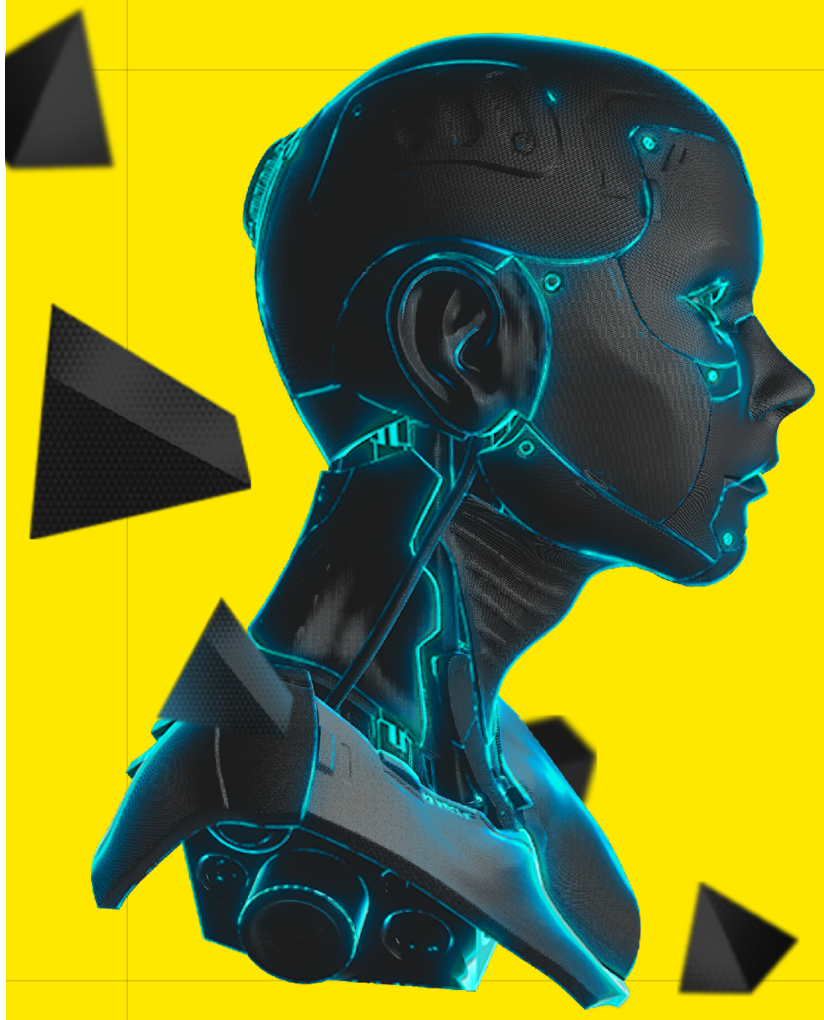
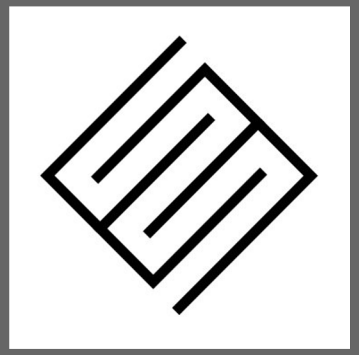


SIGGRAPH 2021

SWISH: NEURAL NETWORK CLOTH SIMULATION ON MADDEN NFL 21

Chris Lewin, James Power, James Cobb

INTRODUCTION



EA SEED

Chris Lewin - Senior Physics Software Engineer

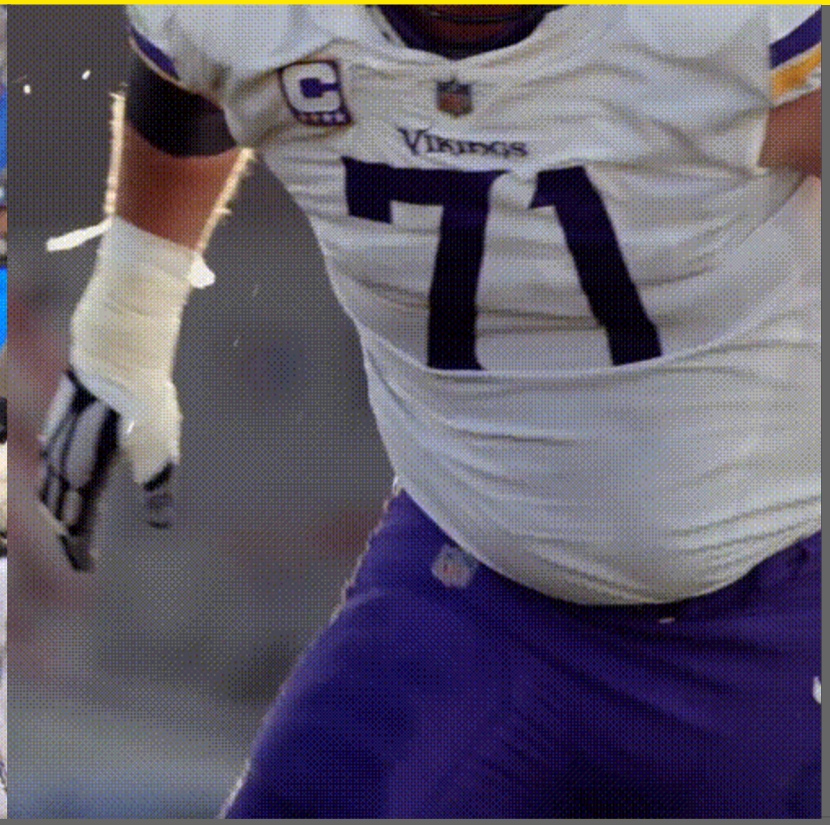
EA Tiburon

James Cobb - Engine Lead

James Power - Rendering Software Engineer

MOTIVATION

SIGGRAPH 2021



COMPARISON

SIGGRAPH 2021

Old System

New System

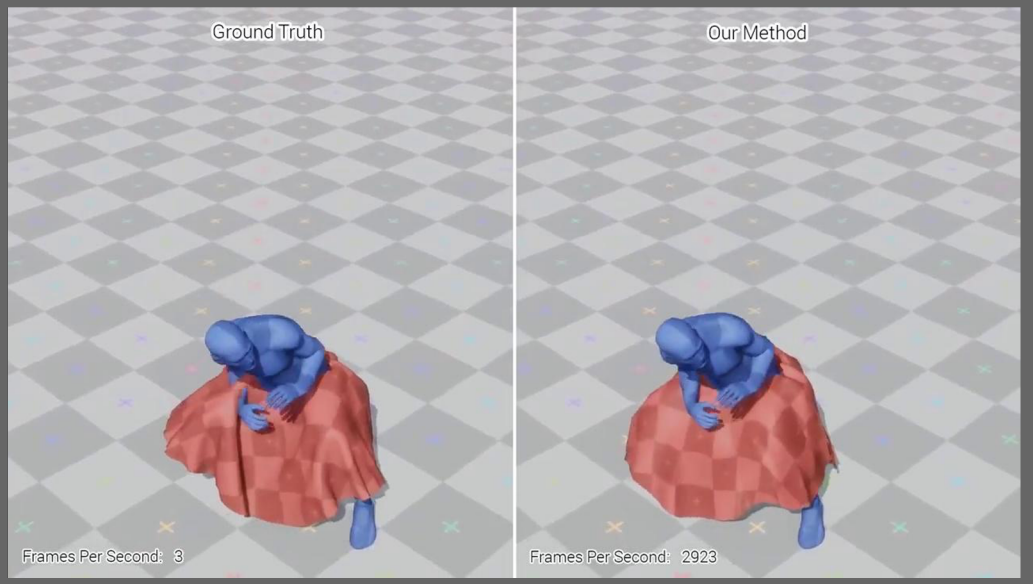
SIGGRAPH 2021

DETAILS

SUBSPACE NEURAL PHYSICS

SIGGRAPH 2021

- [Holden et al 2019]
- Learn physics behaviour with a neural network!
- Doesn't really work for production though
  - Low resolution
- Difficult because it has to learn dynamic behaviour



STATELESS CLOTH DEFORMER

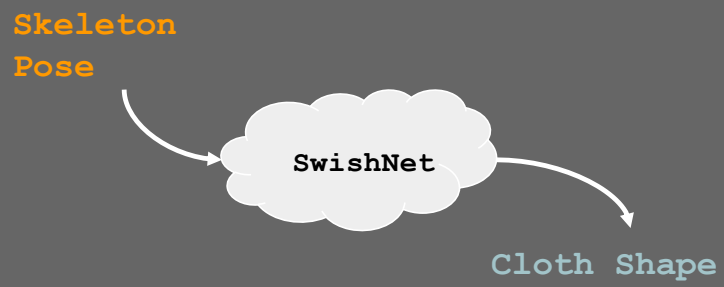
- Let's change the problem!
- Don't try to learn dynamics
- If cloth is tight enough, static behaviour is enough

- Try to learn a **stateless pose based deformer**

- Input: bone deformations
- Output: Cloth state

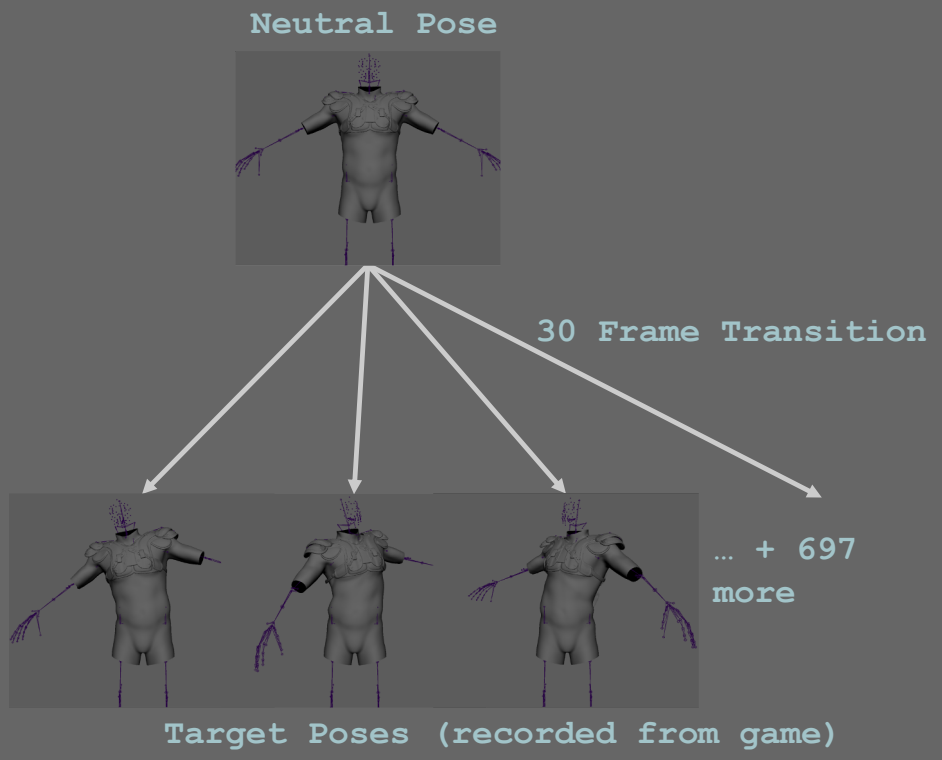
- Can simplify the problem further:
  - Only bottom half of jersey
  - Only look at the 4 spine joints
  - No bending forward/backward

- This makes iteration times feasible



STATELESS CLOTH DEFORMER

- Cloth sims are not static
  - Can't make them static just by changing sim settings
  - They have *intrinsic history dependence*
- Need to make sure deformation history is consistent between pose space samples
- Idea: simulate each pose separately!
- Record all poses the character can get into.
- For each pose, simulate a transition from the neutral to that pose.
- Take the final cloth shape and associate that uniquely with the character pose



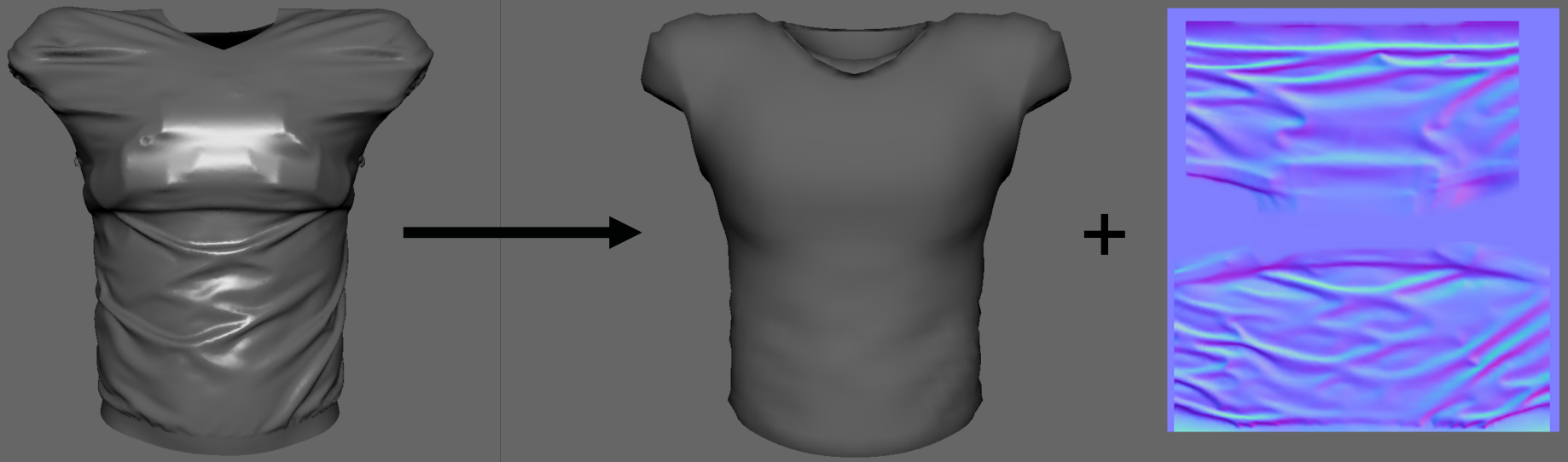
CLOTH SHAPE

SIGGRAPH 2021

Source Simulation

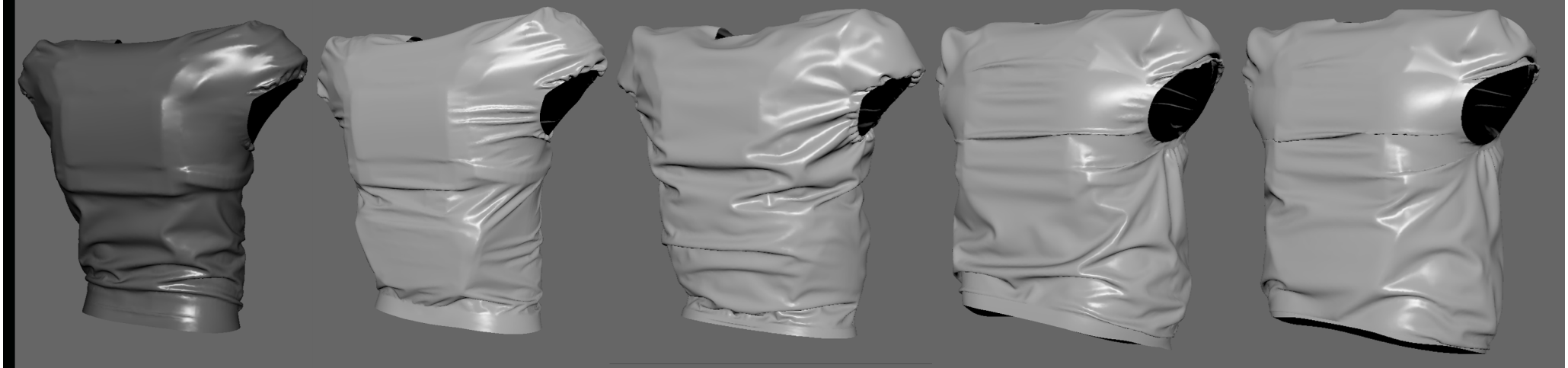
Ingame Mesh

Normal Map



DIFFERENT ASSETS

SIGGRAPH 2021



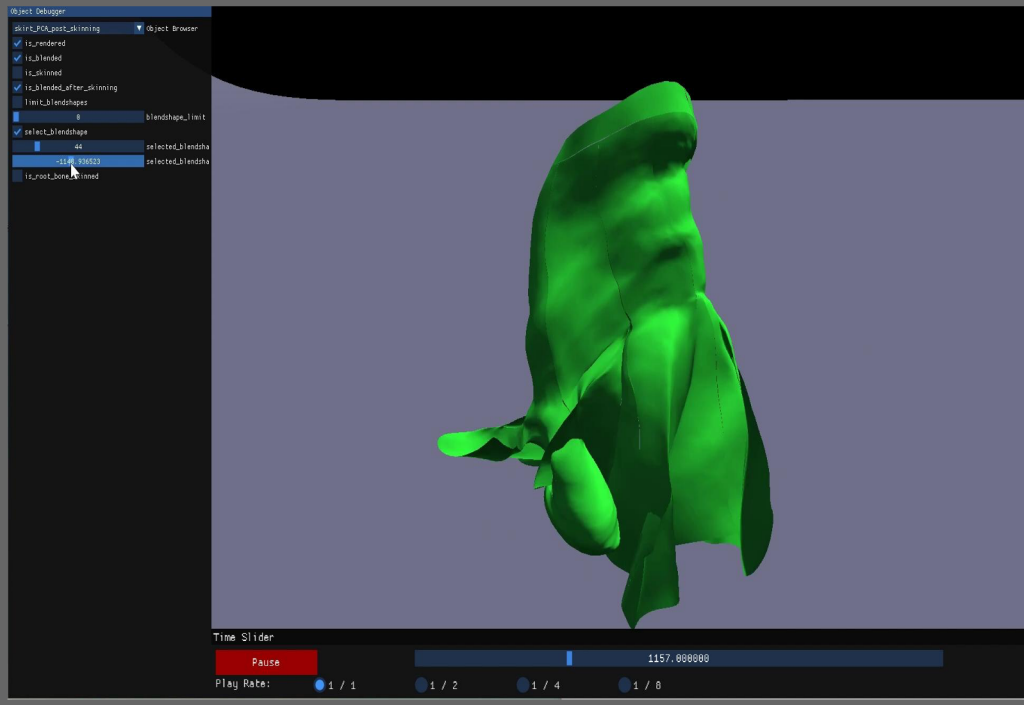
Default Body | Default + Backplate | Default + Flak Jacket | Heavy Body | Heavy + Backplate

MESH DEFORMATIONS

SIGGRAPH 2021

- Input: bone rotations
- Output: Cloth State

- Mesh Deformations: PCA
  - Neural Network outputs ~32 PCA weights
  - GPU adds together weighted sum of 32 PCA shapes.



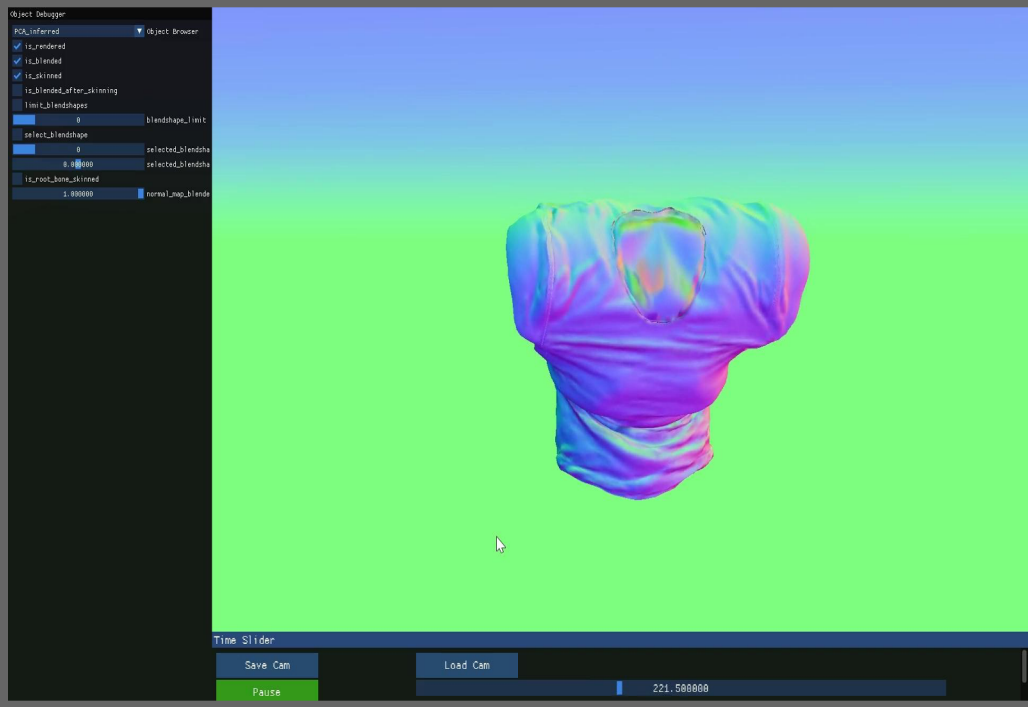
NORMAL MAPS

SIGGRAPH 2021

- Input: bone rotations
- Output: Cloth State

- Mesh Deformations: PCA
  - Neural Network outputs ~32 PCA weights
  - GPU adds together weighted sum of 32 PCA shapes.

- Normal Maps: PCA



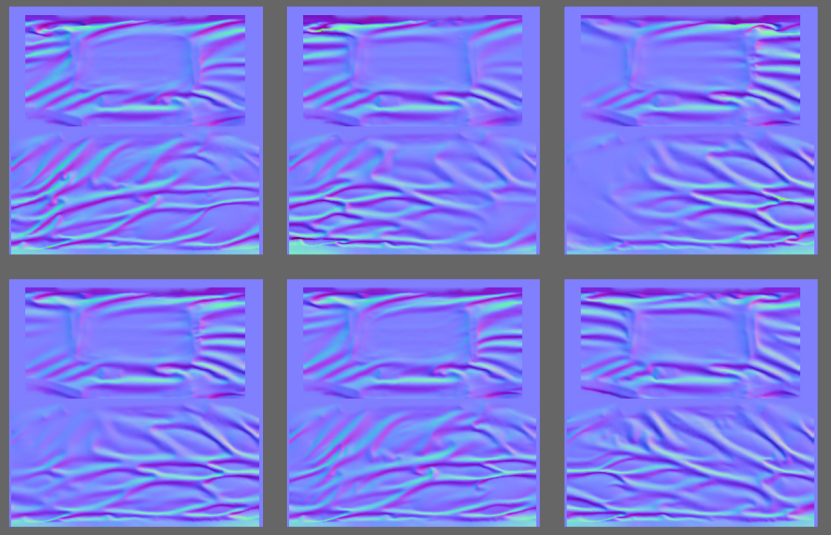
NORMAL MAPS

SIGGRAPH 2021

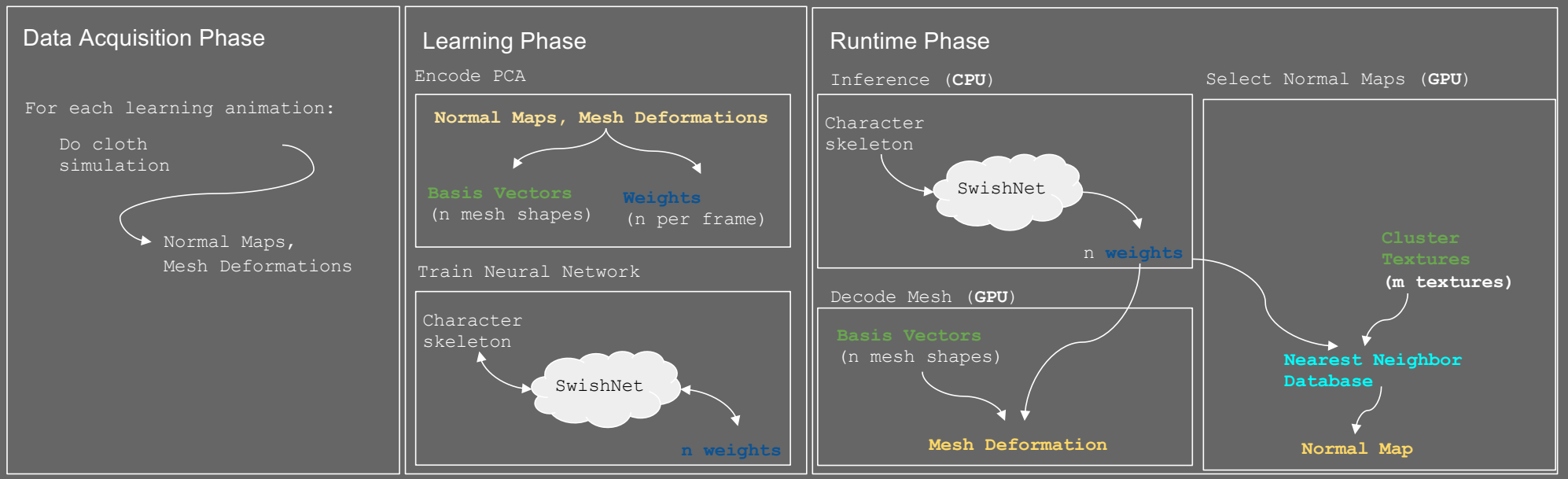
- Input: bone rotations
- Output: Cloth State

- Mesh Deformations: PCA
  - Neural Network outputs ~32 PCA weights
  - GPU adds together weighted sum of 32 PCA shapes.

- ~~Normal Maps: PCA~~
- Normal Maps: Nearest Neighbor Classifier
  - Still have PCA weights…
  - … but use them to look up into database.
  - DB contains 128 selected *original* textures.
    - Static appearance is always **lossless**
    - Dynamic transitions may pop



FULL ALGORITHM



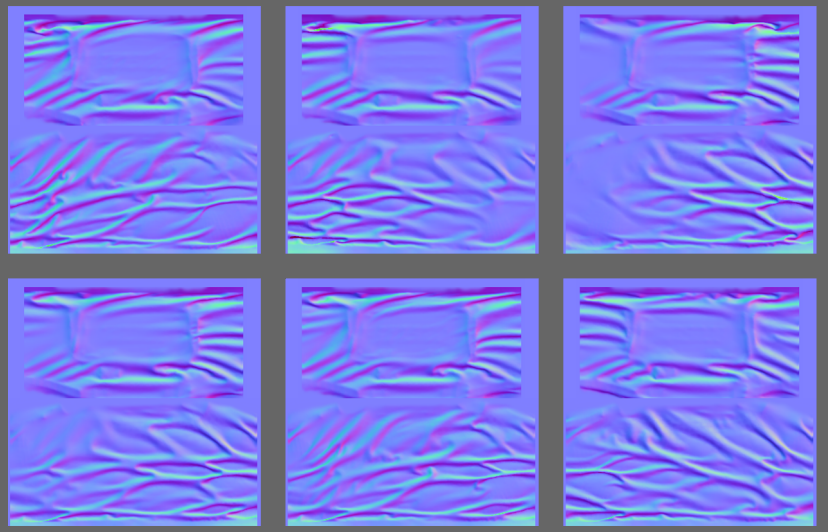
DRAMA FACTOR

SIGGRAPH 2021

- Lack of dynamics means it looks a little boring
- Add a factor that simply exaggerates the bone deviations from the neutral pose by a fixed amount
- This causes the cloth to develop more dramatic poses

NORMAL MAP COMPRESSION

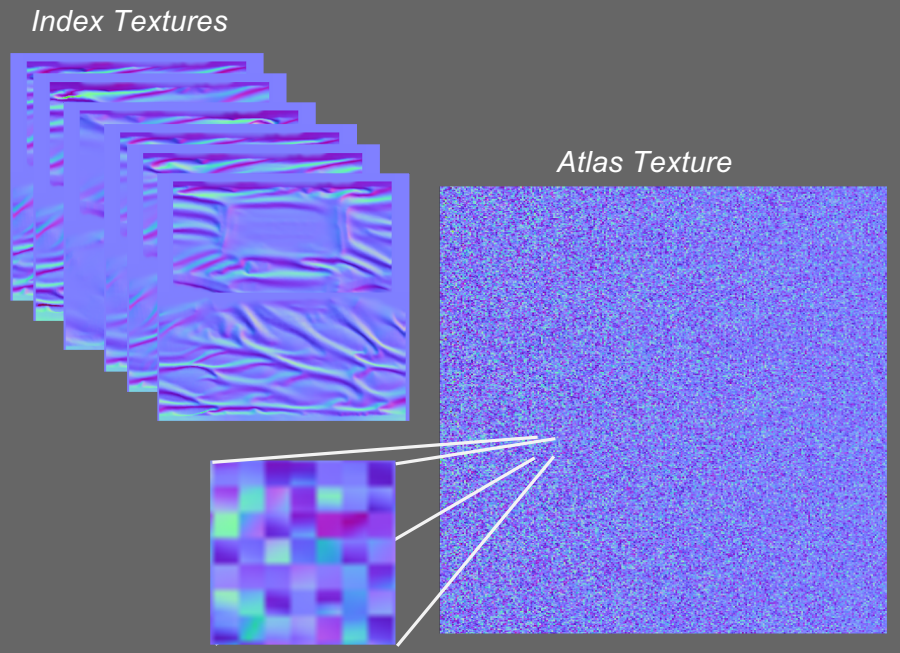
- 128 normal maps… sounds heavy
- 5 assets → 80MB texture memory alone @ 360x352 (BC5)
- They all look the same though!



NORMAL MAP COMPRESSION

- 128 normal maps… sounds heavy
- 5 assets → 80MB texture memory alone @ 360x352 (BC5)
- They all look the same though!

- Compress the whole set of normal maps using Vector Quantization (VQ)

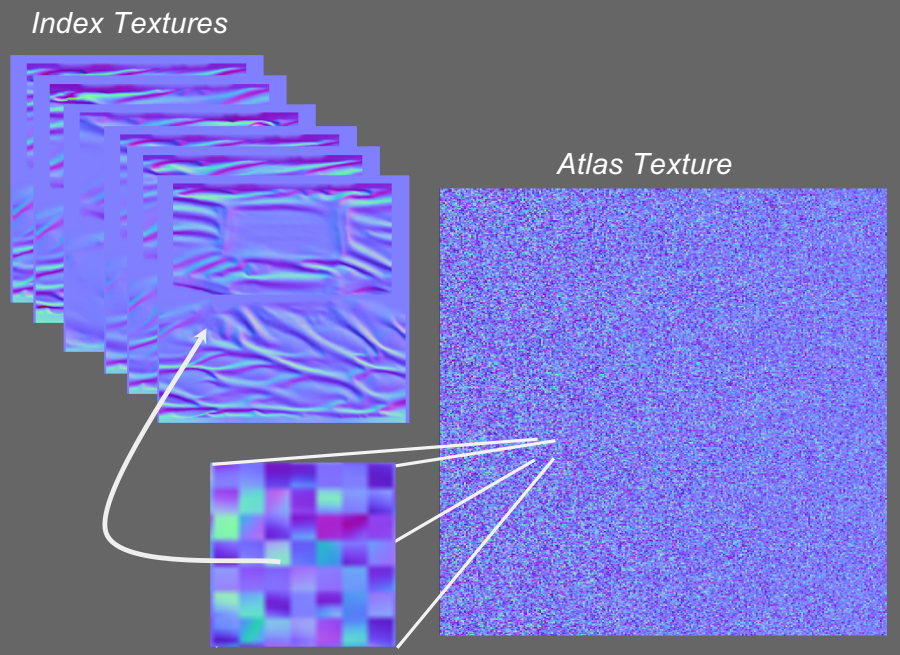


NORMAL MAP COMPRESSION

- 128 normal maps… sounds heavy
- 5 assets → 80MB texture memory alone @ 360x352 (BC5)
- They all look the same though!

- Compress the whole set of normal maps using Vector Quantization (VQ)

- 80MB -> 15MB = ~5x memory saving



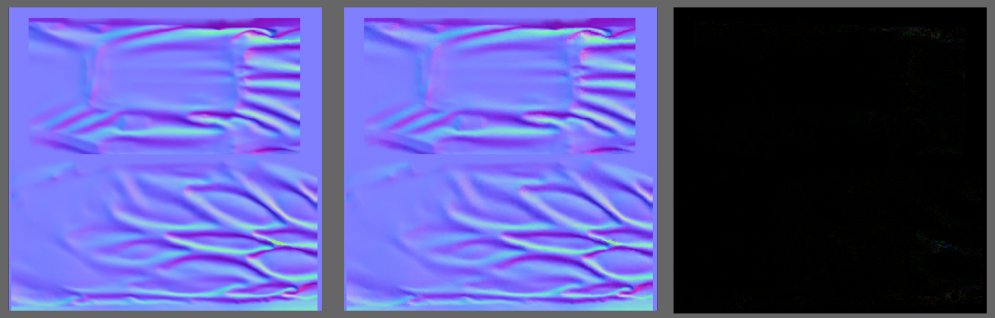
NORMAL MAP COMPRESSION

- 128 normal maps… sounds heavy
- 5 assets → 80MB texture memory alone @ 360x352 (BC5)
- They all look the same though!

- Compress the whole set of normal maps using Vector Quantization (VQ)

- 80MB -> 15MB = ~5x memory saving

- With our assets, difference is undetectable in game



SIGGRAPH 2021

RESULTS

RESULTS

SIGGRAPH 2021

FEATURE COMPARISON

SIGGRAPH 2021

Full Effect

Mesh Only

Normals Only

PERFORMANCE DATA

- Neural Network inference uses a custom C++ library
  - Not heavily optimized
- Most processing time associated with Swish is not neural network related
  - Frostbite MeshCompute overhead
  - Skinning and normal calculation in CS

Stage	Device	Per Player (µs)
Neural Network Inference	CPU	20 (thread time)
Mesh Deform Dispatch	CPU	451 (thread time)
Mesh Deform	GPU	115
Unpack Normal Maps	GPU	5

PERFORMANCE DATA

- Neural Network inference uses a custom C++ library
  - Not heavily optimized
- Most processing time associated with Swish is not neural network related
  - Frostbite MeshCompute overhead
  - Skinning and normal calculation in CS
- With compression, texture DB is relatively small
- Per-instance normal maps are required for our temporal filter
  - Quantity is not optimized

Stage	Device	Per Player (µs)
Neural Network Inference	CPU	20 (thread time)
Mesh Deform Dispatch	CPU	451 (thread time)
Mesh Deform	GPU	115
Unpack Normal Maps	GPU	5

Major Allocations	Single Alloc	Count	Total Alloc
Normal Map DB (Texture Array)	3M	5	15M
Normal Output (per instance)	375K	93	35M
Vertex Offsets	12K	90	1M
Total	~	~	51M

SIGGRAPH 2021

DISCUSSION

LEARNINGS

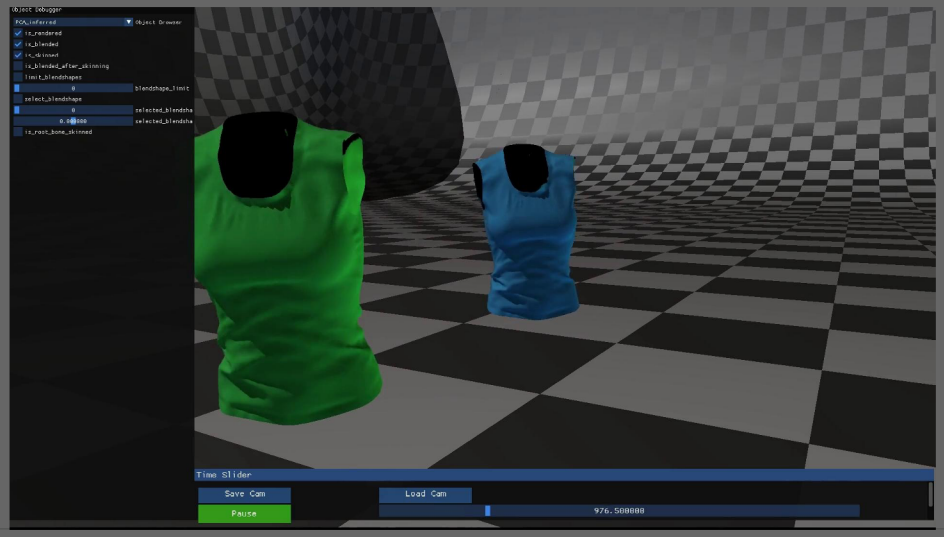
- Neural Networks can be efficient to ‘game standards’
  - We significantly overshot our performance expectations
  - Lots of space to grow into

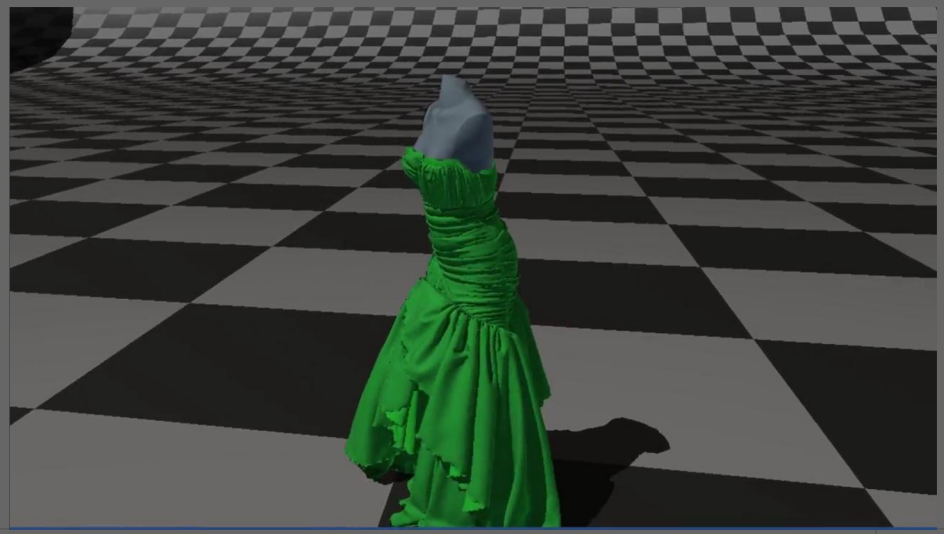
- Offline cloth sims are expensive in time and space
  - We had to decrease the complexity of our model due to very long data generation times (48 hrs -> 6 hrs per asset)

- Efficiently generating image data using a neural network is still difficult
  - Current method is extremely memory hungry relative to its limitations

DYNAMICS

- Static-only is the main restriction of Swish
- Cloth can add much more to a character if it supports dynamics





THANK YOU!

SIGGRAPH 2021

Chris Lewin @magnadiver

Madden @EAMaddenNFL

SEED @SEED

Frostbite @FrostbiteEngine