



릭 로직

메타휴먼 페이스릭의
런타임 평가

백 서





목차

소개	3
용어	3
배경	5
리얼타임 페이스 리크 평가에서의 난관	6
솔루션	7
조건	7
조인트 디포메이션 감소	7
레벨 오브 디테일(LOD)	9
메타휴먼 DNA 파일 포맷	11
교정 표정	12
런타임 전략	15
조인트	15
블렌드 셰이프	16
애니메이티드 맵	16
퍼포먼스	16
결론	17

소개

릭 로직은 3레터럴에서 개발한 런타임 페이스 릿 평가 솔버 시스템입니다. 릿 로직은 고급 캐릭터 커스터마이징 및 플레이어 디지털화 시스템을 위한 인프라 기반으로 높은 퀄리티의 릿을 제공하며, 다용도 캐릭터화 툴이기도 합니다.



그림 1: 릿 로직 유저 인터페이스

릭 로직은 인간의 얼굴 근육 체계를 정의하는 보편적 룰 세트와, 3D 오브젝트의 릿과 지오메트리의 완전한 설명을 저장할 수 있도록 설계된 3레터럴의 전용 파일 포맷 '메타휴먼 DNA'를 바탕으로 합니다.

릭 로직을 개발한 **3레터럴**은 2012년 설립된 이래로 페이스 릿 및 애니메이션을 중점적으로 연구하고 개발해 온 기업으로, *세누아즈 사가: 헬블레이드 2(Senua's Saga: Hellblade II)*와 *마블스 스파이더맨(Marvel's Spider-Man)* 등 수많은 영화와 게임 프로젝트를 진행했습니다. 3레터럴은 2019년 에픽게임즈에 합류했으며 릿 로직은 현재 에픽게임즈의 **메타휴먼 크리에이터** 툴에 쓰이고 있습니다.

이 백서에서는 메타휴먼 DNA 파일 포맷을 결정하게 된 이유와 릿 로직 시스템을 구축하는 과정에서 결정한 기타 사항, 이 포맷이 페이스 포즈 및 애니메이션의 리얼타임 렌더링에 적합한 이유에 대해 설명합니다.

용어

릭 로직이 완전 리얼타임 솔루션으로 발전한 방식에 대해 설명하려면 우선 몇 가지 용어를 정의해야 합니다.

표정(Expression) - 미소, 찡그림 또는 음소와 같이 구체적인 얼굴의 표정을 의미합니다. 페이스 리깅 및 애니메이션에서의 표정은 조인트 트랜스폼(이동, 회전, 스케일링 등)과 블렌드 셰이프(일명 모프 타겟) 그리고 애니메이션드 맵에 의해 정의됩니다.



그림 2: 메타휴먼 캐릭터의 얼굴 표정

조인트(Joint) - 페이스 리크은 얼굴의 실제 뼈, 관절, 근육을 어느 정도 모방하는 가상 본과 조인트로 이루어져 있습니다. 실제 근육은 정확하게 모방하기가 어렵기 때문에 가상 조인트가 이를 대체하여 얼굴 모델의 벡스 애니메이션을 구동합니다. 각 조인트에는 애니메이션을 적용하는 벡스 세트가 할당됩니다.

애니메이션 커브(Animation Curve) - 시간 경과에 따른 벡스의 XYZ 이동 표현입니다.

컨트롤(Control) - 표정에 관여하는 조인트, 블렌드 셰이프, 애니메이션드 맵을 연산하는 데 사용되는 단일 기본 표정 웨이트를 갖춘 메커니즘입니다 이상적으로 **FACS 액션 유닛**당 컨트롤이 하나씩 존재합니다.

릭 컨트롤 인터페이스(Rig control interface) - 컨트롤을 조합하거나 사용하기 쉽게 만드는 모든 인터페이스를 말합니다. 슬라이더를 갖춘 그래픽 유저 인터페이스(GUI)에서부터 컨트롤을 수집하고 사용하여 다른 액션을 유발하는 내부 인터페이스까지, 모든 유저 인터페이스가 여기에 포함될 수 있습니다.

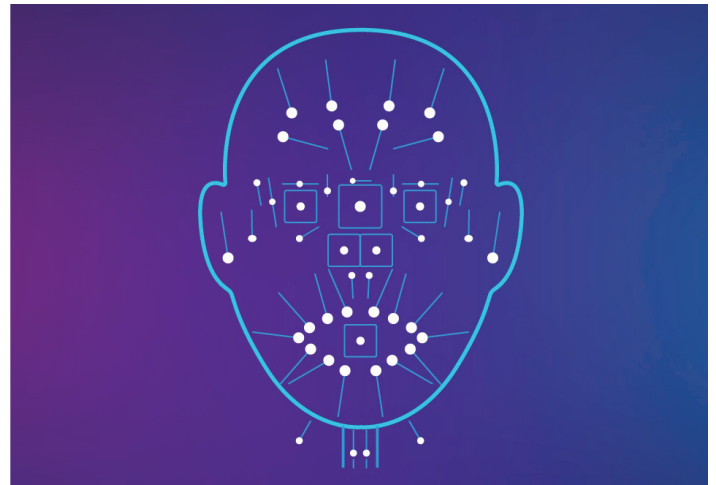


그림 3: 슬라이더를 갖춘 GUI, 릭 제어 인터페이스

복잡도(Complexity) - 페이스 리크이 갖춘 릭 컨트롤의 수를 가리키는 총칭입니다. 컨트롤이 200개 있는 릭은 50개 있는 릭보다 복잡도가 더 높습니다. 컨트롤이 많을수록 교정 표정에 필요한 잠재 표정의 조합도 많아지면서 연산량이 증가하고 보다 많은 메모리를 사용하게 됩니다.

교정 표정(Corrective expression) - 복잡한 표정을 다듬는 내부적 조정입니다. 교정 표정은 이 백서의 뒷부분에서 더 자세히 다루겠습니다.

배경

릭 로직은 중대한 반복작업을 여러 번 거치며 발전했습니다. 최신 세대 버전은 수년간의 연구 개발을 통해 축적된 종합적인 작업과 개선의 결과입니다.

성공적인 페이스리 릭에서 입력 값은 각각 단일 페이스리 **FACS** 표정을 나타내는 컨트롤입니다. 이런 입력마다 연산이 수행되어 다음과 같은 출력 값을 도출합니다.

- 조인트 트랜스포메이션
- 블렌드 셰이프 채널 웨이트
- 애니메이션드 맵에 대한 셰이더 배수

위 세 가지 출력 값은 현재 쓰이고 있는 가장 성공적인 페이스리 릭 시스템의 핵심입니다. 저희의 연구는 전반적으로 이 세 값을 더 정확하고 효율적으로 연산하여 출력하는 데 중점을 두고 진행되었습니다.

3레터럴이 리얼타임 릭 작업을 처음 시작했을 때는 게임, 영화 및 TV를 비롯한 다양한 분야의 3D 아티스트들이 애용하는 Autodesk의 디지털 콘텐츠 제작 패키지, Maya에서 제공되는 툴킷으로만 릭이 빌드되었습니다. Maya 툴킷에는 교정 표정과 셰이더 네트워크로 보완되는 드리븐 키(Driven Key, DK)가 포함됩니다. 디포메이션은 얼굴을 성공적으로 애니메이션하기 위해 필요한 세 가지 출력 값과 함께 제공되는 추상화만 사용하여 연산되었습니다.

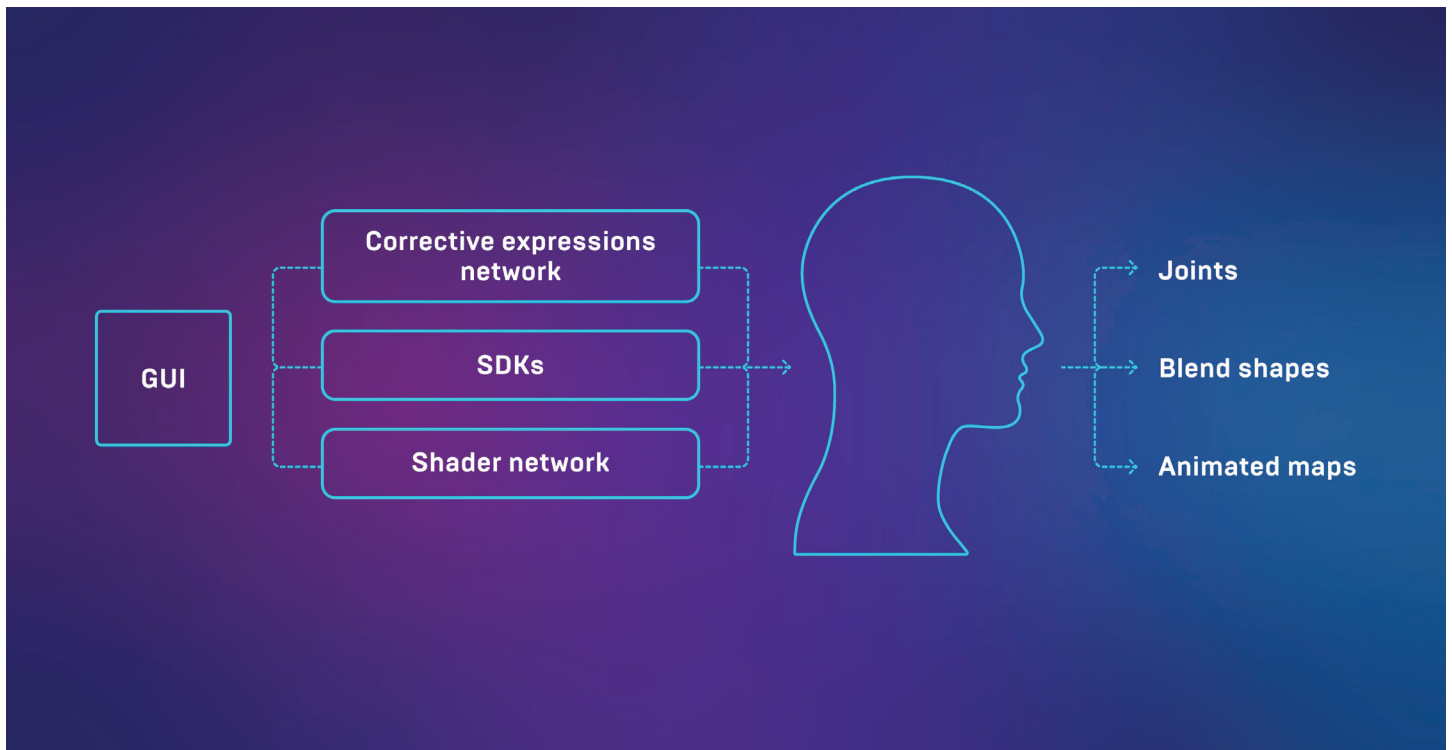


그림 4: 기존 3레터럴 리얼타임 페이스리 릭 평가 프로세스에서의 데이터 플로우

이런 솔루션은 꽤 잘 작동했고 높은 수준의 퀄리티를 제공하기는 했지만, 릭 평가는 굉장히 느렸으며 다른 기술과의 통합도 복잡했습니다.

리얼타임 페이스 리깅 평가에서의 난관

여기서 마주쳤던 난관은 30FPS 이상의 실시간 연산 속도를 유지하면서 동일하거나 비슷한 수준의 퀄리티를 유지하는 페이스 리깅 시스템을 개발하는 것이었습니다. 또한 이 시스템을 다른 기술과도 통합 가능하도록 만들고자 하였습니다.

Maya에서 사용하는 것과 동일한 유형의 입력 값을 바탕으로 디포메이션 표현 출력 값을 연산하되 더 빠르게 실행하는 리깅 로직을 제작하는 것이 핵심이었습니다. 여기에 얼굴 셰이프와 관계없이 여러 캐릭터에서 재사용 가능한 리깅을 만들어, 리깅 솔루션을 어디로든 익스포트하고 다양한 기술에서의 사용성을 극대화한다는 목표를 더했습니다.

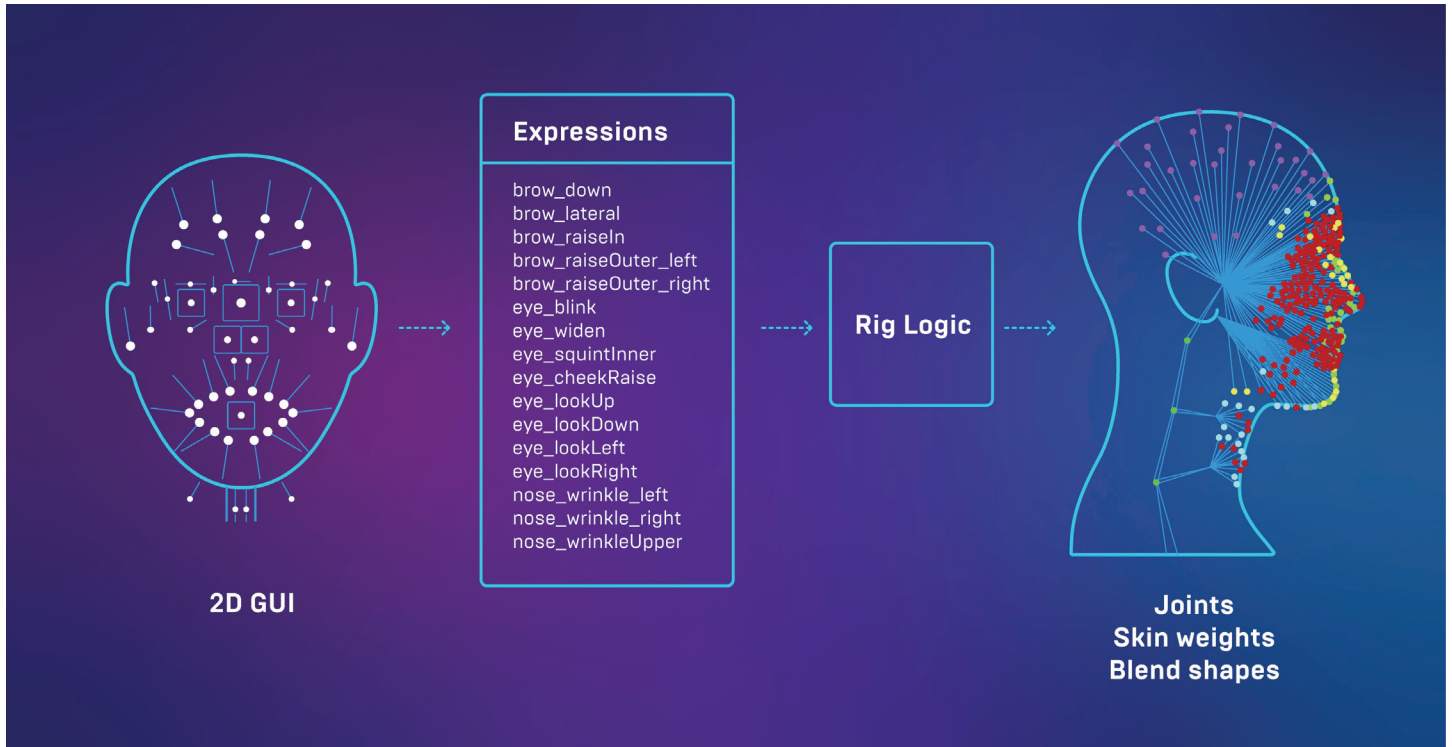


그림 5: 리깅 컨트롤부터 표정, 리깅 로직 연산, 마지막으로 페이스 리깅 조인트로 이어지며 원하는 표정을 짓는 데이터 플로우

고퀄리티 페이스 애니메이션을 제작하는 최첨단 페이스 리깅에는 조인트 800개, 표정 컨트롤 200개 이상이 포함되고 1,000개 이상의 교정 표정으로 보완됩니다. 즉, 리얼타임 출력을 수용할 수 있는 런타임 솔루션을 개발해야 한다는 뜻이었습니다.

메타휴먼 크리에이터 페이스 리깅에 사용한 가장 중요한 요소는 다음과 같습니다.

- 조인트 및 스킨 웨이트 디포머
- 블렌드 셰이프 디포머
- 베이스 및 애니메이션 텍스처 구성
- 리깅 로직 플러그인 및 메타휴먼 DNA 파일
- 유저 인터페이스
- 내장 LOD

솔루션

조건

페이셜 리깅 및 애니메이션에 사용 가능한 런타임 솔루션을 완성하고자, 릭 로직은 릭과 그 작동에 대해 다음과 같은 조건을 만족해야 합니다.

- **런타임 평가.** 이상적인 페이셜 릭의 런타임 평가는 애니메이션의 리얼타임 재생이 30FPS 이상의 적절한 속도로 재생된다는 것을 의미합니다.
- **입출력 파라미터 감소.** 파라미터 수를 줄이면 시스템 실행 속도도 자연스럽게 빨라집니다.
- **무손실 애니메이션 압축.** 압축된 데이터는 공간을 덜 차지하기 때문에 보다 빨리 처리할 수 있습니다. 일반적으로 데이터 압축은 최소한의 CPU 비용으로 데이터 용량을 10~15배 정도 줄여 주지만, 이런 감소는 무손실 압축일 때만 유효합니다.
- **릭 재사용성.** 서로 다른 캐릭터 간의 손쉬운 애니메이션 데이터 공유가 가능하도록, 시스템은 페이셜 셰이프, 비율, 데모그래피 등에 대한 릭 종속성이 없도록 설계되어야 합니다.
- **유연한 LOD 사용.** 동일한 애니메이션이 모든 LOD 사양을 구동할 수 있어야 합니다.
- **더 효율적인 파이프라인.** 런타임 중에 평가가 이루어지는 릭을 생성하면 자연스럽게 특정 파이프라인의 효율성 개선으로 이어집니다. 예를 들어, 데이터 용량이 훨씬 작아진다면 애니메이션의 익스포트도 더욱 간편해지며, 시스템이 제대로 설계되었다면 릭을 업데이트해도 애니메이션을 전부 리익스포트할 필요가 없어집니다.
- **모든 페이셜 애니메이션 소프트웨어와의 손쉬운 연결.** 이 솔루션은 릭의 근육 기반 조직을 바탕으로 하기 때문에, **스피치 그래픽스(SpeechGraphics)**와 같은 오디오 기반 페이셜 애니메이션과 직관적인 통합이 가능합니다.
- **비선형 애니메이션 조합 지원.** 시스템은 런타임 중에 다이나믹한 반응을 지원해야 합니다. 예컨대, 캐릭터의 눈에 비치는 손전등 불빛 같은 게임 내 이벤트에 캐릭터가 인터랙티브한 반응을 보이면서 연기를 펼쳐야 할 수도 있습니다. 솔루션은 애니메이션의 감정 레이어링도 지원해야 합니다.

조인트 디포메이션 감소

Maya로부터 릭을 분리하고자 릭 로직은 페이셜 릭을 표현하는 필수 데이터를 캡처하고 이를 실시간으로 평가하는 런타임 컴포넌트를 제공하려는 목적으로 개발되었습니다. 릭 로직의 첫 세대는 Maya로부터 데이터를 추출하여 다음 선형 함수로 드리븐 키를 표현하는 방식을 사용했습니다.

$$y = k * x$$

y 값은 x가 증가함에 따라 선형적으로 증가하며, y 증가율의 각도 [기울기]는 k에 의해 결정됩니다.

- x는 입력 컨트롤 값으로, 범위는 [0.0,1.0]으로 제한됩니다.
- y는 출력 디포메이션 값입니다(예: joint.rotation.x 파라미터 값).
- k는 함수의 행동 방식을 정의하는 릭 파라미터입니다.

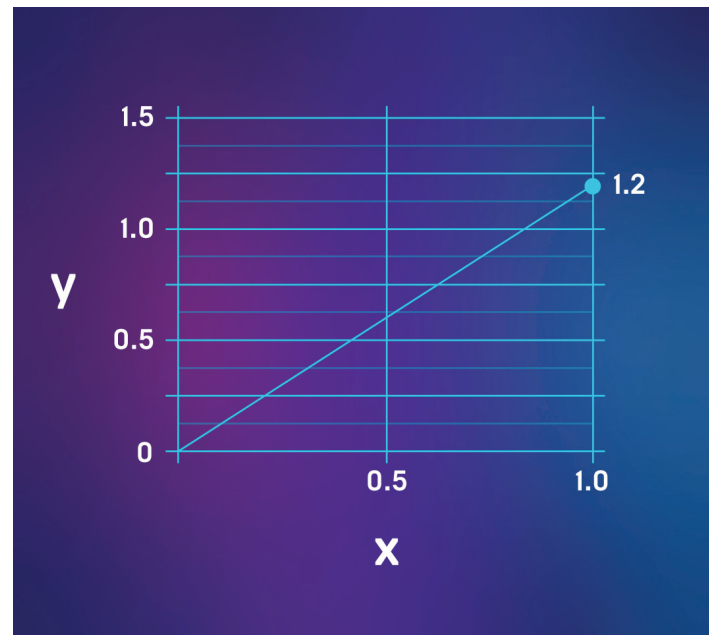


그림 6: x(입력), y(출력), k(릭 파라미터) 값을 결정하는 기울기 함수 $y=k*x$ 표현식



그림 7: 메타휴먼 캐릭터의 얼굴 표정

이런 방식으로 x 및 y 값을 처리하면 기울기 함수의 파라미터를 매트릭스로 인코딩할 수 있습니다.

매트릭스 타입으로는 저장 공간 조건을 최소화하고 적절한 연산 퍼포먼스를 제공할 수 있도록 희소한 **압축 열 스토리지 (Compressed row storage, CRS) 매트릭스**를 선택했습니다. 이 버전의 릭 로직에서는 지오메트릭 프리미티브가 도입되지 않아서, 시스템은 여전히 Maya의 지오메트릭 프리미티브(메시)에 의존하고 있었습니다.

이후 이 매트릭스를 분석한 결과, 해당 파라미터의 고밀도 집합은 대부분 희소 매트릭스에서 생성되고 있다는 점이 밝혀졌습니다. 또한 이와 동일한 고밀도 서브 매트릭스가 얼굴의 특정 영역을 타기팅하는 디포메이션의 파라미터를 포함하는 것으로 관찰되면서 조인트 그룹이라는 개념이 도입되었습니다. 조인트 그룹은 이러한 고밀도 서브 매트릭스를 나타내면서, 조인트의 논리적 분리를 통해 얼굴의 특정 영역을 타기팅하는 조인트 그룹을 형성합니다.

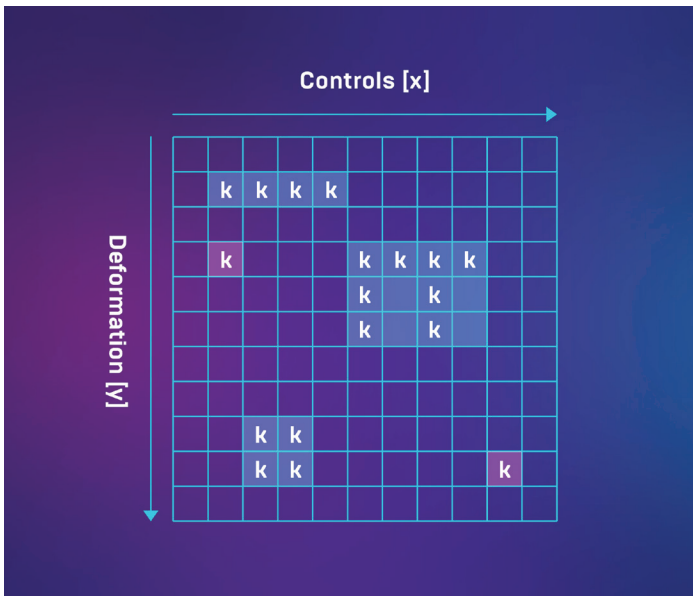


그림 8: 조인트 트랜스포메이션 델타(k)의 샘플 매트릭스 표현, 선형 함수 $y = k \cdot x$ 로 연산됨

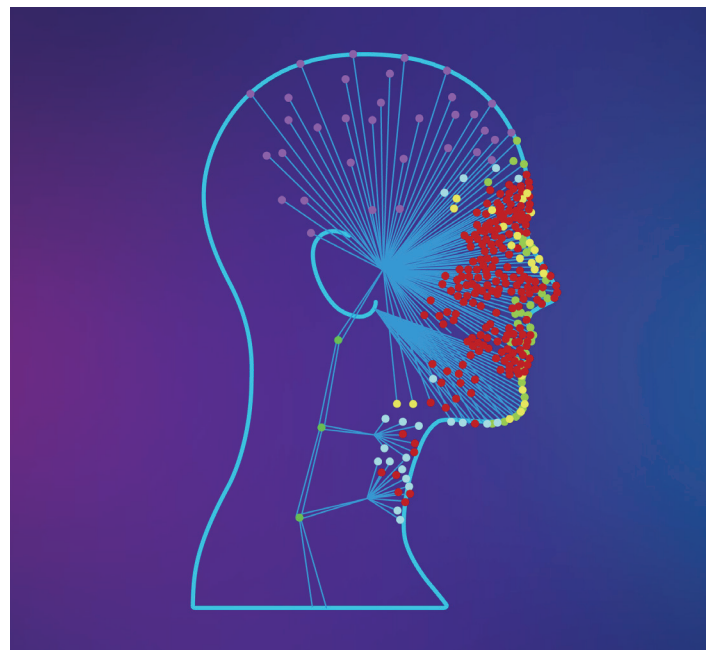


그림 9: 조인트 그룹

그림 8에서 볼 수 있듯이 고밀도 서브 매트릭스의 일부가 아닌 파라미터라는 이상값도 있었습니다. (다이어그램에서 분홍색으로 표시된 것이 이상값입니다.) 이러한 이상값은 해당 조인트에 영향력을 전혀 끼치지 않아야 하는 얼굴 영역에서도 디포메이션을 나타내고 있었습니다. 이런 관찰을 통해 스마트 정리 프로세스가 도입되어 해당 이상값을 제거하고 오직 고밀도 영역만 갖춘 매트릭스만 남겼습니다.

이 정리 프로세스는 런타임 도중 저장 및 평가해야 하는 데이터를 효과적으로 줄이고 잘못된 디포메이션을 제거했습니다. 그 결과 고밀도 영역에서 조인트를 평가하는 초고속 알고리즘을 구현할 기회를 얻었습니다.

이후의 반복작업에서는 조인트, 블렌드 셰이프, 애니메이션드 맵 같은 고유한 데이터 타입을 제각기 조건부 선형 함수를 나타내는 별도의 매트릭스로 분리했습니다.

레벨 오브 디테일(LOD)

얼굴 표정은 보통 두 가지 이상의 기본 표정으로 구성됩니다. 복잡한 표정은 예를 들어 [1] 치켜세운 눈썹, [2] 크게 뜬 눈, [3] 살짝 벌어진 입술 같은 세 가지 기본 표정의 조합으로 구성됩니다.

이러한 복잡한 표정에 최대한의 효율성을 부여하고자, 릭 로직은 LOD 지원을 염두에 두고 설계되었습니다. 모든 표정은 특정 조인트의 트랜스포메이션으로 정의되지만, 모든 조인트가 특정 표정 형성에 기여하는 것은 아닙니다. LOD 0, 즉 가장 디테일한 수준에서는 해당 표정에 참조된 모든 조인트 트랜스폼이 표정을 형성하는 데 사용됩니다. 하지만 LOD 1, LOD 2 등 낮은 LOD에서는 LOD 0에서 참조하는 조인트의 하위 집합만을 사용합니다.

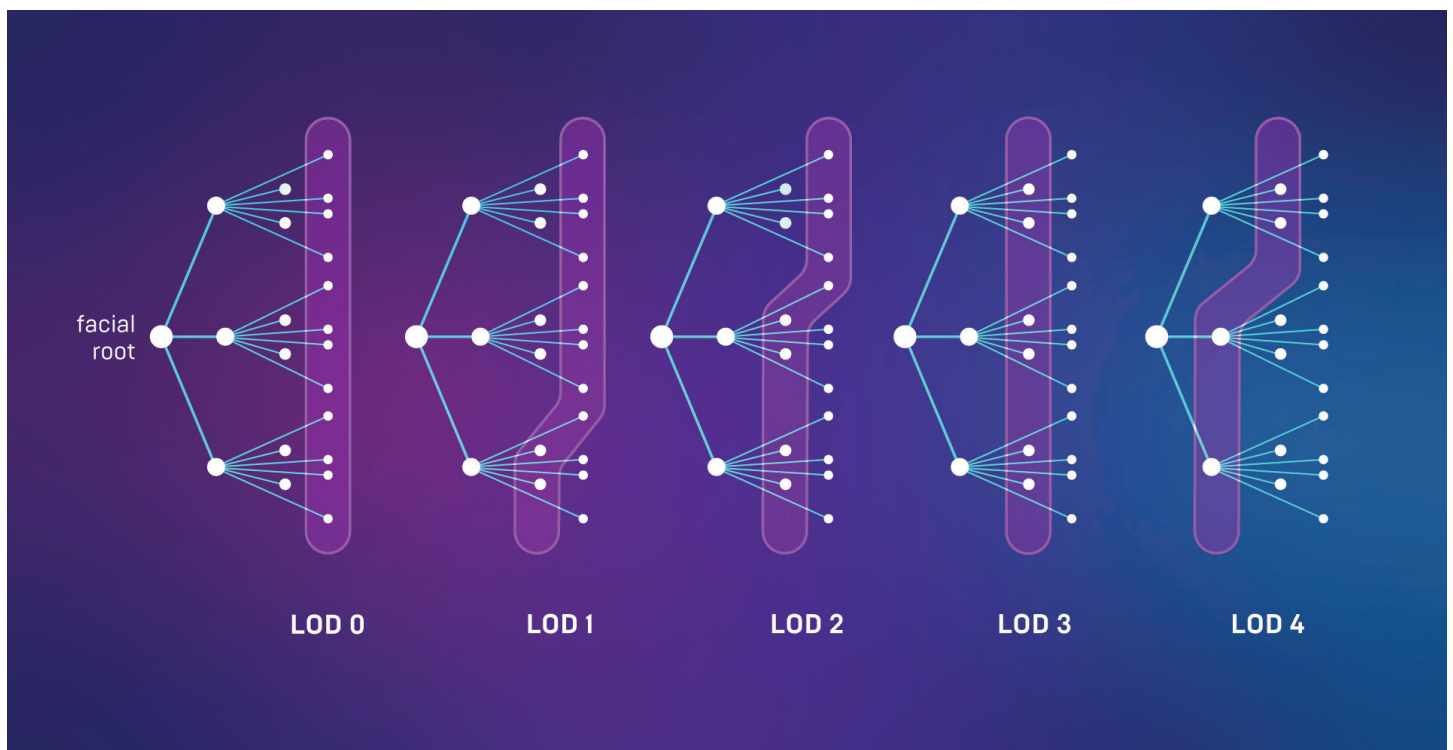


그림 10: LOD 0~LOD 4에서 관련 조인트 수 변화를 보여주는 다이어그램. LOD 0에서 토폴로지 밀도가 가장 높고 조인트 수가 많으며, LOD 4에서 토폴로지 밀도가 가장 낮고 조인트 수가 최소화됨.

하위 LOD에서 참조되는 조인트를 LOD 0에서 참조된 동일 조인트로 제한할 경우, 데이터 낭비를 최소화하면서 아무 비용 없이 LOD 간에 전환할 수 있는 메커니즘이 제공됩니다. 낮은 LOD에서 높은 LOD로 옮겨도 지정된 조인트를 이미 로드된 데이터에 추가할 뿐이며, 높은 LOD에서 낮은 LOD로 옮겨도 참조되지 않는 조인트를 제거할 뿐입니다.

시스템에는 8개의 LOD가 내장되어 있지만, 릿 로직은 필요에 따라 더 추가하도록 커스터마이징이 가능합니다. 얼굴의 다양한 부분에 필요한 조인트 수 제한과 디포메이션 퀄리티에 따라, 이 시스템은 조인트와 스킨 웨이트를 다양한 토폴로지와 간단하게 조합하여 다양한 LOD를 형성할 수 있습니다.

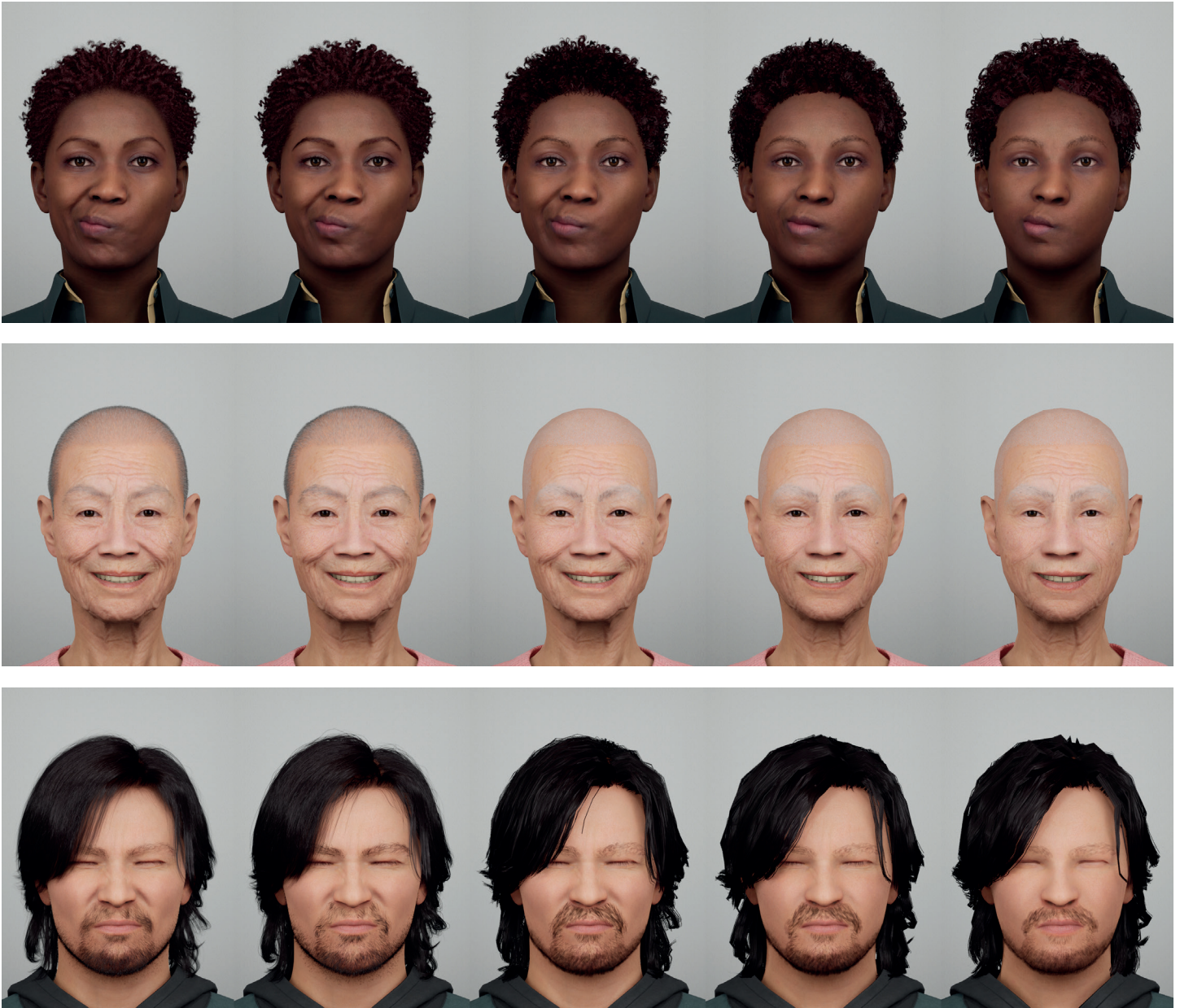


그림 11: 상위 LOD에서 하위 LOD로 표현되는 표정 예시. 비용 없이 LOD가 전환됨.

메모리 사용량을 더 줄이고자 한다면, 런타임 중에 필요하다고 예상되는 LOD를 지정하거나, 최대/최소 LOD를 지정할 수도 있습니다. 선택된 LOD가 참조하지 않는 데이터는 로드되지 않습니다.

메타휴먼 DNA 파일 포맷

3레터럴 메타휴먼 DNA 파일 포맷은 3D 오브젝트 릭 및 지오메트리의 전체 설명을 저장하도록 디자인되었습니다. 따라서 메타휴먼 DNA 파일만으로도 오브젝트의 전체 메시를 재구성하고 완전히 리깅하여 애니메이션 가능한 상태로 만들 수 있습니다. 실제로 메타휴먼 DNA 파일은 인간 캐릭터의 바다, 소품 또는 여타 오브젝트가 아닌, 오로지 얼굴을 저장하는 데만 쓰입니다.

메타휴먼 DNA 파일의 콘텐츠는 전용 바이너리 포맷을 사용하여 인코딩되지만 포맷 사양과 읽는 코드는 오픈소스입니다.

레이어

메타휴먼 DNA 파일에 포함된 데이터는 몇 가지 논리 레이어로 구분됩니다. 레이어는 메타휴먼 DNA 파일의 각 하위 레이어가 그 아래 레이어의 저장된 데이터에 의존하는 느슨한 계층구조로 연결되어 있습니다.

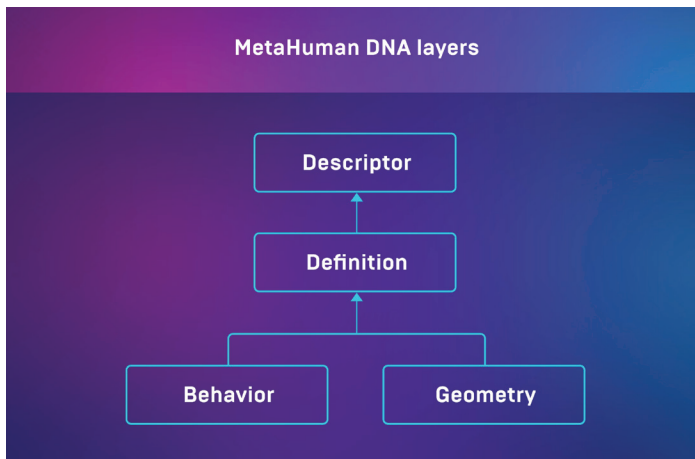


그림 12: 메타휴먼 DNA 파일의 논리 레이어

메타휴먼 DNA 데이터는 지정된 레이어까지만 선택적으로 로드할 수 있습니다. 그림 12에서 볼 수 있듯, 비헤이비어와 지오메트리 레이어는 독립적입니다. 이런 독립성은 메타휴먼 DNA 파일이 지오메트리 데이터에 액세스할 필요 없이 릭만 구동하면 되는 경우(비헤이비어 레이어를 활용한 런타임 평가)에 매우 중요합니다.

이런 독립성 덕분에 솔루션 이식이 가능해져서 Maya 지오메트리나 여타 특정 지오메트리에 적용될 필요가 없으며, 페이스 지오메트리가 있는 다른 애플리케이션과 함께 사용할 수 있습니다. 이 파일 포맷은 리얼타임 페이스 애니메이션을 효율적이면서도 완전한 포맷으로 제작할 수 있도록 세심하게 디자인되었습니다.

디스크립터 레이어

디스크립터 레이어는 다음과 같은 릭 관련 기본 메타데이터를 포함합니다.

- 캐릭터 이름
- 연령
- 페이스 아키타입
- 키/값 쌍 형태의 임의 스트링 메타데이터
- 호환성 파라미터(필요 시, 메타휴먼 DNA 파일 조합처럼 보다 높은 수준의 시스템과 관련이 있음)

정의 레이어

정의 레이어는 다음과 같은 릭의 정적 데이터를 포함합니다.

- 컨트롤, 조인트, 블렌드 셰이프, 애니메이션드 맵, 메시의 이름
- 개별 LOD로의 컨트롤, 조인트, 블렌드 셰이프, 애니메이션드 맵, 메시 매핑
- 조인트 계층구조
- 바인딩 포즈(T포즈 등)의 조인트 트랜스포메이션

이 레이어는 선택한 LOD를 바탕으로 한 하위 레이어에서의 필터링 수행에 필수적인 정보를 포함합니다.

비헤이비어 레이어

비헤이비어 레이어는 다음 용도로 쓰이는 릭의 동적 데이터를 포함합니다.

- 원시 컨트롤 값에 GUI 컨트롤 매핑
- 교정 표정 연산
- 조인트 트랜스포메이션 연산
- 블렌드 셰이프 채널 웨이트 연산
- 애니메이션드 맵 웨이트 연산

지오메트리 레이어

지오메트리 레이어는 캐릭터 메시 재구성에 필요한 데이터와 스킨 웨이트 및 블렌드 셰이프 타깃 델타 등을 전부 포함합니다. 메시 정보 자체는 OBJ 파일 포맷과 유사한 포맷으로 구축됩니다.



그림 13: 메타휴먼 캐릭터의 얼굴 표정

교정 표정

복잡한 표정의 경우, 일부 조인트가 하나 이상의 기본 표정에 영향을 받는 오버랩이 발생할 수 있습니다. 이런 상황에서 단순히 지정된 디포메이션을 전부 활성화했다가는 디포메이션 '곰셈'이 발생할 수 있으며, 이 경우 보통 시각적 결과물에 좋지 않습니다.

교정 표정은 복잡한 표정에서의 조인트 오버랩을 교정하는 솔루션을 나타냅니다. 복잡한 표정마다 이러한 문제 영역에 발생하는 손상을 근본적으로 복구하는 교정 표정을 정의해 주어야 합니다. 이런 복잡한 표정은 기본 표정의 각 조합(또는 최소한 중요하다고 간주되는 조합)에 대해 정의되어야 합니다.

따라서 지원되는 기본 표정의 수가 증가하면 복잡한 표정의 수도 기하급수적으로 증가합니다. 가능한 모든 조합에 교정 솔루션을 활성화하는 것은 메모리 및 연산 비용 때문에 불가능하지만, 가장 중요한 조합은 처리할 수 있습니다.

그림 14는 화를 내며 고함을 지르는 페이스얼 포즈의 페이스얼 디포메이션 사례입니다. 이런 포즈를 생성하려면 몇 가지 기본 표정을 조합해야 합니다.

- jawOpenExtreme
- mouthCornerPull
- mouthStretch
- neckStretch
- noseWrinkler
- squintInner
- browsLower

이 경우 이런 표정들에 다수의 조인트 오버랩이 생길 가능성이 높습니다. 예를 들어, 입과 코 표정을 수정하지 않고 조합한다면 얼굴에서 입과 코 주변 영역이 극단적으로 늘어질 것입니다.



그림 14: 특정 기본 표정 조합에 교정 표정을 적용하기 이전(왼쪽)과 이후(오른쪽)의 얼굴 포즈

왼쪽 얼굴은 위에 나열된 표정들을 그대로 조합한 것으로, 꽤 부자연스러운 결과물이 되었습니다. 이 문제를 해결하려면 특정한 표정 조합이 활성화될 때마다 이에 맞는 교정 표정으로 포즈를 조정해야 합니다.

오른쪽 얼굴은 표정 조합에 교정 표정이 적용된 페이스샷 포즈로, 훨씬 더 자연스럽게 실감나는 포즈를 선보입니다.

교정 표정이 적용된 추가 샘플은 그림 15에서 볼 수 있습니다.

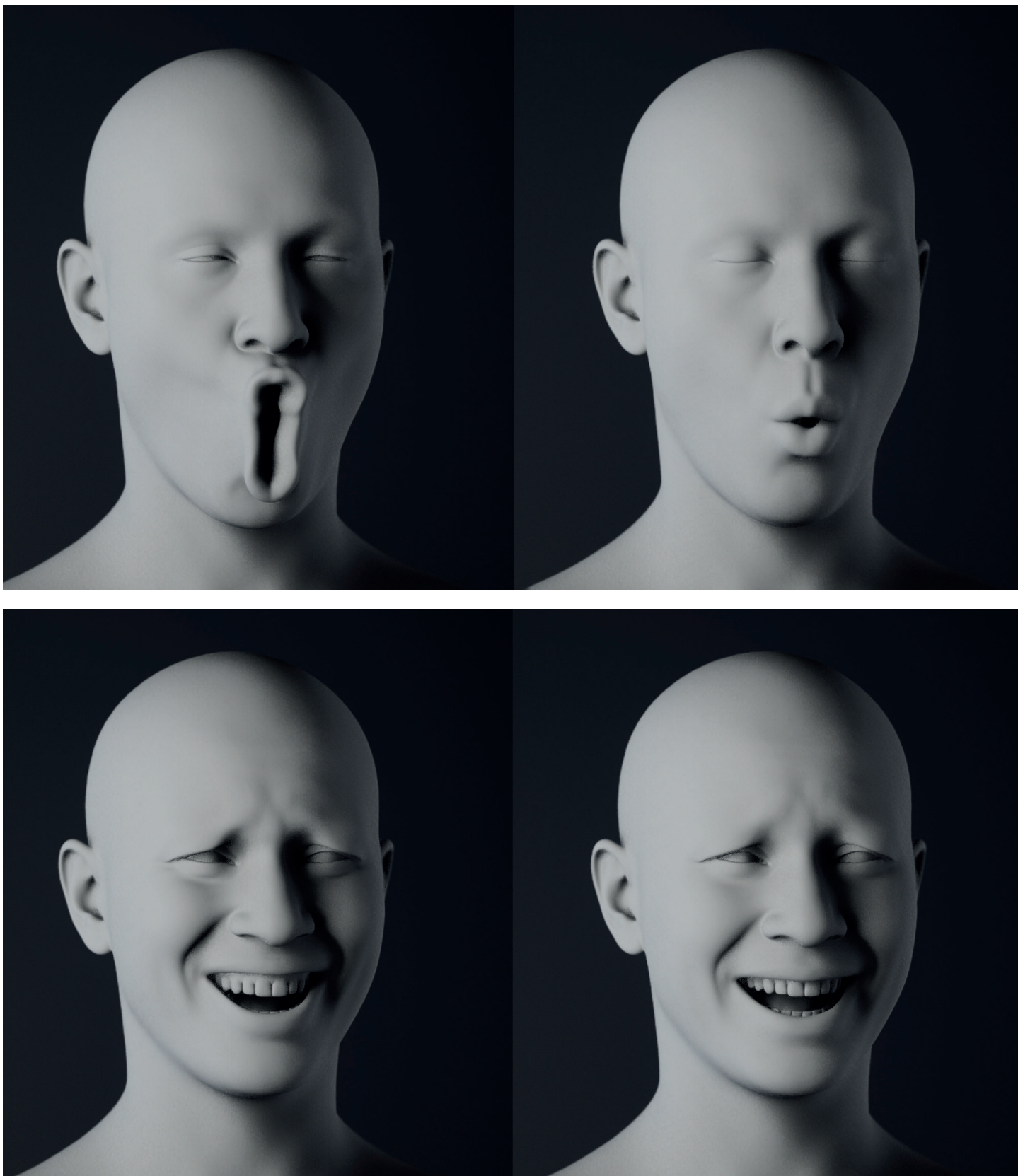


그림 15: 디포메이션이 '곱해진' 모습(왼쪽)과 이를 정정하는 교정 표정이 적용된 얼굴(오른쪽)

런타임 전략

조인트

릭 로직의 조인트는 새로운 정리 전략에 따른 조인트 그룹으로 나뉩니다. 메타휴먼 DNA 파일을 준비하는 과정에서, 비록 손상 자체는 미미하더라도 결과물에 부정적인 영향을 끼칠 수 있는 불필요한 데이터를 제거하여 조인트 트랜스포메이션을 표현하는 데이터를 정리할 수 있습니다. 이를 통해 저장할 데이터 양이 줄어드는 긍정적인 효과가 있으며 런타임에서 수행해야 하는 연산도 줄어듭니다.

전반적인 정리 과정은 다음과 같습니다.

- 얼굴이 다수의 논리 영역(이마, 아래턱, 위턱 등)으로 분리됩니다.
- 조인트가 영향을 끼치는 각 얼굴 영역별 그룹으로 분리됩니다.
- 각각의 표정과 표정의 영향을 받는 조인트 그룹 간에 매핑이 형성됩니다.
- 조인트 그룹이 컨트롤되지 않는 영역에 영향을 미치는 데이터를 포함하지 못하도록 제약할 컨스트RAINT가 정의됩니다.

이는 본질적으로 조인트 그룹이 구성되어 메타휴먼 DNA 파일로 익스포트될 때, 해당 조인트 그룹이 할당된 얼굴 영역에 속한 조인트 데이터만 그룹에 포함될 것이며, 또한 해당 데이터는 처리 중인 조인트 그룹에 영향을 끼칠 수 있는 특정 표정의 트랜스포메이션만을 포함하도록 필터링된다는 의미입니다.

앞서 언급된 조인트의 LOD 메커니즘은 실제로 조인트 그룹별로 LOD마다 개별 정의됩니다.



그림 16: 색상별로 정의된 얼굴 영역

블렌드 셰이프

릭 로직에서 연산된 블렌드 셰이프 채널 웨이트는 블렌드 셰이프 타깃 델타가 활성화되어야 하는 강도를 나타냅니다. 블렌드 셰이프 타깃 델타는 릭 로직 자체에서 생성되지 않으며, 대신 메타휴먼 DNA 파일의 지오메트리 섹션에 저장됩니다. 이 연산 프로세스는 릭 로직이 받는 입력 컨트롤 웨이트의 단순 리매핑입니다.

조인트와 마찬가지로, 블렌드 셰이프 채널 웨이트를 연산할 데이터도 LOD 기준으로 정렬됩니다. 따라서 간단한 슬라이싱 작업만으로도 선택된 LOD의 연산을 제한할 수 있습니다.

애니메이티드 맵

애니메이티드 맵에서 생성된 출력은 배수이며, 이는 주름이나 피하 혈류 이펙트 등을 제어하는 셰이더에 직접 전달됩니다.

애니메이티드 맵은 다른 솔루션과 동일한 법칙에 따라 LOD 기준으로 정렬되며, LOD 교체 시 비용이 들지 않는 혜택 또한 똑같이 적용됩니다.

퍼포먼스

조인트를 조인트 그룹으로 나누는 과정을 통해 희소 매트릭스 대신 고밀도 서브 매트릭스를 조인트 스토리지로 활용할 수 있었으며, 이는 조인트 트랜스포메이션을 연산하는 매우 효율적인 알고리즘을 구현하는 데 유용한 것으로 나타났습니다. 데이터를 캐시 및 **SIMD** 친화적인 소형 블록으로 분할하는 커스텀 매트릭스 레이아웃을 사용하였으며, 스칼라 나머지 루프가 필요하지 않도록 최소한으로 패딩하였습니다. 덕분에 전체 코드 경로가 완전히 벡터화되었습니다(패딩 메모리 오버헤드 2% 미만).

동일한 작업에 단순 CSR 매트릭스를 사용했던 기존 릭 로직과 비교해 볼 때, 신규 버전은 평가 시간이 최대 6배까지 향상되었습니다. 또한 평가되는 조인트의 수도 대폭 증가했습니다. 덕분에 결과 코드에서 거의 이상적인 CPU 파이프라인 활용성을 달성했습니다.



결론

릭 로직은 이전의 리얼타임 페이스 애니메이션 솔루션에 비해 큰 향상을 이루었습니다. 높은 효율성, 이식성 및 기타 특징을 갖춘 릭 로직은 메타휴먼 크리에이터와 그 외 구현에 적용할 수 있는 솔루션이란 점이 증명되었습니다.

앞으로도 지속적인 개발을 통해 릭 로직을 한층 더 향상시켜, 에픽게임즈의 프로젝트 요구사항에 부합하도록 발전시킬 계획입니다. 또한 릭 로직의 향후 버전 개발 및 향상에 기여할 유망 기술도 주의 깊게 관찰할 것입니다.

문서 정보

기술 개발

3레터럴

저자

Andrean Franc

기여

Pavel Kovac

편집

Michele Bousquet

그래픽 및 레이아웃

Jung Kwak

참고: 릭 로직이 통합된 제품으로는 메타휴먼 크리에이터, 메타휴먼 캐릭터 에셋, 언리얼 엔진이 있으며, 그 사용은 해당 제품에 적용되는 라이선스의 보호를 받습니다.

