 필요로 하는 개발 과정을 포함하고 있으며, 강화 학습을 위한 데이터 추출 단계에서 활용되거나 ROS(Robot Operating System)를 기반으로 한 리눅스 환경 구축과 코딩 지식이 요구

이에,

디자이너에게 익숙한 3D 애니메이션 환경에서 제작된  
키프레임 애니메이션 데이터를 직접 변환하여  
로봇의 모터 제어에 적용할 수 있는 프레임워크를 제안

# Frame Work

디자이너에게 익숙한 3D 애니메이션 환경에서 제작된  
키프레임 애니메이션 데이터를 직접 변환하여  
로봇의 모터 제어에 적용할 수 있는 프레임워크

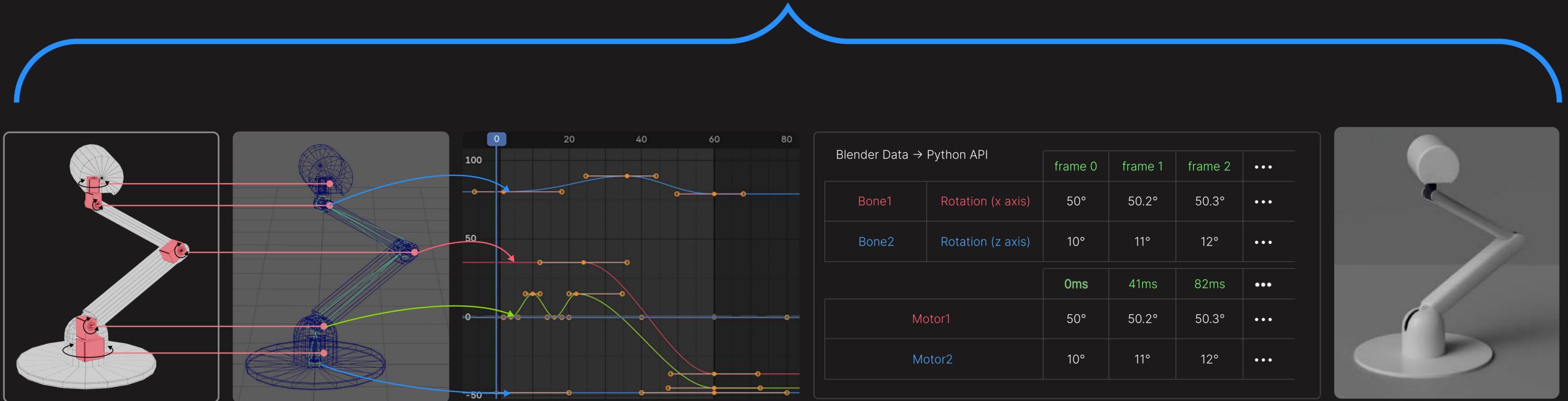
# Robot Motion

- 애니메이션의 12가지 원리 활용
- 디자이너 개입 영역 확장

## 3. 프레임워크 개요 및 구현 방법

### 3.1 개념

## Framework Overview



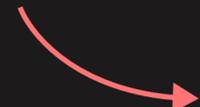
로봇 구조 및 모터 배치

3D환경에서 리깅

애니메이션 제작

애니메이션 데이터 추출 및 변환

모터로 데이터 전송



### 3.1 개념

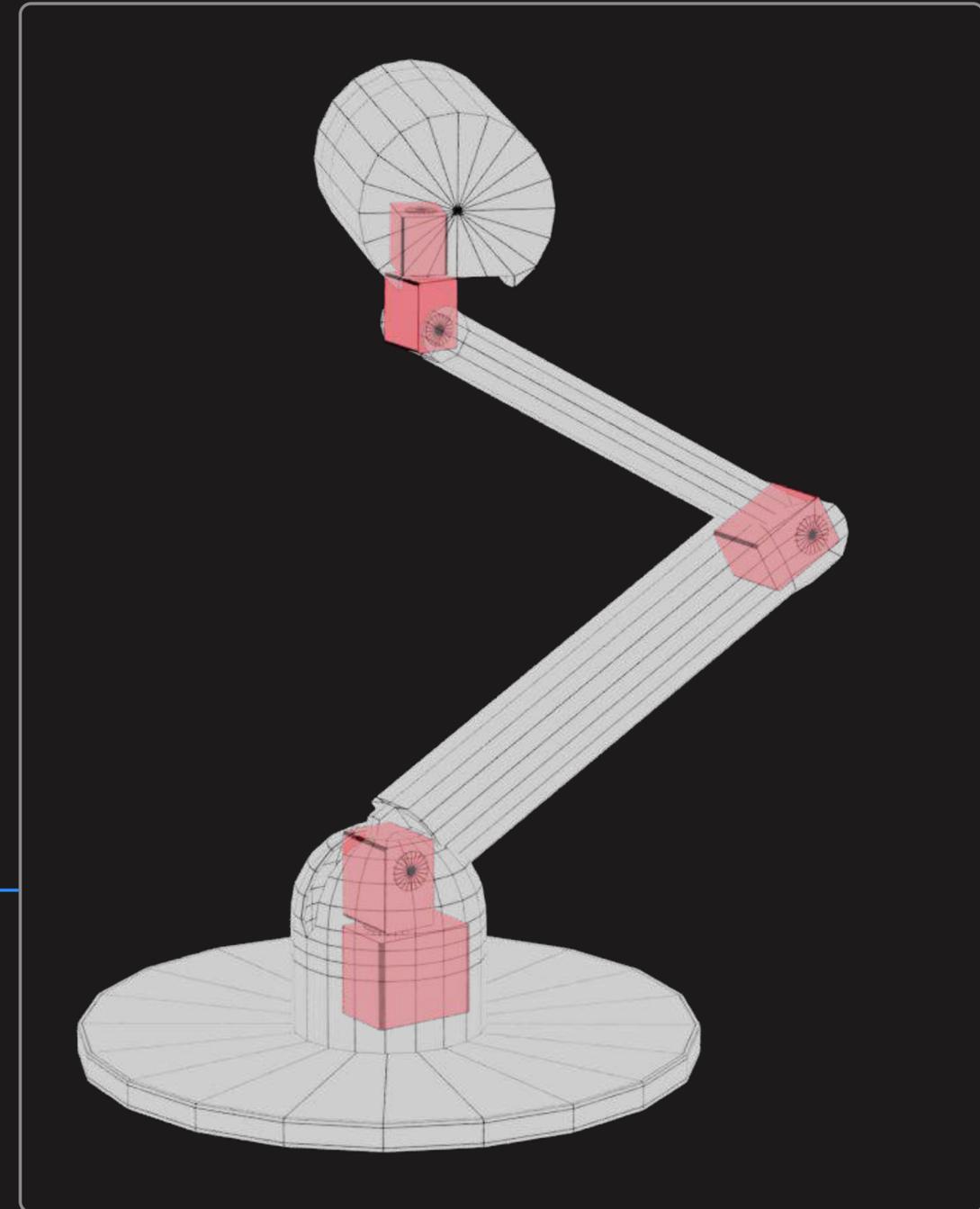
## 로봇의 모터구조와 3D 환경에서 모델링과 리깅

- a. 모델링 = 셸: 로봇의 외형 Form
- b. 리깅 = 모터 : 뼈대Bone (3D 환경에서의 관절)이 셸과 연결된다.

모터가 움직이면 셸이 움직인다.

구조상의 연결의 위계가 있어, 상위의 구조가 움직이면

그에 연결된 구조가 함께 움직인다.



로봇의 셸과 모터

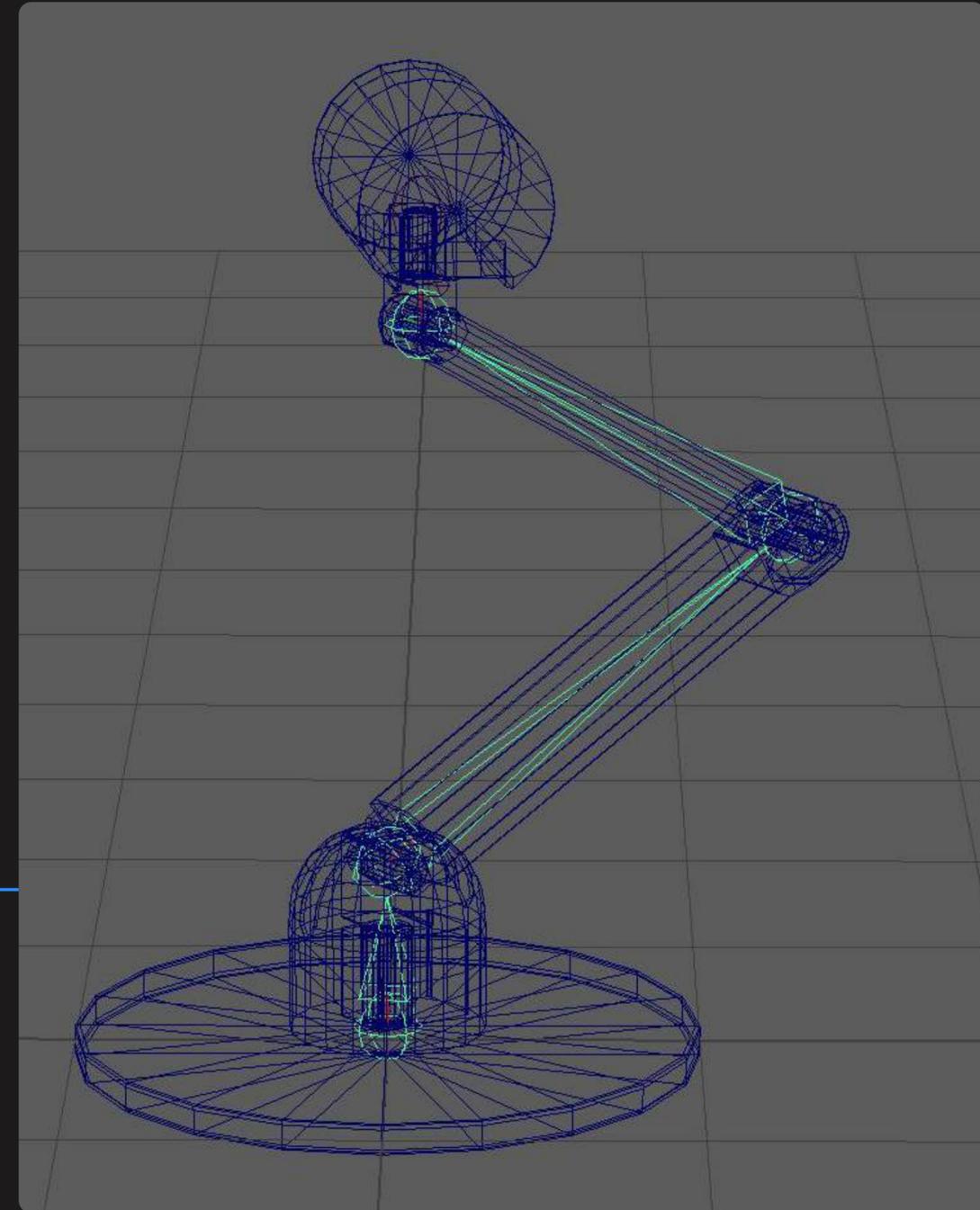
### 3.1 개념

## 로봇의 모터구조와 3D 환경에서 모델링과 리깅

- a. 모델링 = 셸: 로봇의 외형 Form
- b. 리깅 = 모터 : 뼈대Bone (3D 환경에서의 관절)이 셸과 연결된다.

Bone이 움직이면 메시가 움직인다.

Bone끼리는 연결의 위계가 있어, 상위의 Bone이 움직이면 그에 연결된 Bone(및 메시)이 함께 움직인다.

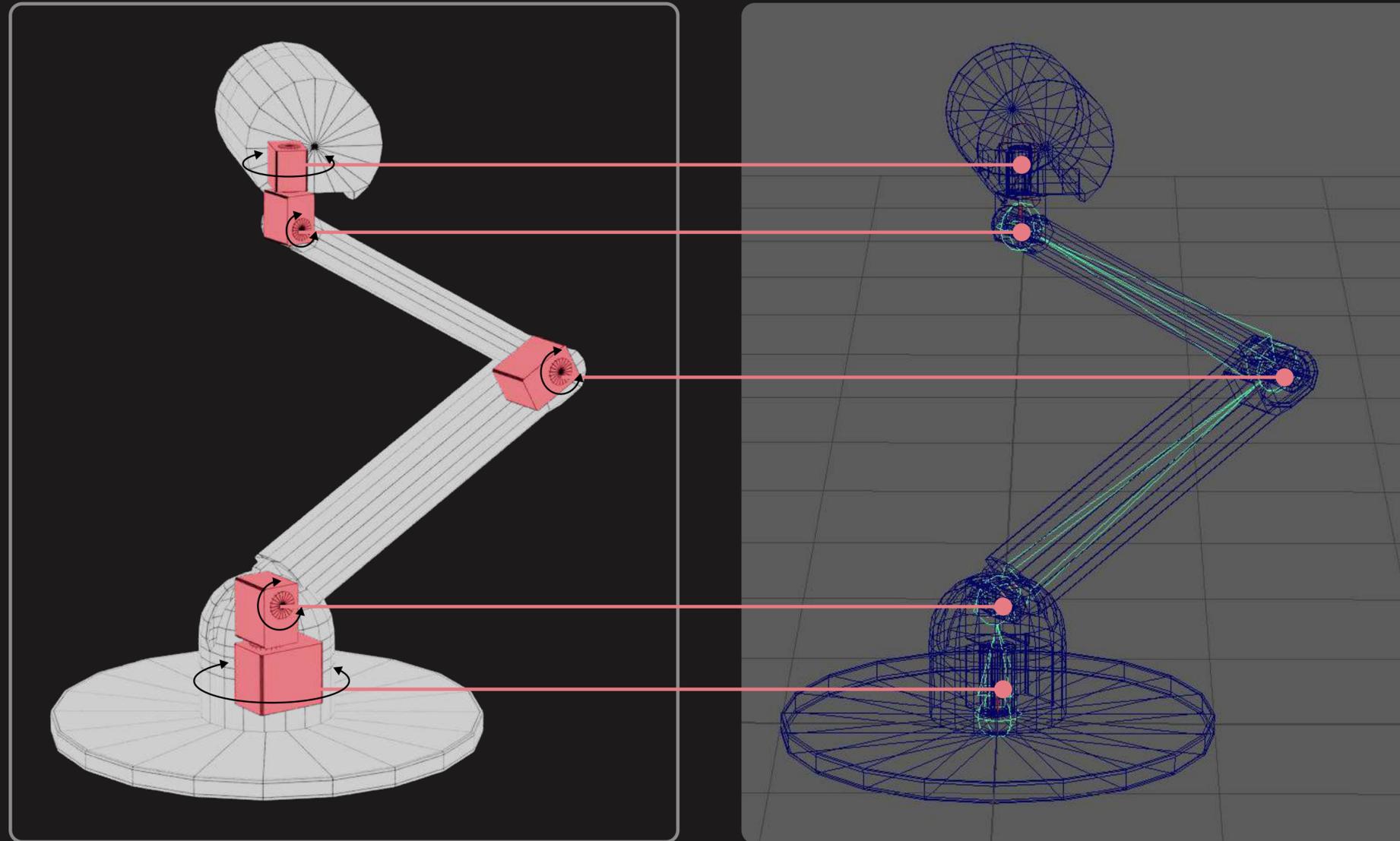


로봇의 모델링과 리깅된 뼈대(Bone) - 3D 프로그램

### 3.1 개념

Key Point !

## 모터 회전 방향 - 리그 회전축 일치시키기

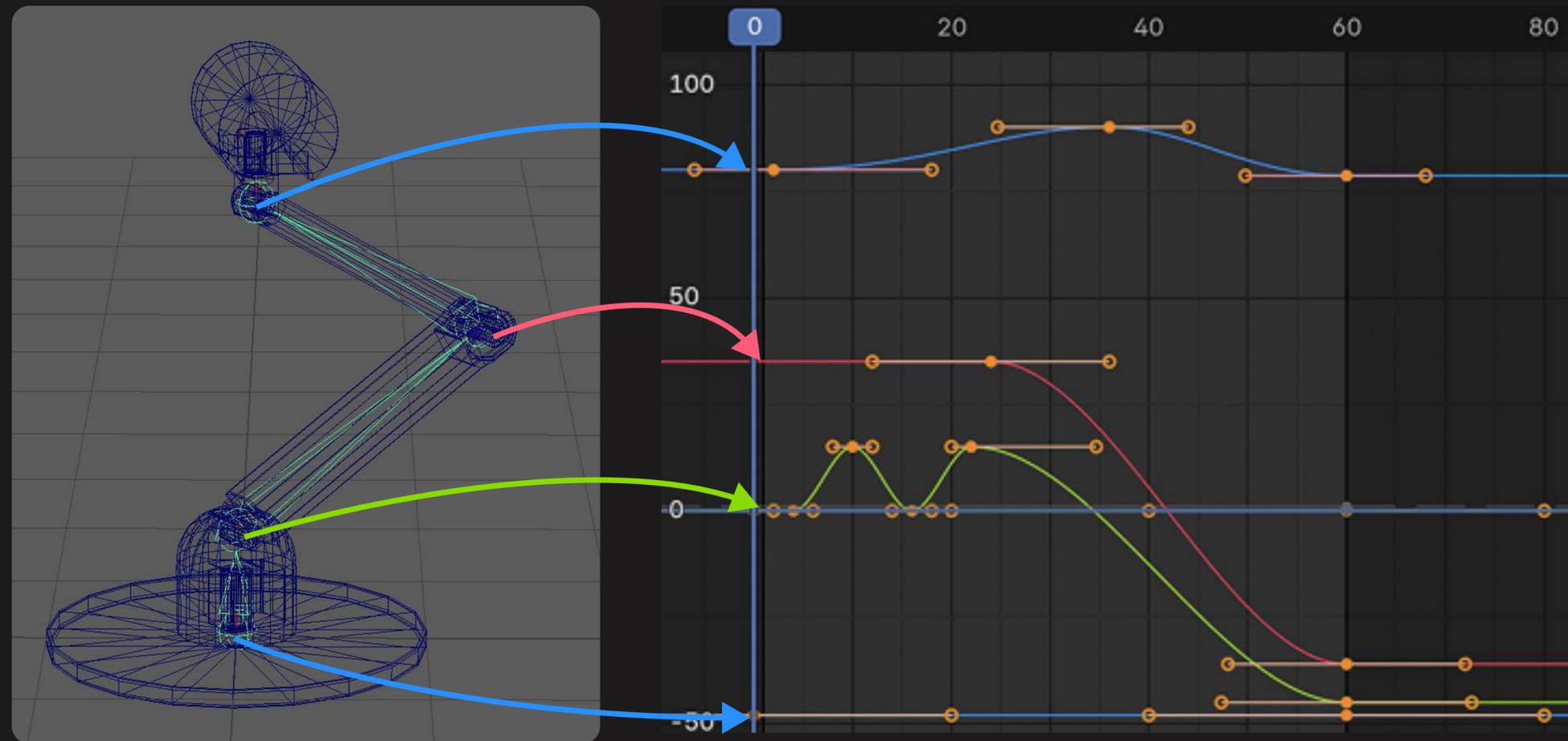


모터와 관절 대응: 하나의 모터는 하나의 축으로 회전한다. 그러나 기본적인 리그의 Bone은 X,Y,Z축 모두로 움직임  
→ 따라서 모터 회전 방향에 맞춰 각 축 중 일치하는 것만 남기고 Lock

## 3.1 개념

## 리그된 환경에서 3D 애니메이션 프로그램의 인터페이스를 활용해 모션 제작

→ 그래프 에디터, 키프레임 에디터 등 블렌더 애니메이션 기능 활용 가능



### 3.2 애니메이션 기반 움직임 설계 과정

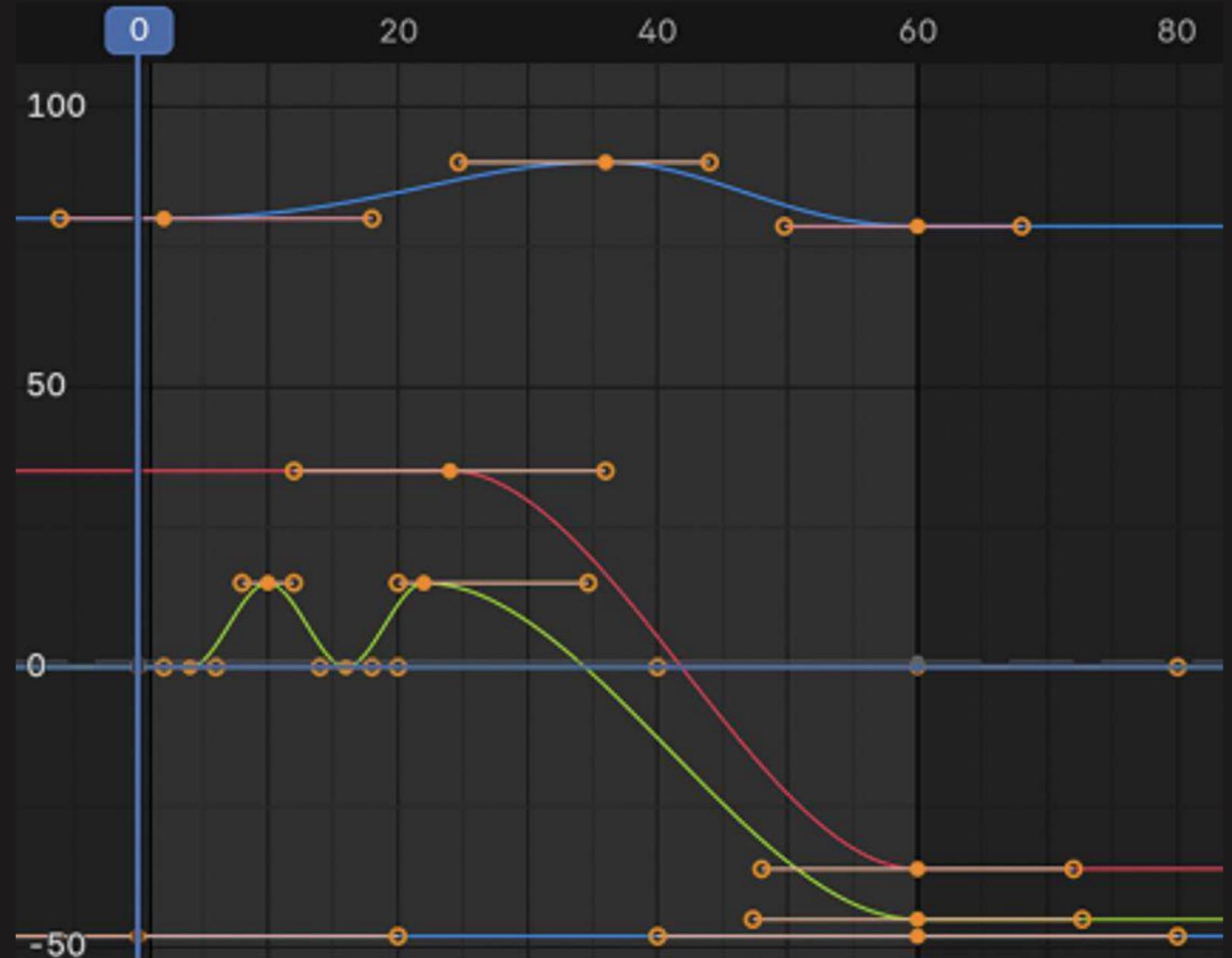
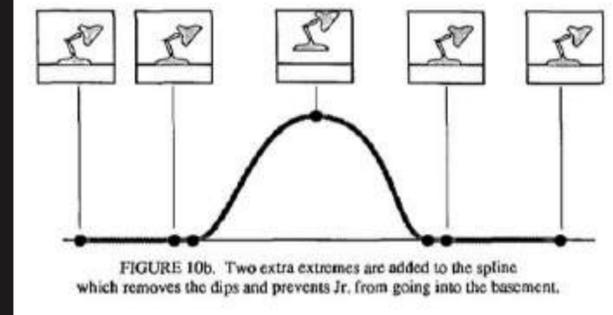
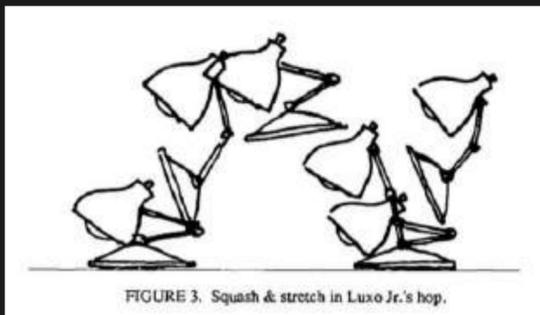
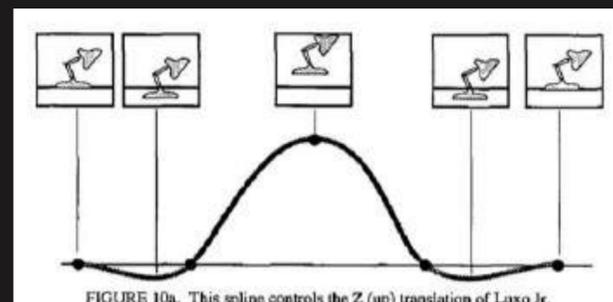
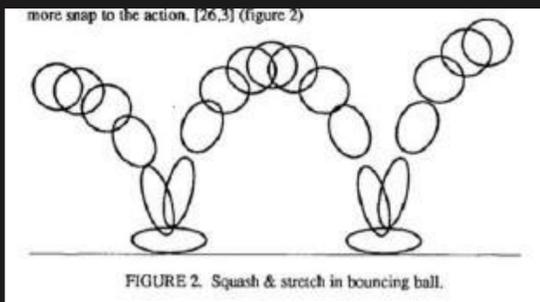
## 따라서, 로봇 움직임에 애니메이션 원리 반영이 용이

Lasseter(1987)의 애니메이션 원리에 기반하여

타이밍(Timing)과 아크(Arc)를 반영하여 각 키프레임을 설정

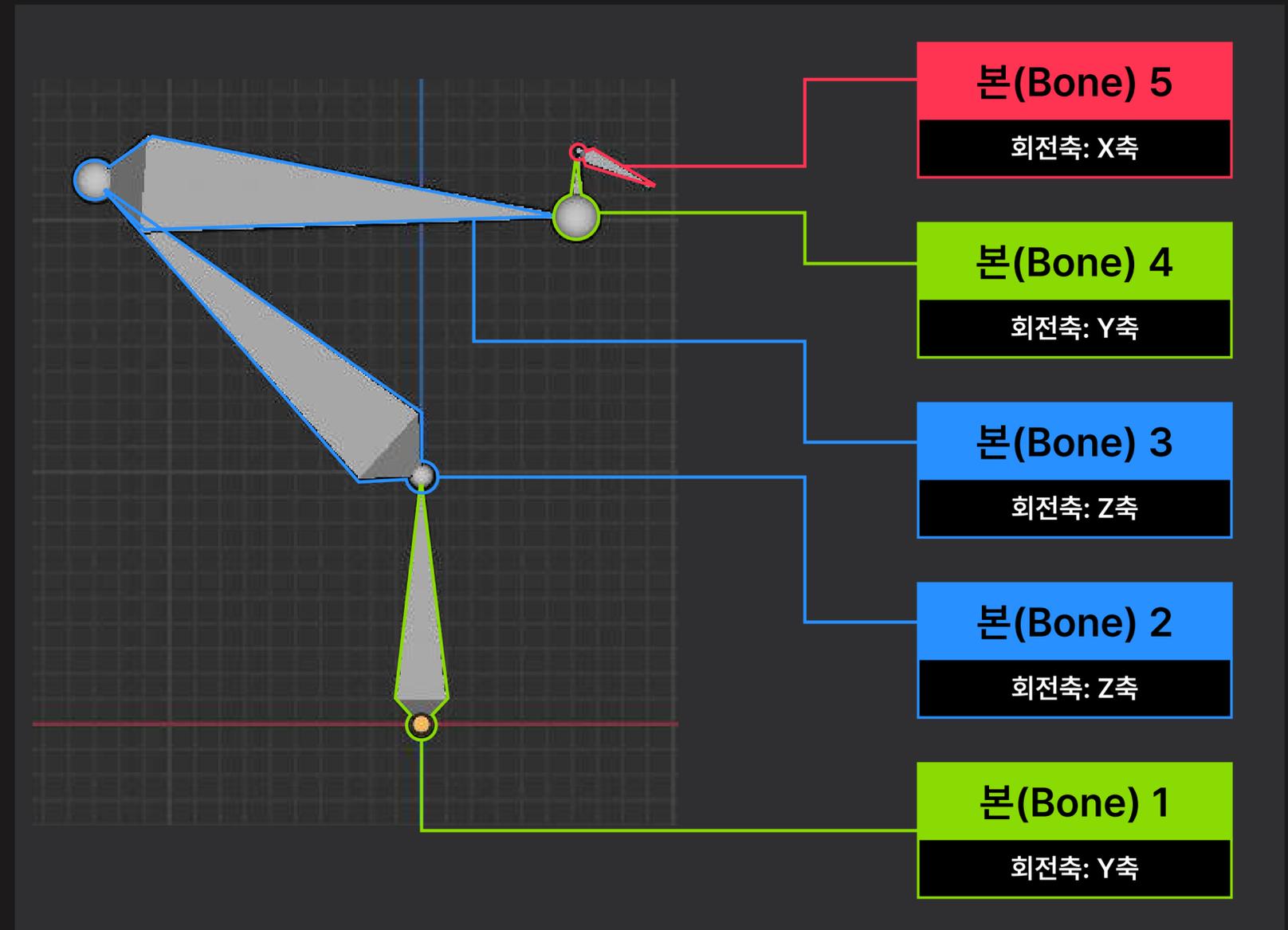
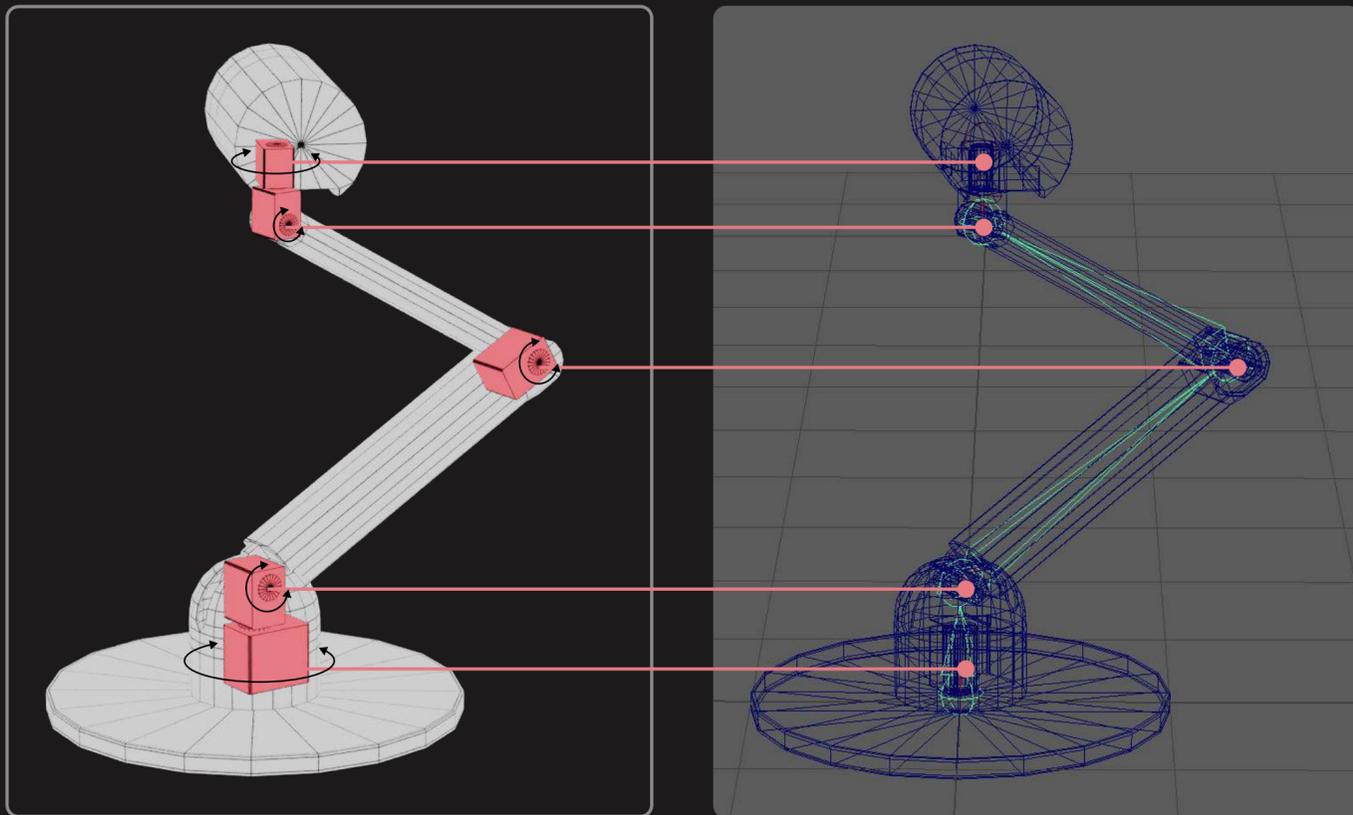
그래프 편집기(Graph Editor)에서 키프레임 간의 곡선 보간을 조정

→ 완급 조절(ease-in/out)과 곡선형 움직임(motion arc)을 구현



### 3.2 애니메이션 기반 움직임 설계 과정

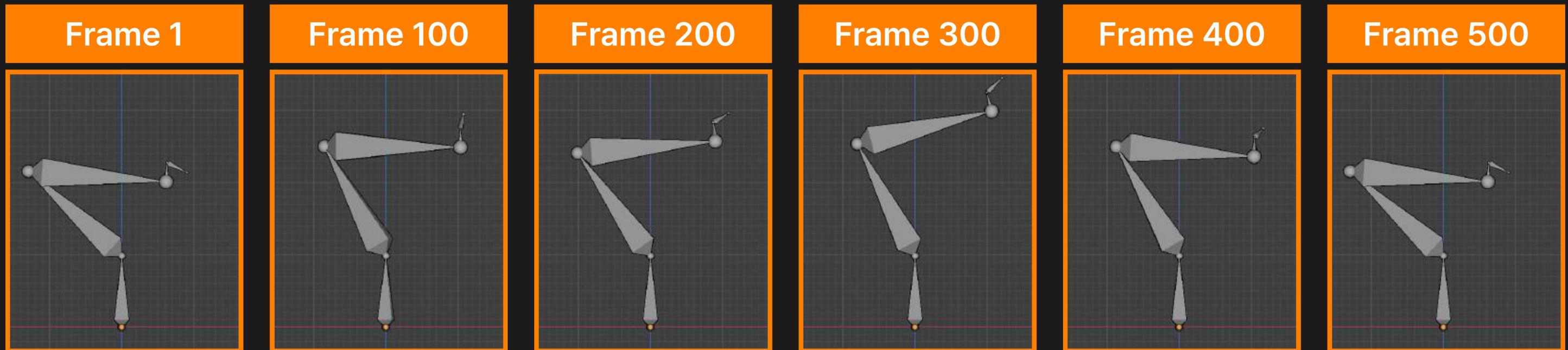
## 예시 로봇 시나리오 : 춤추는 듯이 움직이기



### 3.2 애니메이션 기반 움직임 설계 과정

## 예시 로봇 시나리오 : 춤추는 듯이 움직이기

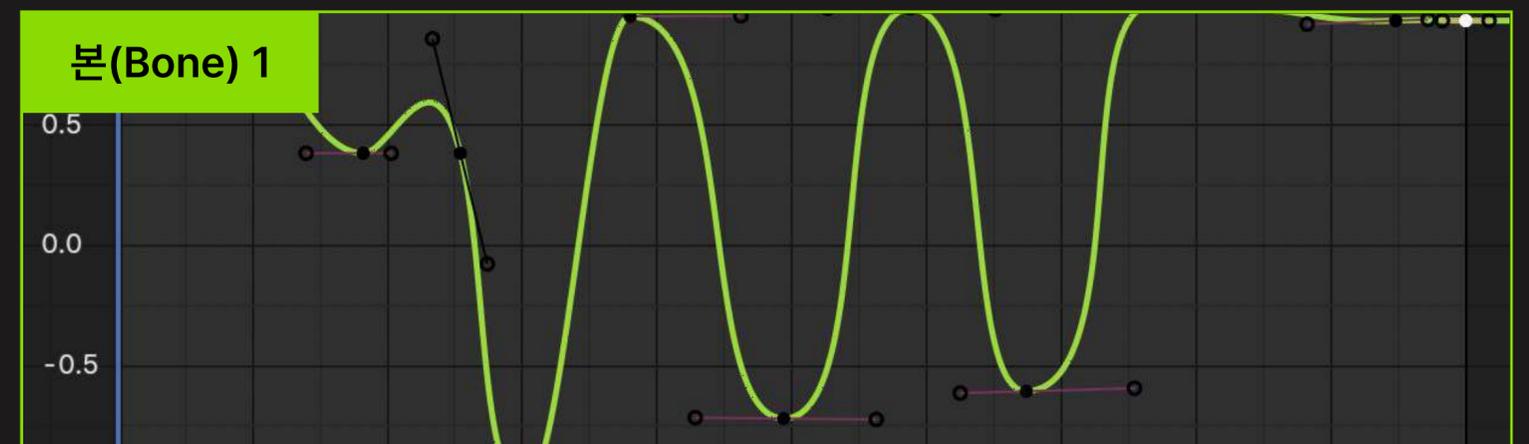
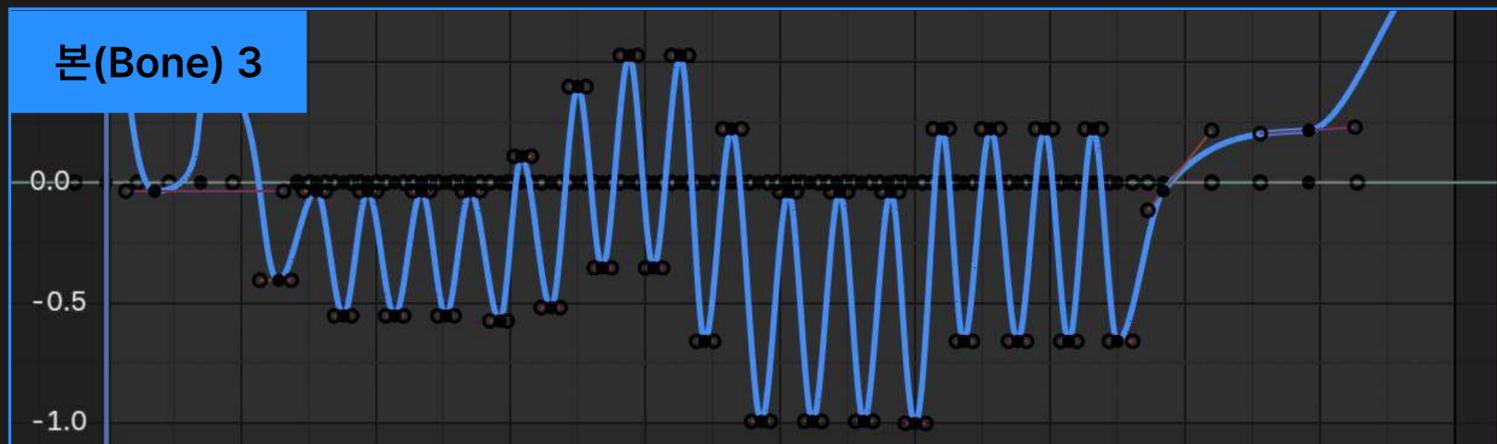
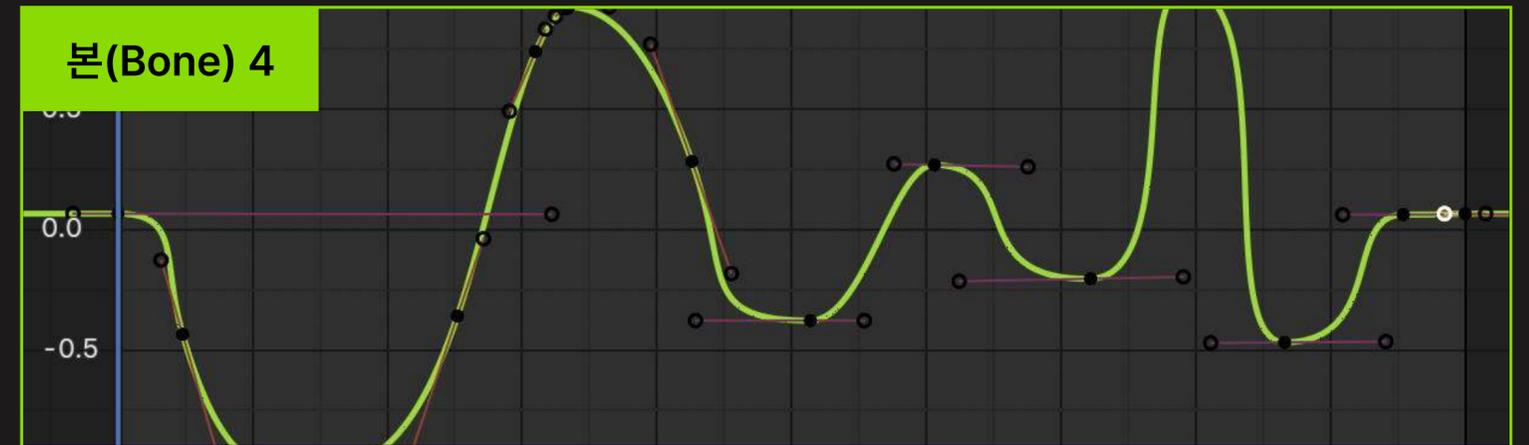
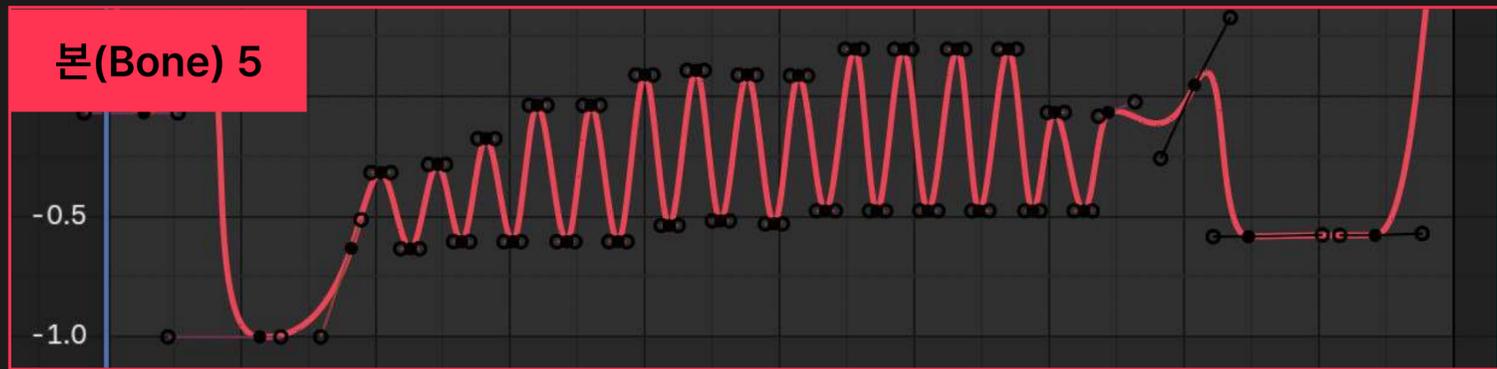
몸통과 고개를 위-아래 바운스하며, 4박에 맞춰 좌우로 움직인다



### 3.2 애니메이션 기반 움직임 설계 과정

## 예시 로봇 시나리오 : 춤추는 듯이 움직이기

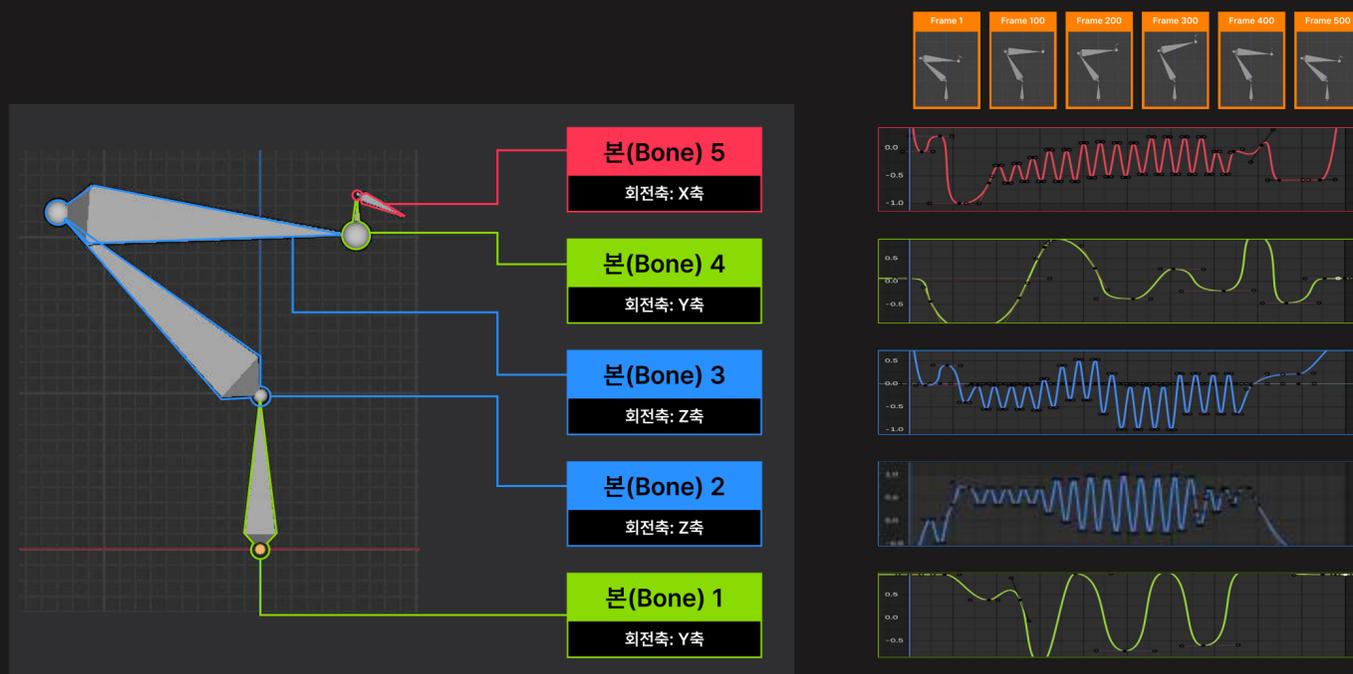
몸통과 고개를 위-아래 바운스하며, 4박에 맞춰 좌우로 움직인다



### 3.2 애니메이션 기반 움직임 설계 과정

# 예시 로봇 시나리오 : 춤추는 듯이 움직이기

몸통과 고개를 위-아래 바운스하며, 4박에 맞춰 좌우로 움직인다



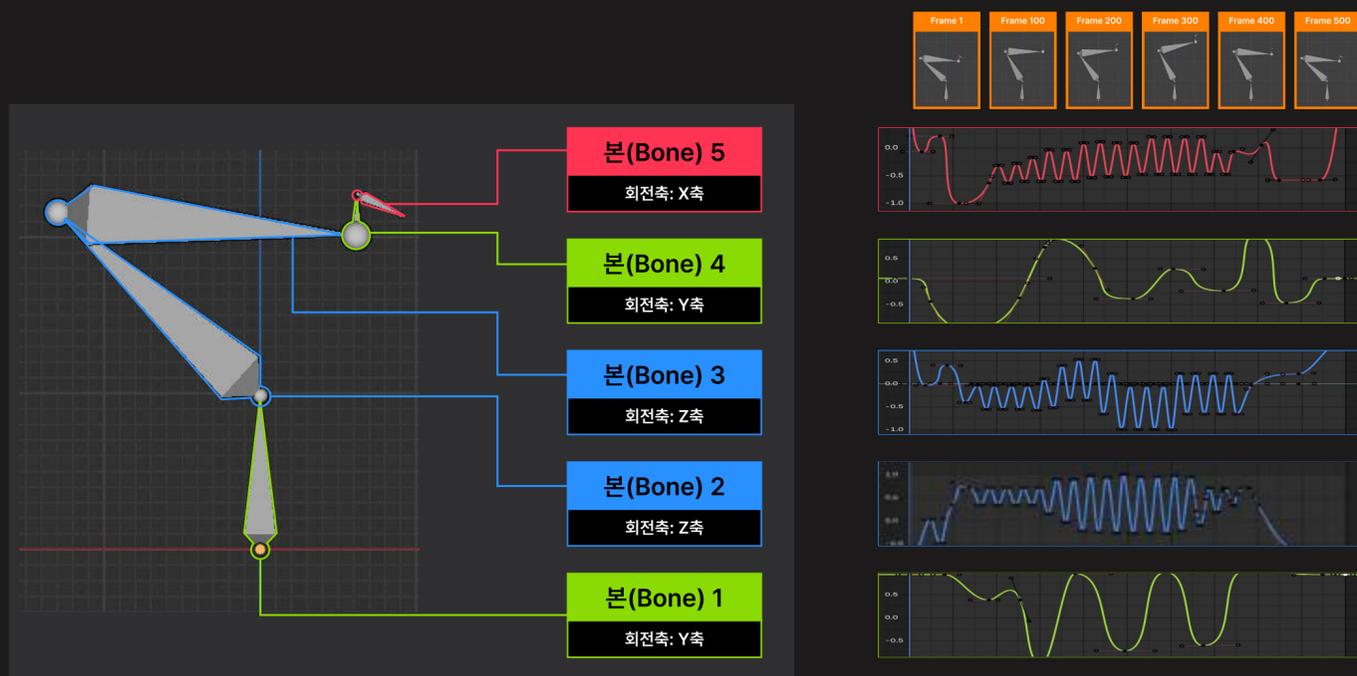
각 본(Bone)의 키프레임에 애니메이션 정보인 회전값이 기록됨

Blender Data		frame 0	frame 1	frame 2	...
Bone1	Rotation (x axis)	50°	50.2°	50.3°	...
Bone2	Rotation (z axis)	10°	11°	12°	...

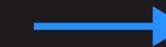
### 3.2 애니메이션 기반 움직임 설계 과정

# 예시 로봇 시나리오 : 춤추는 듯이 움직이기

몸통과 고개를 위-아래 바운스하며, 4박에 맞춰 좌우로 움직인다



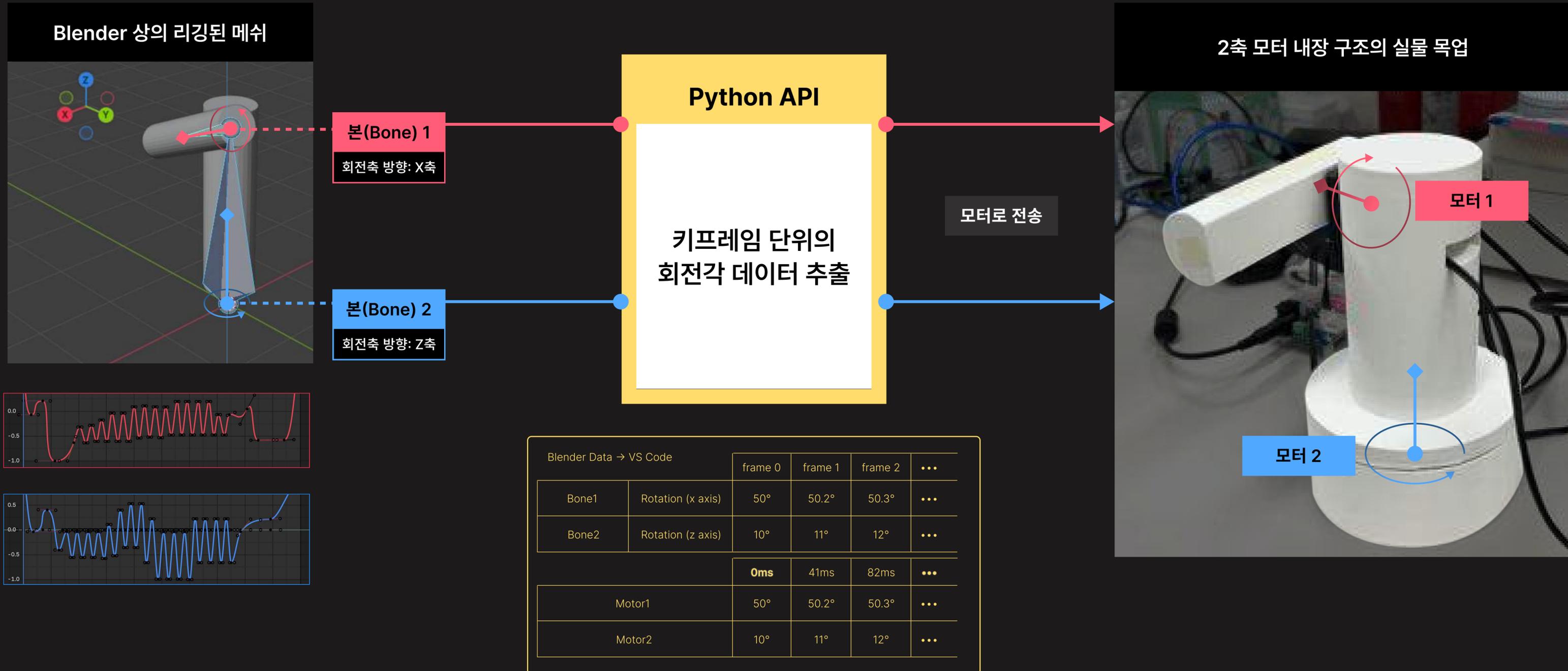
각 본(Bone)의 키프레임에 애니메이션 정보인 회전값이 기록됨



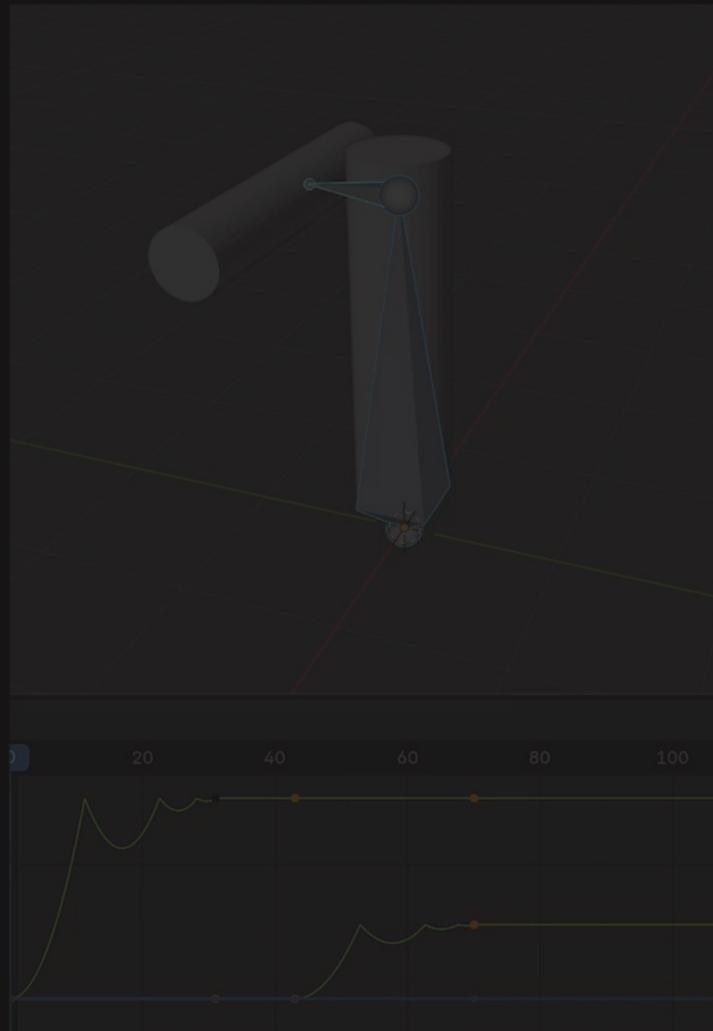
모터에 적용할 수 있는 데이터로 변환

Blender Data → Python API		frame 0	frame 1	frame 2	...
Bone1	Rotation (x axis)	50°	50.2°	50.3°	...
Bone2	Rotation (z axis)	10°	11°	12°	...
		0ms	41ms	82ms	...
Motor1		50°	50.2°	50.3°	...
Motor2		10°	11°	12°	...

### 3.3 구현 예시



### 3.3 구현 예시



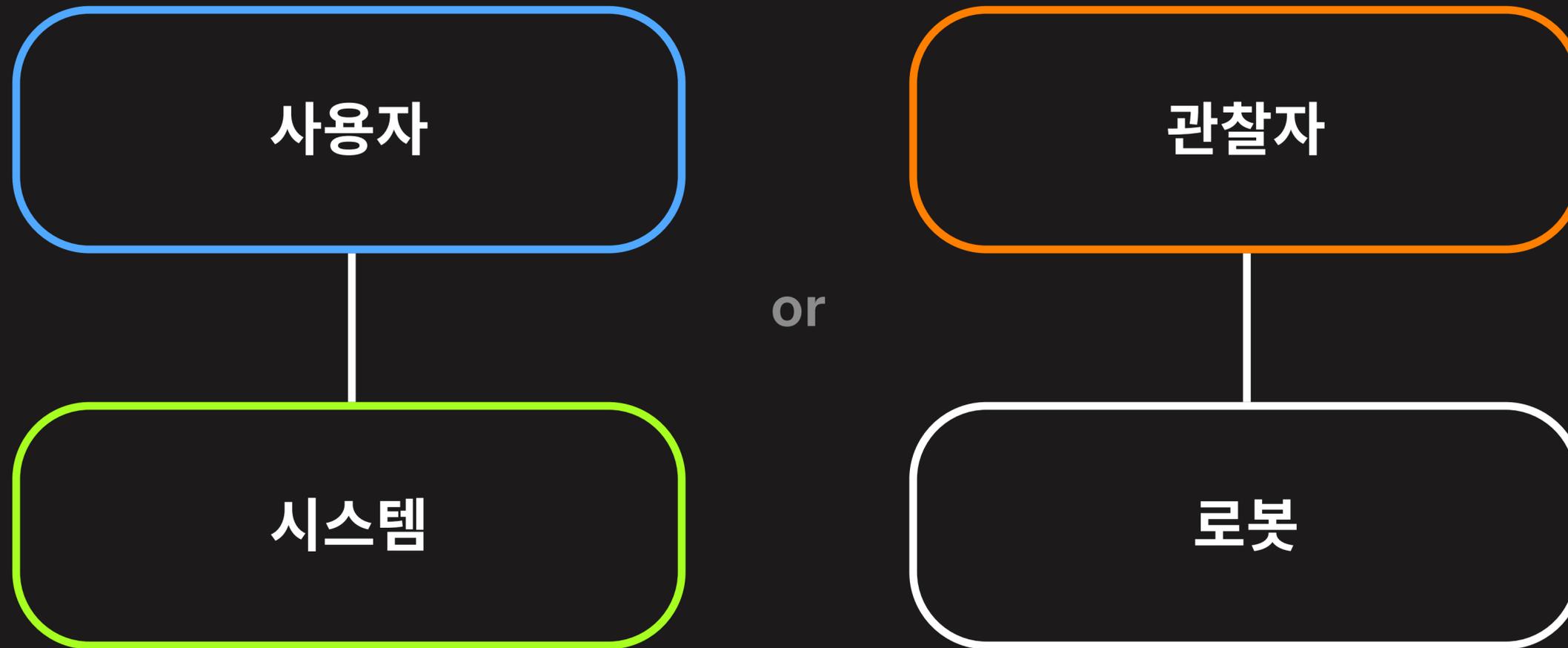
## 4. 프레임워크 기반 확장 리서치

## 4.1 리서치 설계

# 움직임 설계 경험 → 감성 인식 영향

Q. 사용자가 느낀 사용성은 관찰자가 느낀 감성 인식에 어떤 영향을 미칠까?

## 4.1 리서치 설계



## 4.1 리서치 설계

### 연구 구조 확장

Q. 사용자가 느낀 사용성은 관찰자가 느낀 감성 인식에 어떤 영향을 미칠까?

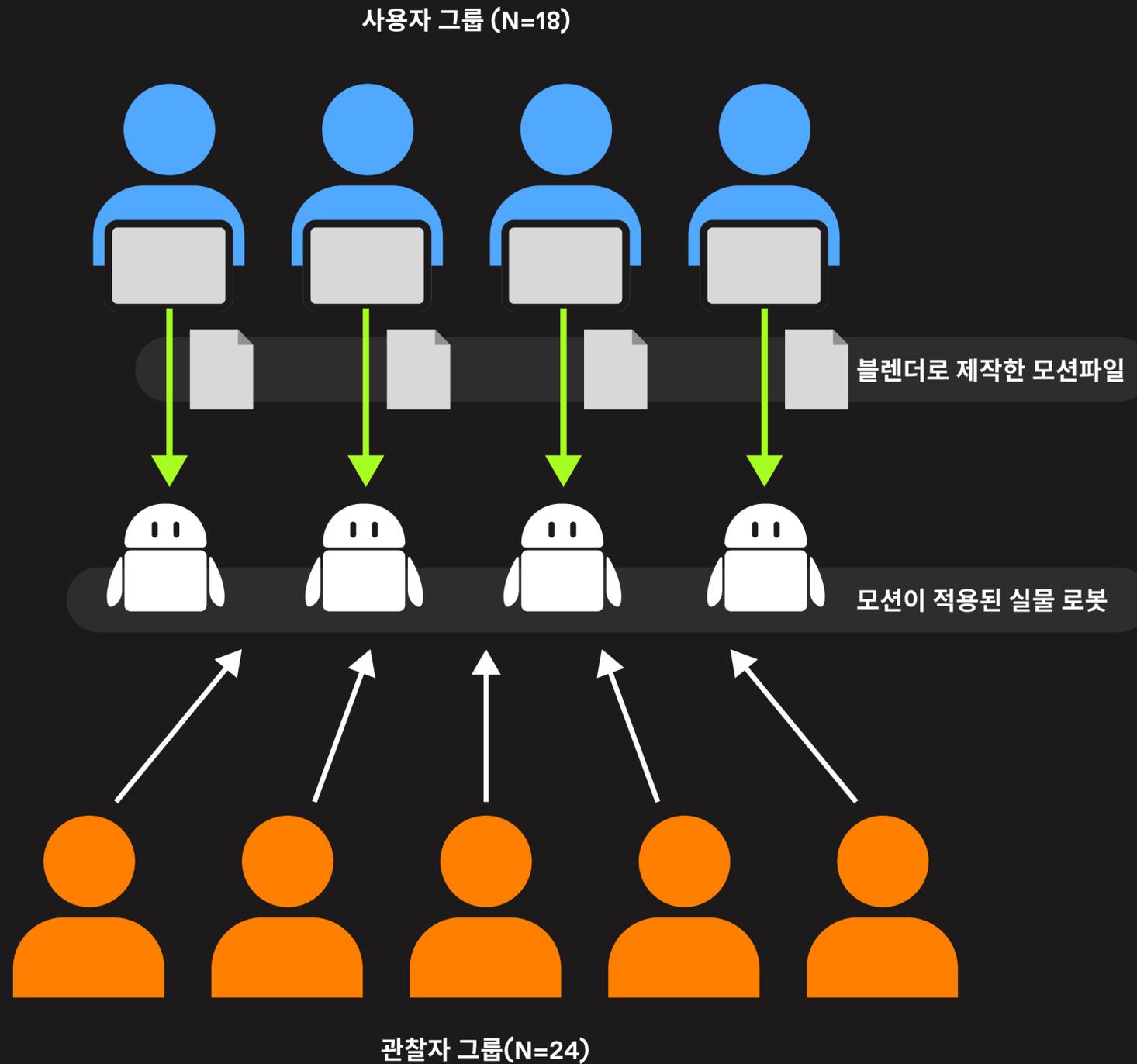


## 4.1 리서치 설계

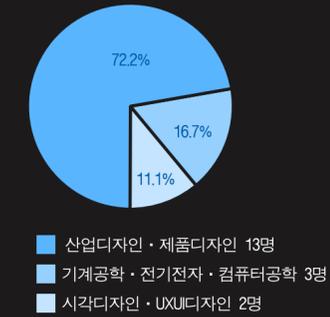
### Main Focus



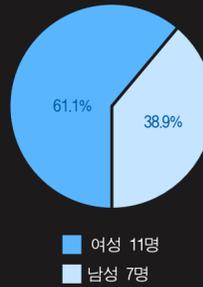
## 4.1 리서치 설계



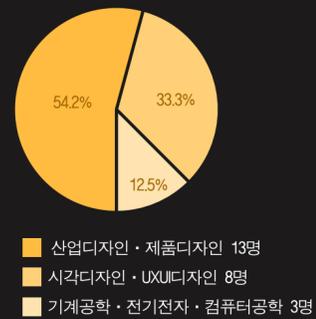
사용자 그룹 전공 분포



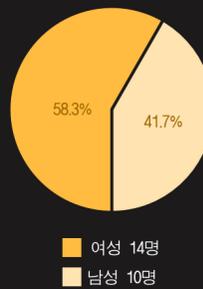
사용자 그룹 성별 분포



관찰자 그룹 전공 분포



관찰자 그룹 성별 분포



### 4.1 리서치 설계

#### 사용성 평가 8개 항목

코딩 지식 없이 사용 가능 여부

모션 표현 자유도

다수 모터 동작 구현 용이성

미리보기 기능 유용성

키프레임 조작의 직관성

모션 수정 편의성

제작 및 검증 신속성

모터-애니메이션 일치성

#### GQS 5개 영역 24개 항목

##### Anthropomorphism (의인성)

Fake – Natural  
Machinelike – Humanlike  
Unconscious – Conscious  
Artificial – Lifelike  
Moving rigidly – Moving elegantly

##### Animacy (생동감)

Dead – Alive  
Stagnant – Lively  
Mechanical – Organic  
Artificial – Lifelike  
Inert – Interactive  
Apathetic – Responsive

##### Likeability (호감도)

Dislike – Like  
Unfriendly – Friendly  
Unkind – Kind  
Unpleasant – Pleasant  
Awful – Nice

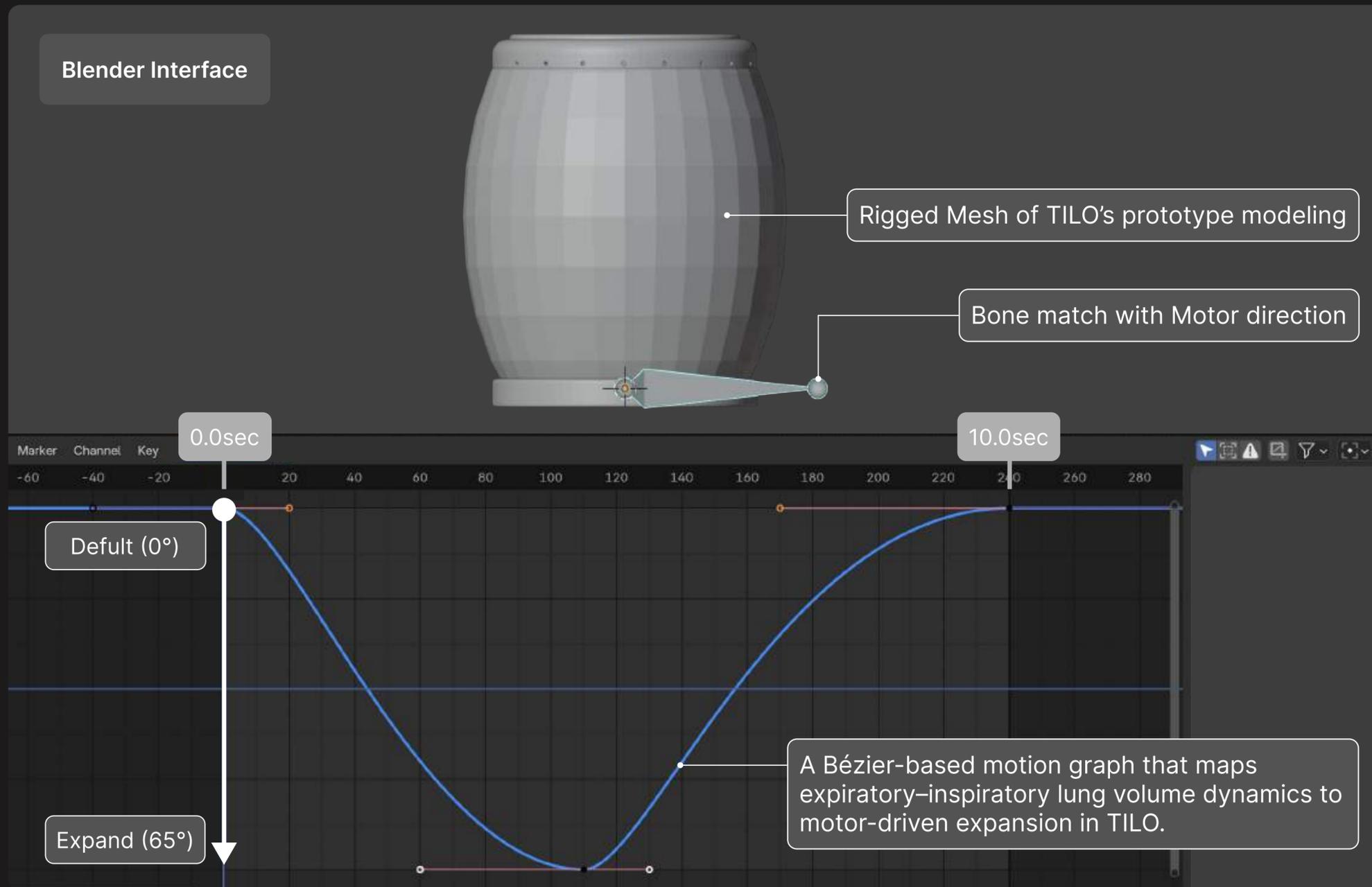
##### Perceived Intelligence (감지된 지능)

Incompetent – Competent  
Ignorant – Knowledgeable  
Irresponsible – Responsible  
Unintelligent – Intelligent  
Foolish – Sensible

##### Perceived Safety (감지된 안전성)

Anxious – Relaxed  
Agitated – Calm  
Quiescent – Surprised

### 프레임워크를 로봇 프로토타이핑에 활용한 예



## 4.2 분석 결과

**모션을 만든 경험(모션 표현 자유도, 모터-애니메이션 일치성, 모션 수정 편의성)이 로봇의 인상(Animacy, Likability, Perceived Safety 등)에 투영되었다**

[표2] 사용성 항목과 GQS 항목 간 Spearman 상관분석 결과

사용성	GQS	$\rho$	p (<0.05)
모션 표현 자유도	Likeability (Dislike – Like)	0.539	0.046
	Likeability (Unfriendly – Friendly)	0.522	0.026
	Likeability (Awful – Nice)	0.539	0.036
	Perceived Safety (Agitated – Calm)	0.496	0.021
모터-애니메이션 일치성	Animacy (Dead – Alive)	0.497	0.036
	Likeability (Unpleasant – Pleasant)	0.511	0.030
	Likeability (Dislike – Like)	0.534	0.023
	Likeability (Unfriendly – Friendly)	0.493	0.038
모션 수정 편의성	Perceived Safety (Anxious – Relaxed)	0.499	0.035

**Key Insight**

**모션 표현 자유도 → 친근감·안정감 상승**

**애니메이션-모터 일치성 → 생동감 증가**

## 4.2 정성 인터뷰

**표현력·생동감 설계에는 강점을 보이나, 비전공자 접근성과 물리적 정확성은 보완 필요**

### 장점

눈으로 예상되는 결과물이 보여서 좋았어요

모션을 잡기위한 방식이 기존에 하던 인터페이스라 익숙했어요

수정이 필요하면 바로 작업하고 반영할 수 있어 편했습니다

### 한계

Blender에 적응하는게 어려웠어요

VS Code에 코드 붙여넣기를 처음 해봐서 어려웠어요

실물 구동 시 토크·하중 문제로 수정이 필요한 경우가 있었어요

# 5. 결론

## Key Contribution

**본 연구는 애니메이션 원리에 기반한 로봇 움직임을  
디자이너가 직접 수행하고 즉시 검증할 수 있는 새로운 프레임워크를 제안**

## Research Result

**일부 감성 인식 항목과의 상관관계를 통해,  
표현적 움직임 설계 경험이 로봇의 생동감과 호감도 같은  
사회적 인상에 실제로 영향을 미칠 수 있음을 확인**

## Future Work

**향후 Blender Add-on 형태의 인터페이스 제공을 통해 비전문가 접근성을 높이고  
물리 기반 검증(토크/하중 반영)을 강화하여 실사용 가능성을 확대하는 방향을 검토**

**E.O.D**

0523ljy@gmail.com