Further Steps in Precise Shape Completion

- Lab Engineers: Koby Kohai, Orly Widgerson - CRML Lab; Yaron Honen - GIP Lab
- Project Supervisor: Ido Imanuel - GIP Lab
- Project Students: Yiftach Edelstein and Hadas Romov

Motivation

In recent reality, people are looking for new ways to connect with each other.

Virtual reality can be the new meeting ground for people all around the world

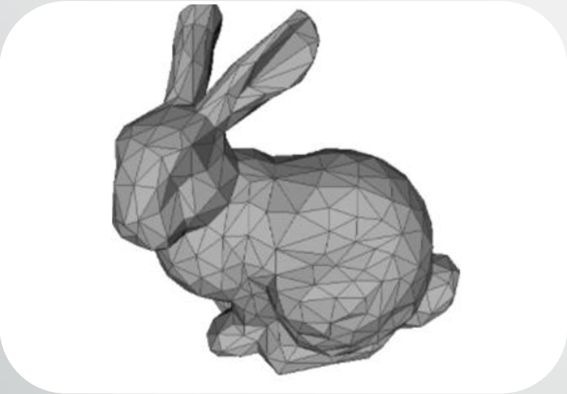
Capturing a person to view in VR is limited by equipment and capture location

We want to achieve a simple setup to enable everyone to participate in this new frontier.

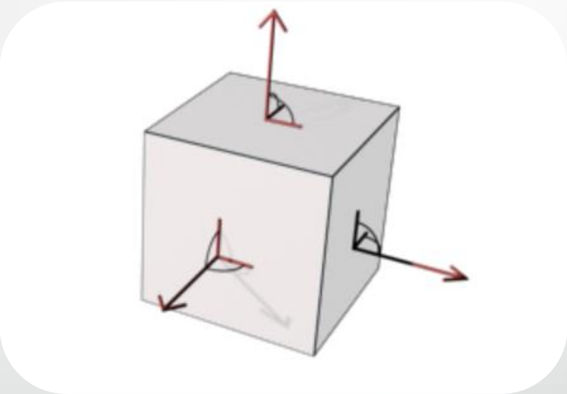
Project statement

- Further investigating issues with the base paper ”Towards precise completion of deformable shapes”
- Improving the network performance in those cases, by changing the base loss functions, architecture, and data processing pipeline

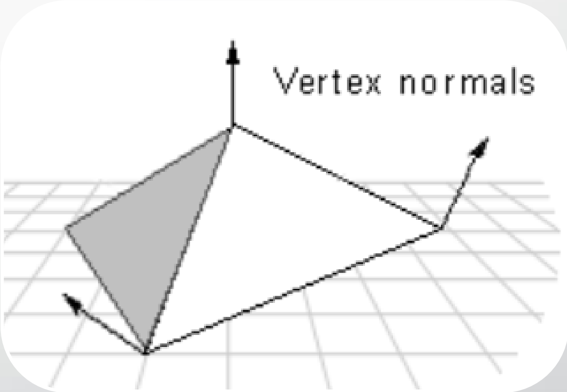
Introduction Training - Geometric Processing & Shape Analysis



Triangular Mesh



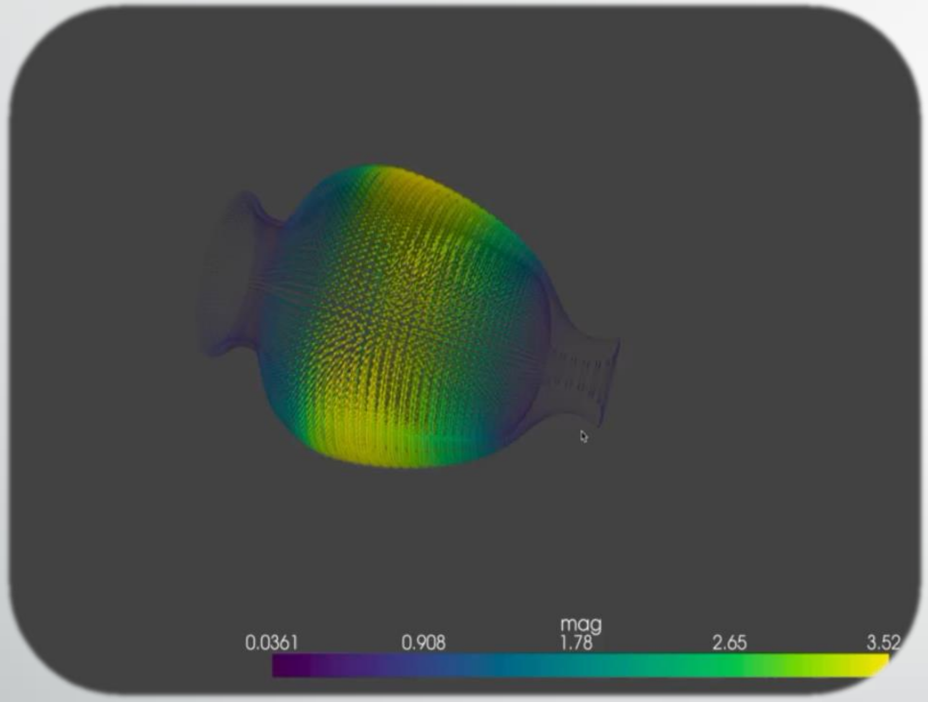
Surface Normal



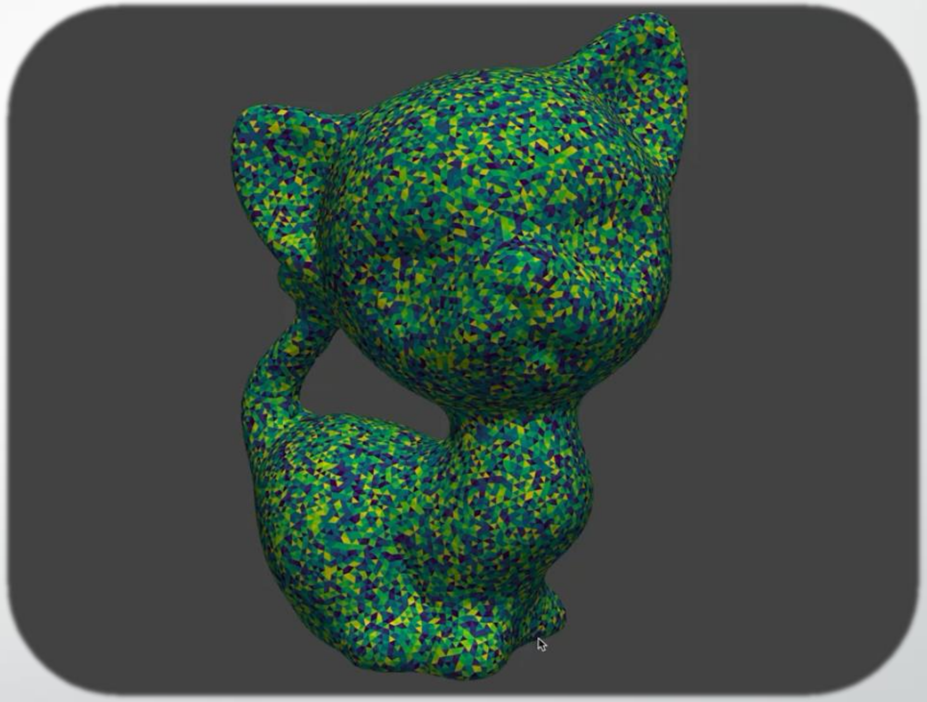
Vertex Normal

CMU’s Keenan Crane and MIT’s Justin Solomon

Visualizing Triangular Meshes



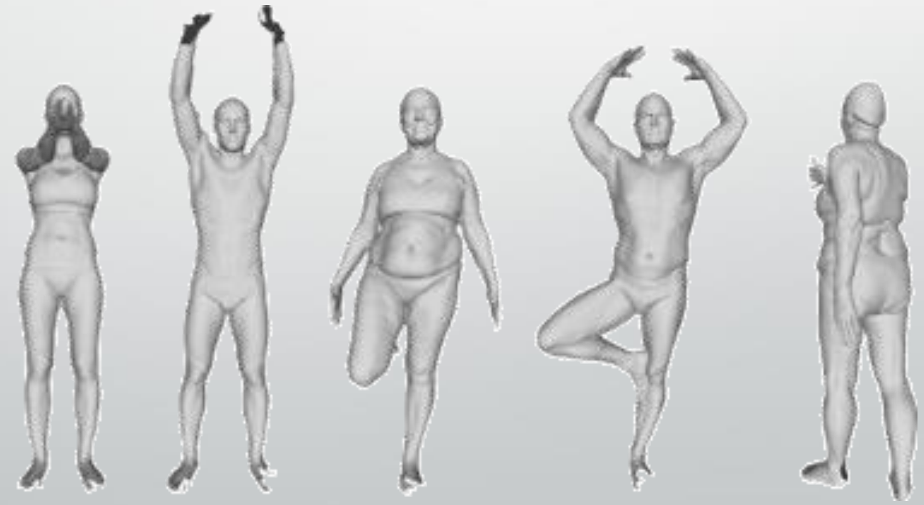
Plotting face normals



Plotting the surface

FAUST data set

- Contains 300 real, high-resolution human scans of 10 different subjects in 30 different poses
- Each scan is a fully corresponding, high-resolution, triangulated mesh acquired with a 3D multi-stereo system.



Dynamic FAUST data set

- Contains 10 subjects, each in a different number of starting poses, captured at 60 fps.
- Extends the FAUST dataset to dynamic 4D data.
- 128 animation sequences
- 40952 total frame count

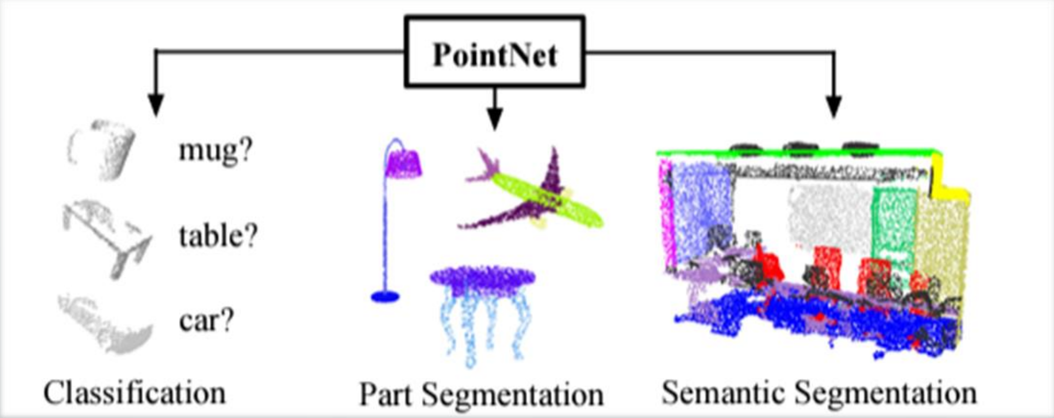
Previous work

"PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation"

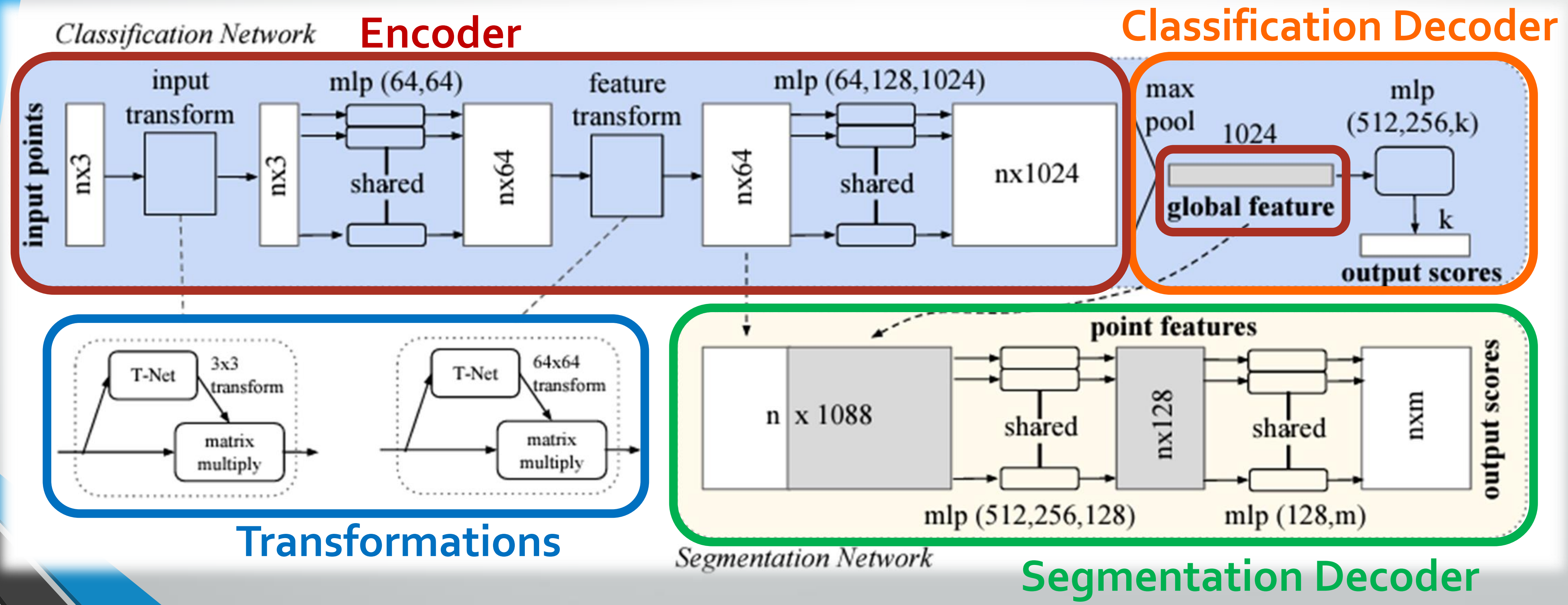
"Towards Precise Completion of Deformable Shapes"

PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation

- Point clouds are data structures of irregular shape
- Deep Learning on point clouds using the popular methods is inefficient
- Other architectures involving basic 3d convolution or RNNs prove inadequate as well
- The paper proposes a novel type of neural network that directly consumes point clouds, which respects the permutation invariance of points in the input.

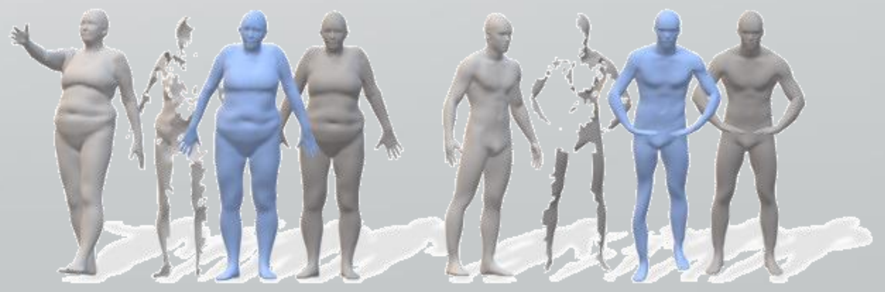


PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation

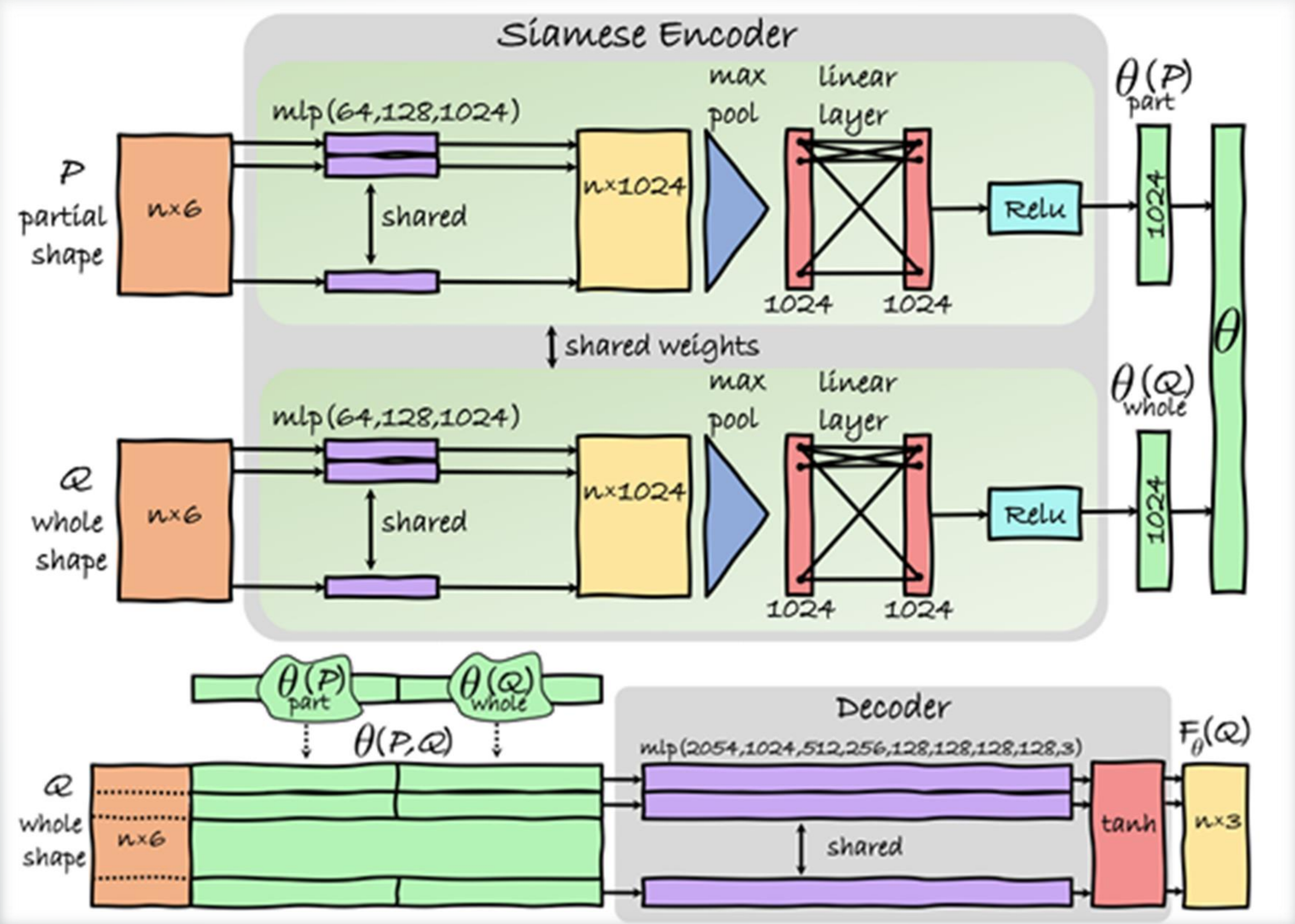


Towards Precise Completion of Deformable Shapes

- Applications require precise completion of incomplete point clouds
- The paper addresses the new problem of matching a partial scan to the whole while reconstructing the new pose from its partial observation
- The proposed model does not require a consistent vertex labeling at inference time
- It can be used on unorganized point clouds as well as on triangular meshes

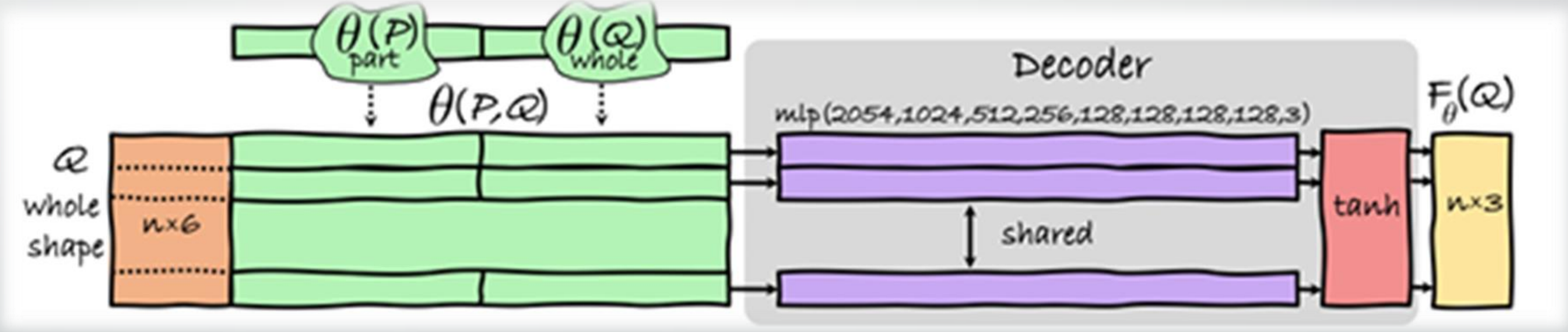


Towards Precise Completion of Deformable Shapes



Towards Precise Completion of Deformable Shapes

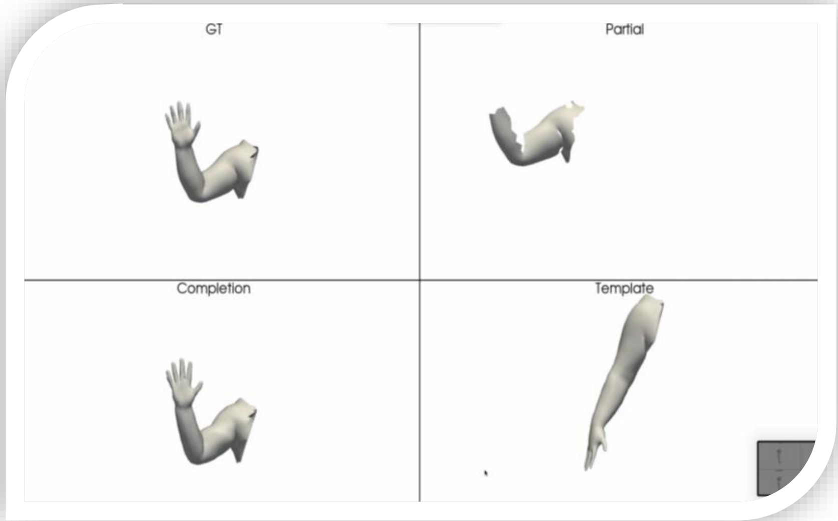
$$L = \underbrace{\sum_{v \in V} \left| v_x - v_y \right|_2^2}_{pointwise\ L_2\ loss} + \lambda \cdot \underbrace{\sum_{v \in V} \left| VN_X(v) - VN_Y(v) \right|_2^2}_{vertex\ normal\ \ L_2\ loss}$$

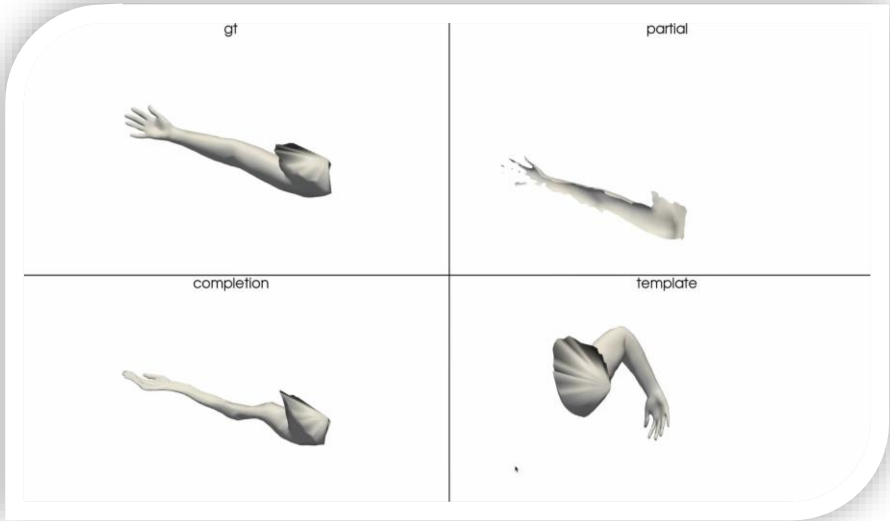


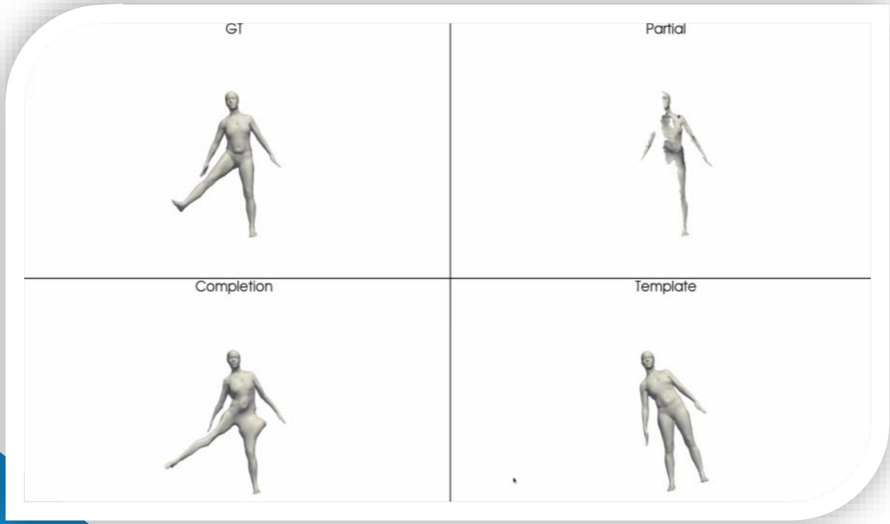
Baseline Results Summary

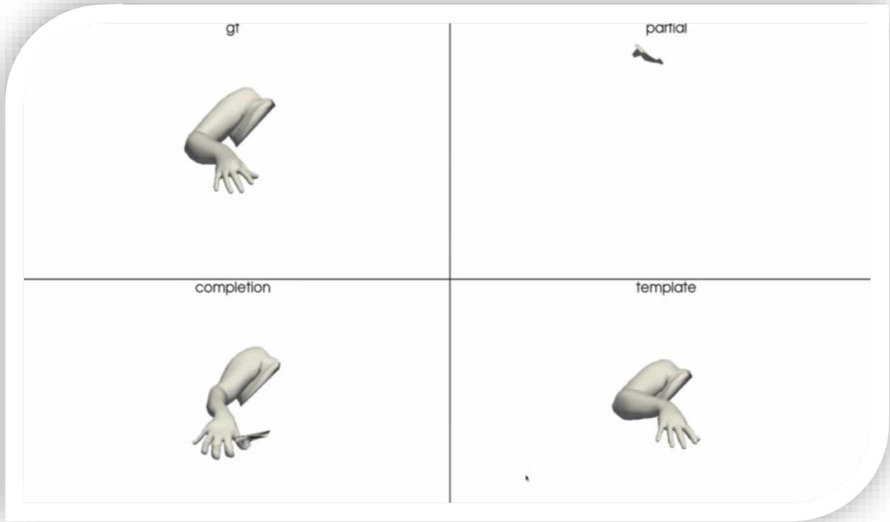
Stage	Completion to GT Vertex MSE	Mean L1 Completion to GT Volume Deformation [%]
Train	0.03735	5.278
Validation	0.03879	8.09
Test	0.04319	9.373

Static Results



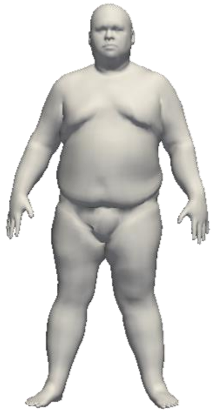






Animations

50002_punching



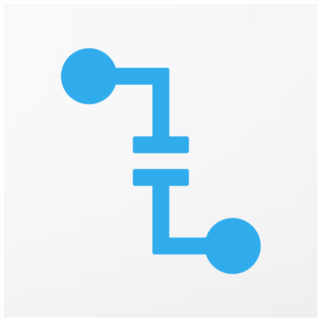
50004_knees

50004_shake_hips

50009_hips

Our Solution

Loss function modifications:



Network architectural Changes:

To handle the spatial deformations

Loss function modifications – Signed Volumetric Loss

We define it as such:

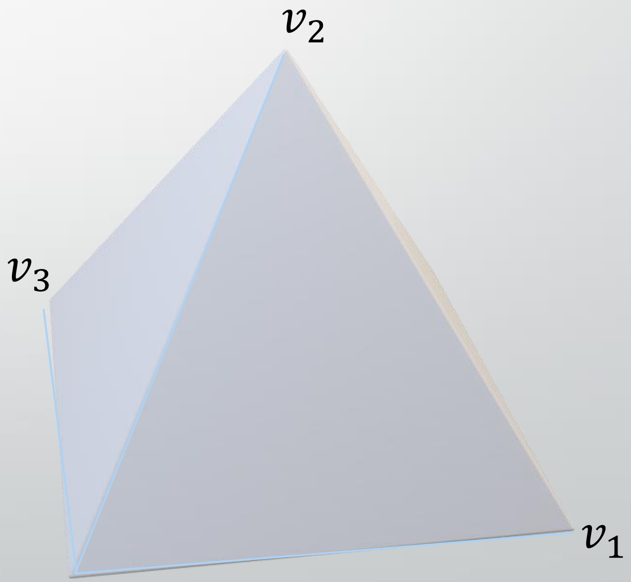
$let\ f \in F$ where F is the Faces of the mesh.

$let\ V_X(f)$ be the signed volume of the face in the completion, and $V_Y(f)$ the signed volume in the Ground Truth

$$L_{volumetric} = \sum_{f \in F} |\, V_X(f) - V_Y(f)|_2^2$$

By applying the tetrahedral volume of each triplet of points, we try and add a term that will push the network to preserve mesh volume (per face).

$$Volume = v_1 \cdot (v_2 \times v_3)$$

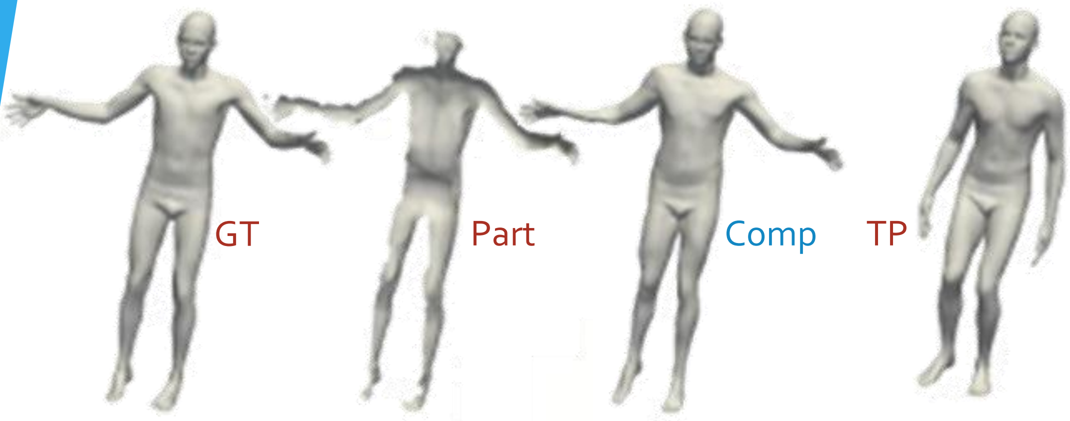


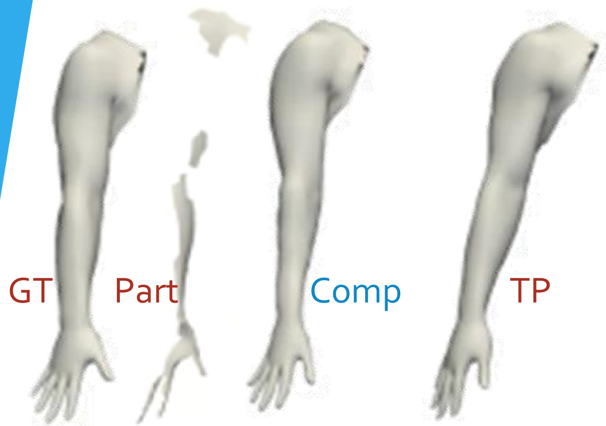
Volume Loss

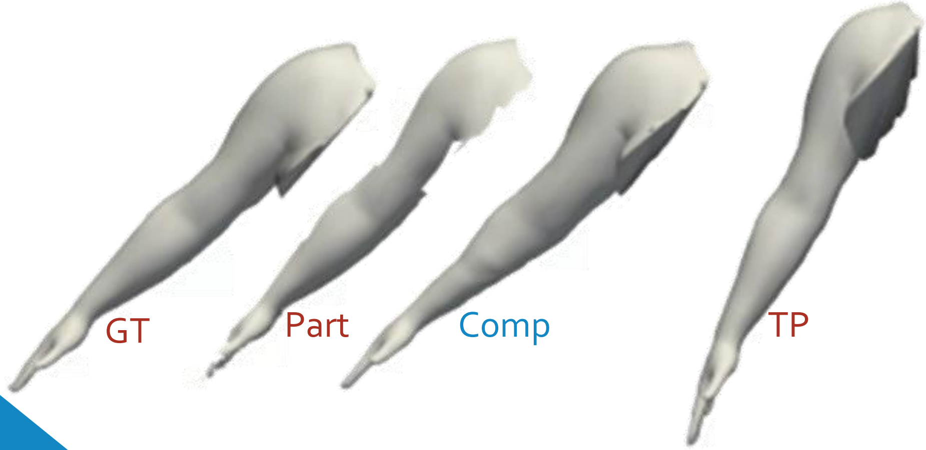
- Modifying the loss function to handle spatial deformations
- Using Segmentation maps, "choose" on which body part to calculate volume deformation
- Use said volume to calculate the L2 Error of that specific body part
- We ran the following experiments:

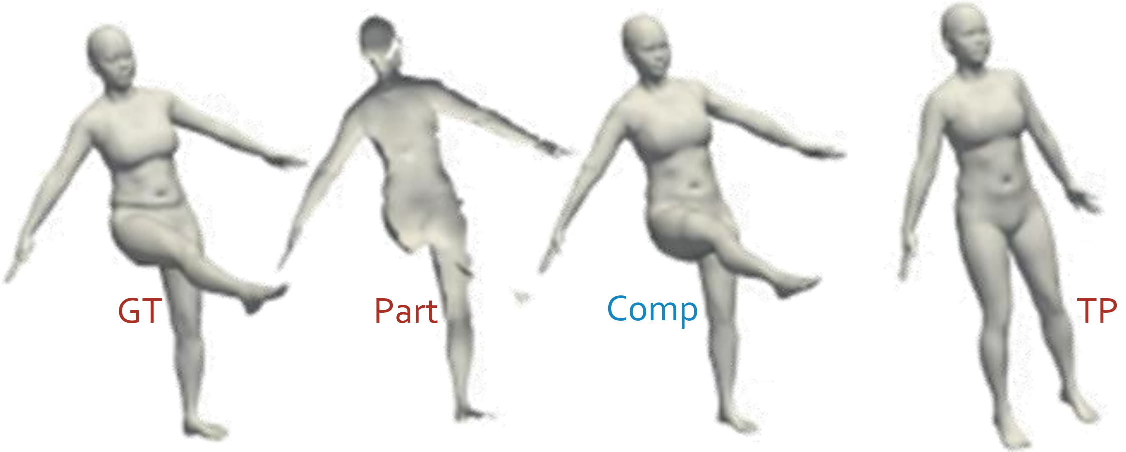
- Full body volume loss
- Right arm volume loss
- Left arm volume loss
- Both arms volume loss

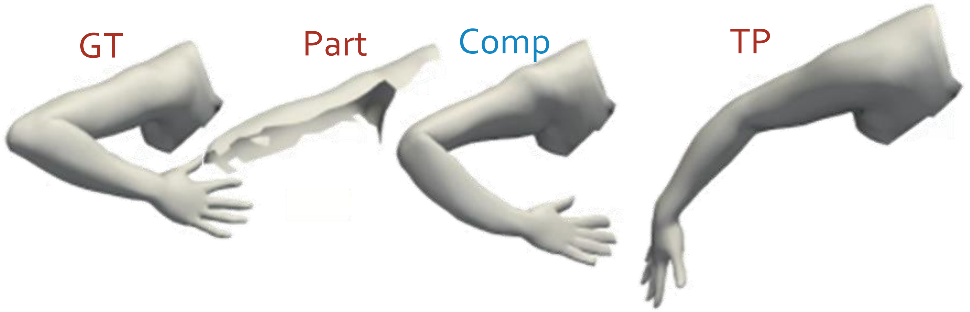
Results

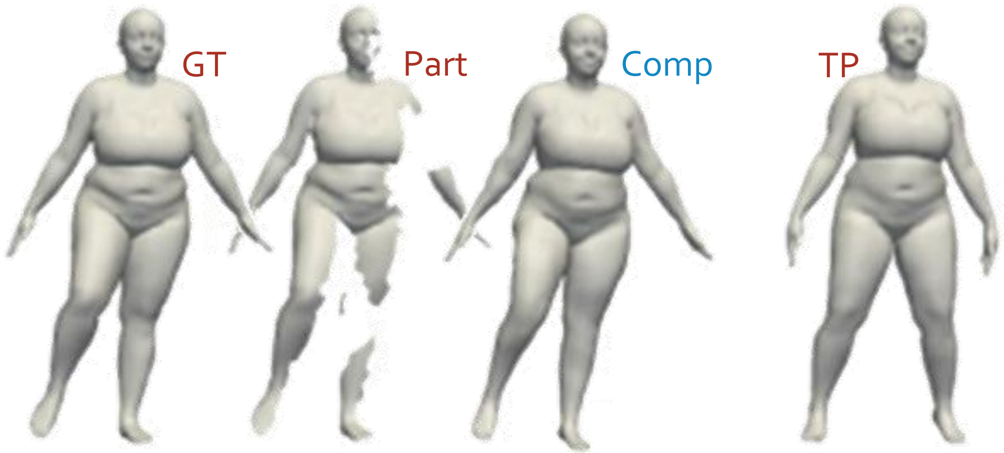

GT
Part
Comp
TP


GT
Part
Comp
TP

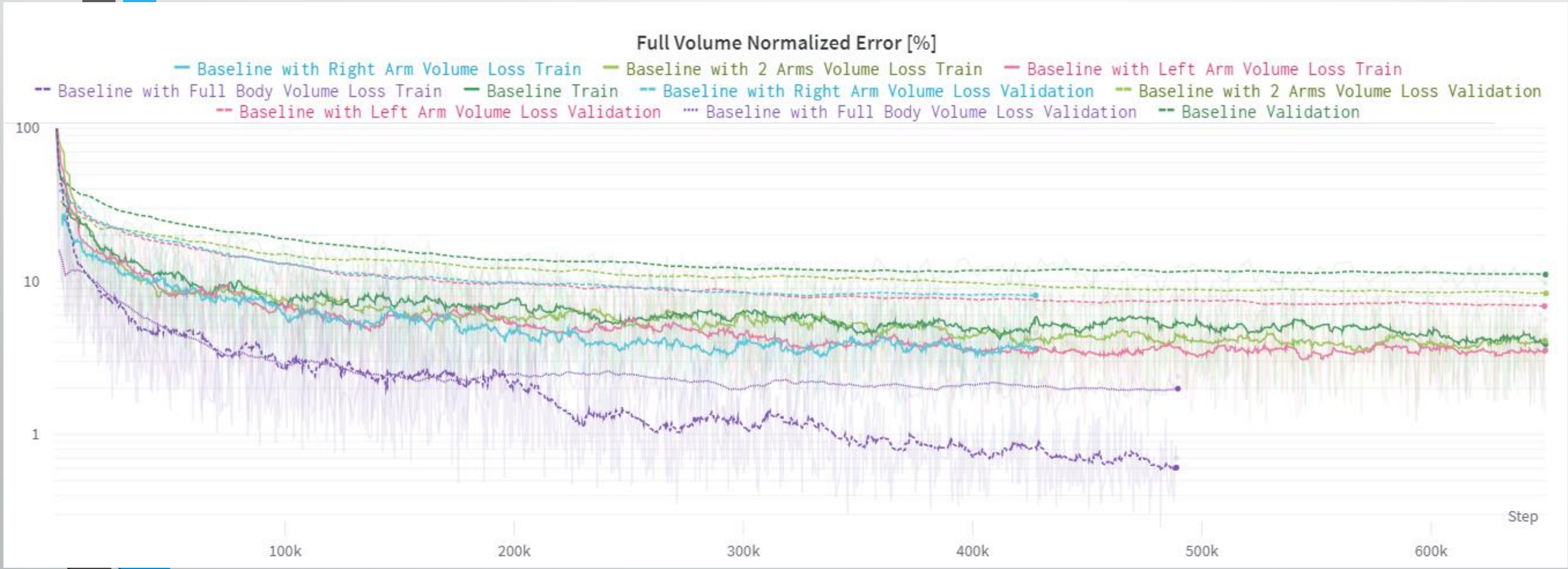

GT
Part
Comp
TP


GT
Part
Comp
TP

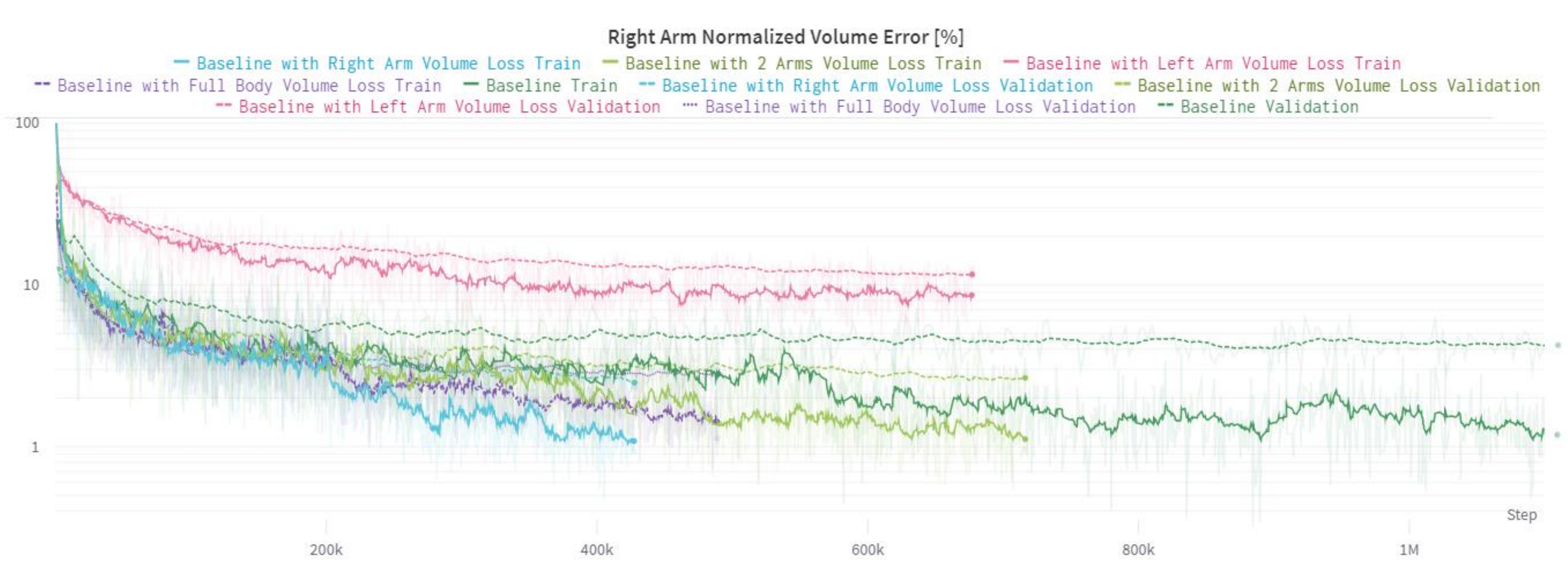

GT
Part
Comp
TP


GT
Part
Comp
TP

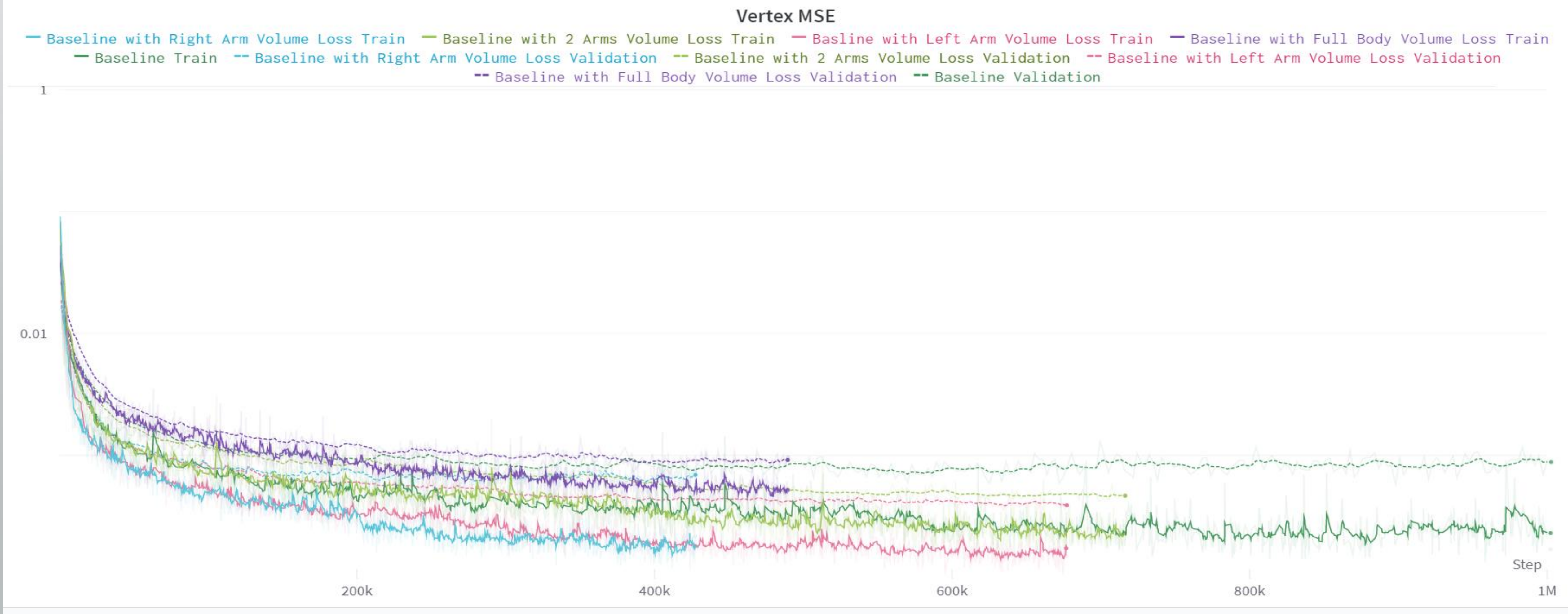
Errors Compared across Experiments



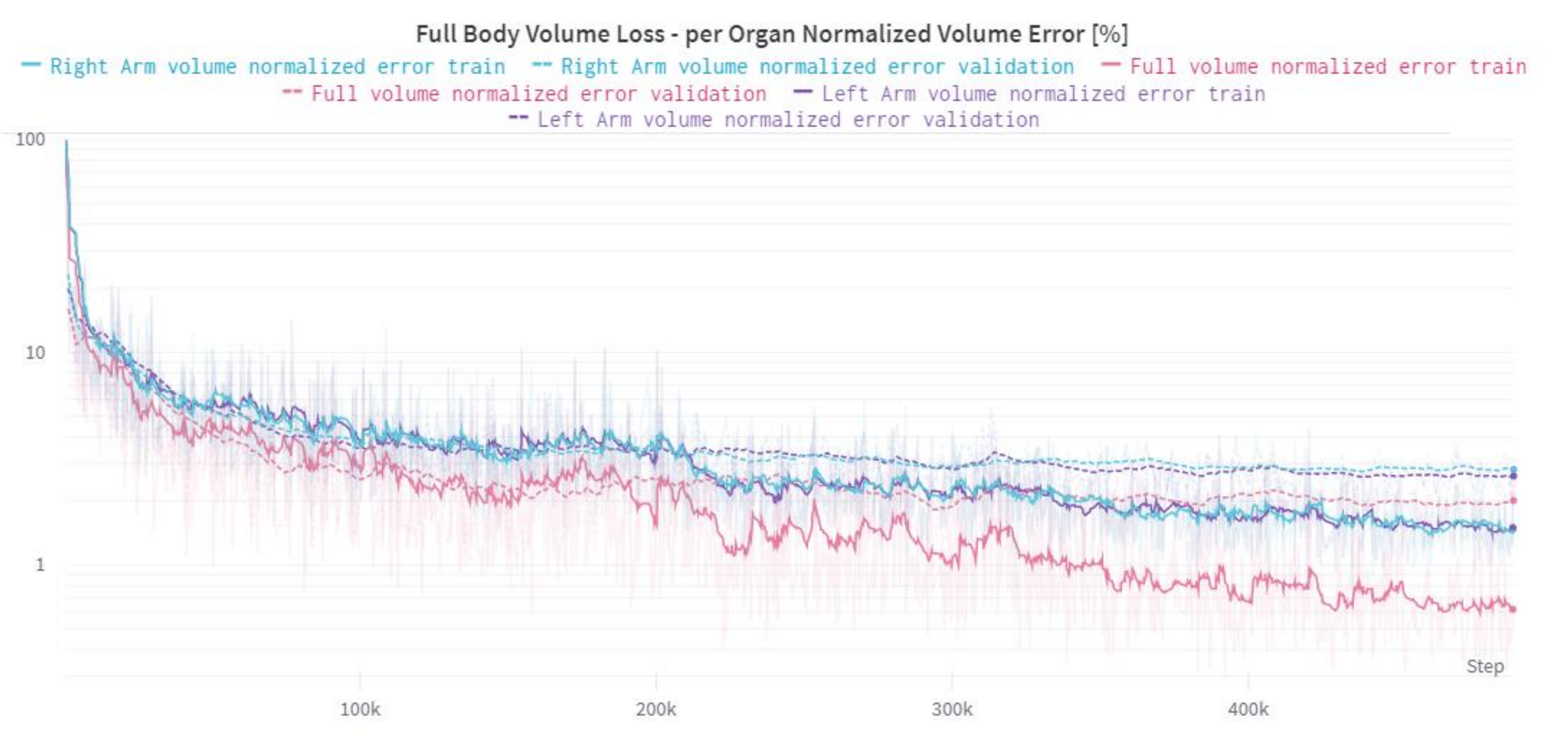
Errors Compared across Experiments



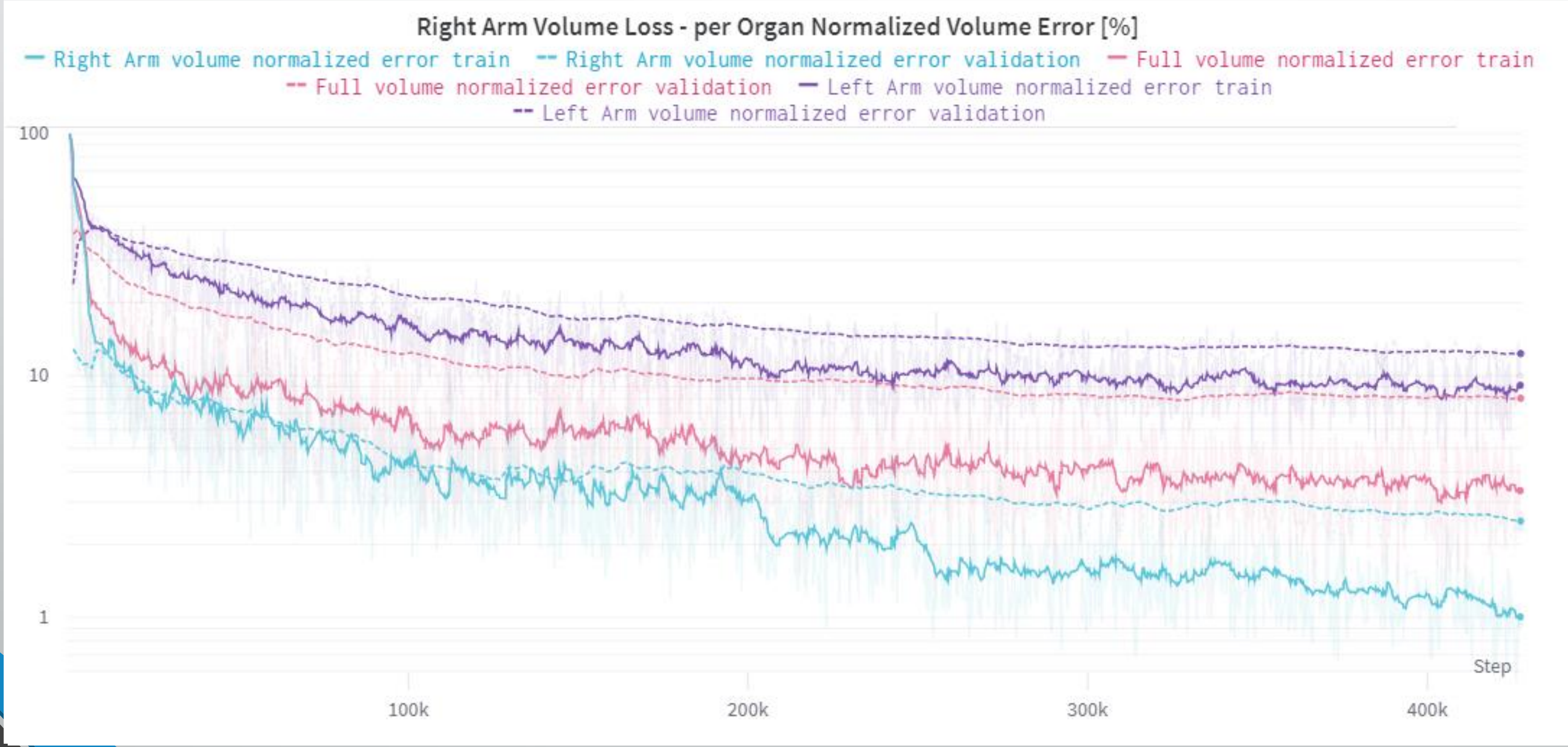
Errors Compared across Experiments



Volume Errors per specific Experiment

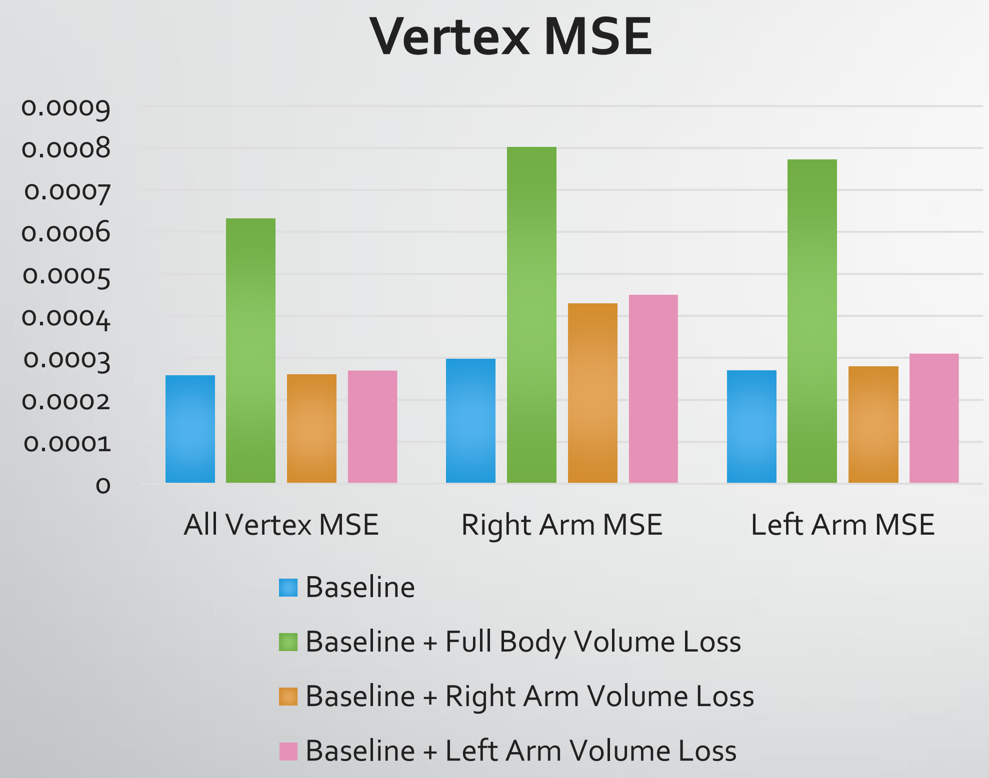


Volume Errors per specific Experiment

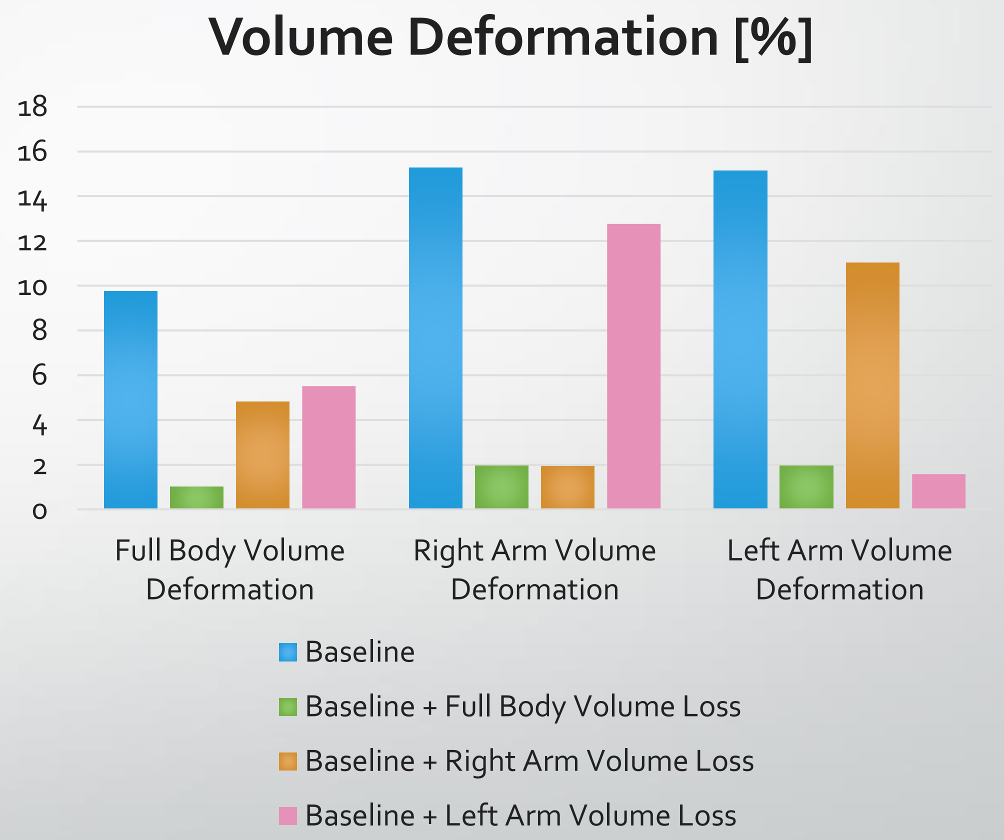


Best Mean Test Scores

Experiment/metric	Baseline	Baseline + Full Body Volume Loss	Baseline + Right Arm Volume Loss	Baseline + Left Arm Volume Loss
All Vertex MSE	0.0002581	0.0006321	0.000261	0.00027
Right Arm MSE	0.0002973	0.0008013	0.0004295	0.00045
Left Arm MSE	0.0002709	0.000772	0.0002793	0.00031
Full Body Volume Deformation [%]	9.746	1.005	4.827	5.517
Right Arm Volume Deformation [%]	15.258	1.958	1.916	12.765
Left Arm Volume Deformation [%]	15.138	1.967	11.036	1.585

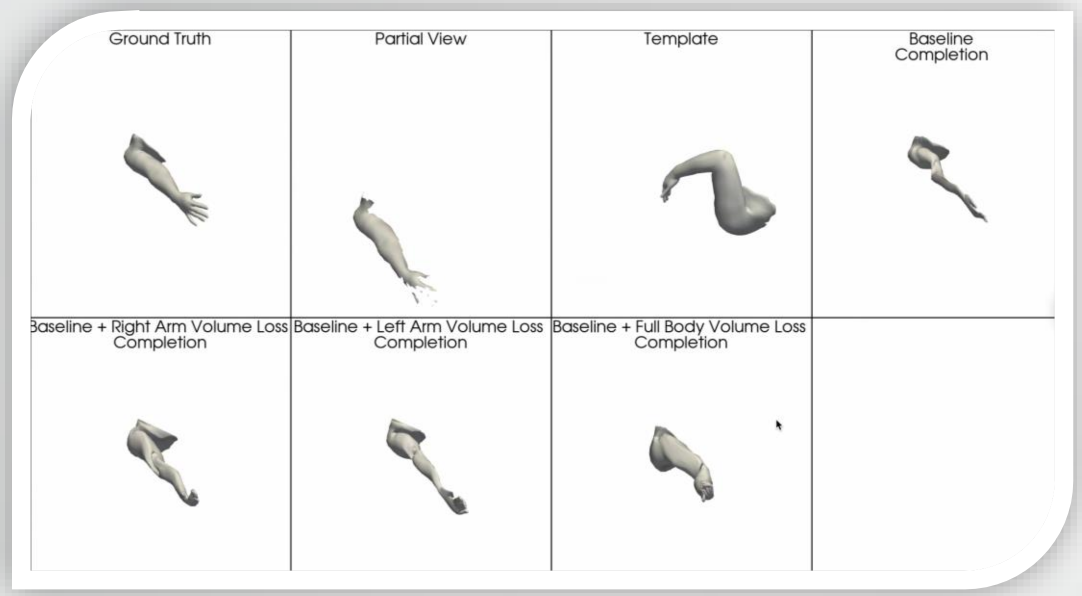


Vertex MSE
0.0009
0.0008
0.0007
0.0006
0.0005
0.0004
0.0003
0.0002
0.0001
0
All Vertex MSE
Right Arm MSE
Left Arm MSE
Baseline
Baseline + Full Body Volume Loss
Baseline + Right Arm Volume Loss
Baseline + Left Arm Volume Loss



Volume Deformation [%]
18
16
14
12
10
8
6
4
2
0
Full Body Volume Deformation
Right Arm Volume Deformation
Left Arm Volume Deformation
Baseline
Baseline + Full Body Volume Loss
Baseline + Right Arm Volume Loss
Baseline + Left Arm Volume Loss

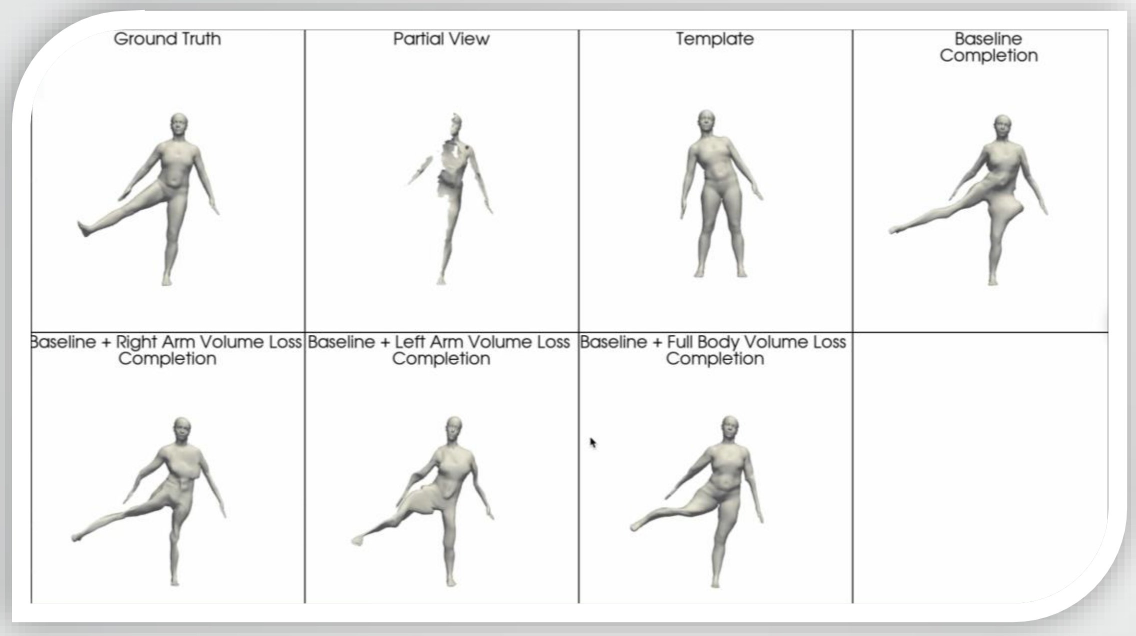
Deep Dive – Right Arm Completion



GT Volume: 0.005207

	Baseline	Baseline +Right Arm Volume Loss	Baseline +Left Arm Volume Loss	Baseline +Full Body Volume Loss
GT-comp Vertex MSE	0.000388	0.000774	0.000543	0.002692
Completion Volume	0.002719	0.005396	0.003469	0.005667
Volume Deformation [%]	47.7818	3.6182	33.3789	8.8304

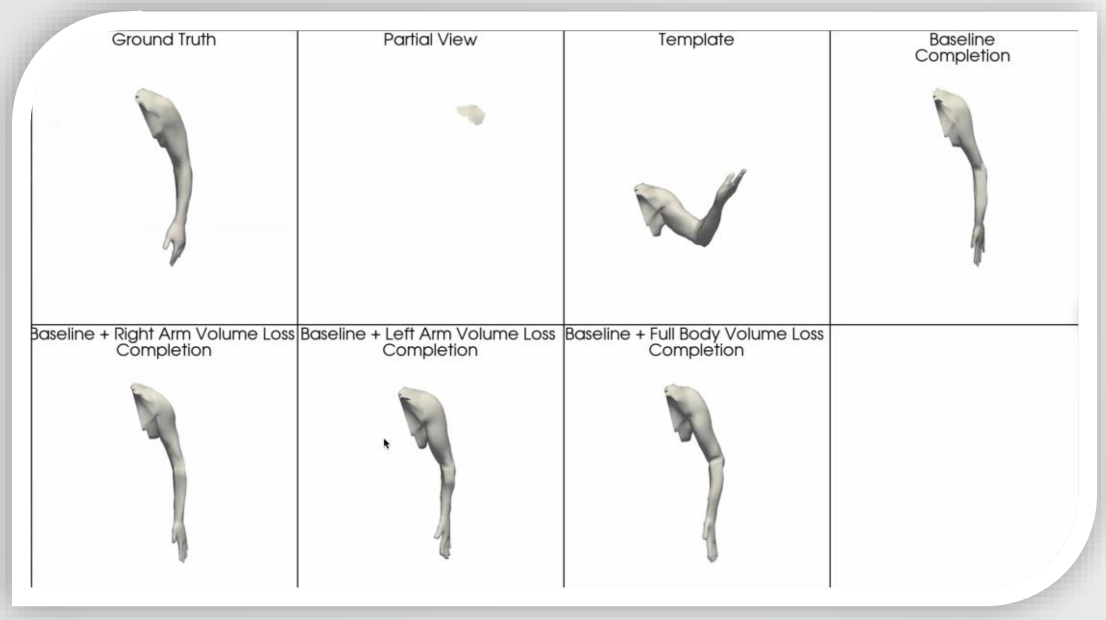
Deep Dive – Full Body Completion



GT Volume: 0.06814

	Baseline	Baseline + Right Arm Volume Loss	Baseline + Left Arm Volume Loss	Baseline +Full Body Volume Loss
GT-comp Vertex MSE	0.001797	0.001568	0.001746	0.001753
Completion Volume	0.058038	0.051763	0.053462	0.066683
Volume Deformation [%]	14.8224	24.0387	21.5466	2.1455

Deep Dive – Left Arm Completion



GT Volume: 0.006344

	Baseline	Baseline + Right Arm Volume Loss	Baseline + Left Arm Volume Loss	Baseline +Full Body Volume Loss
GT-comp Vertex MSE	0.000924	0.000474	0.000546	0.000369
Completion Volume	0.003650	0.004096	0.006329	0.006094
Volume Deformation [%]	42.4609	35.4364	2.3926	3.9412

Analysis

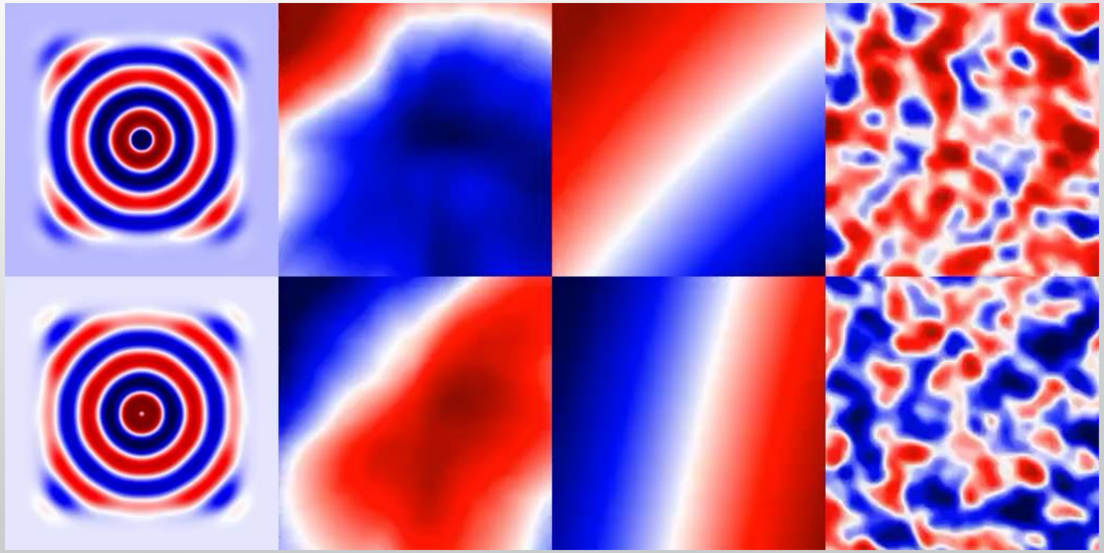
- As can be seen, applying volume loss on a specific organ improves its volumetric error, while compensating for that by severing the other organs' volume errors.
- Compensation is also evident compared to the other metric – adding the volume loss term minimizes the matching organ's volumetric error, but increases the vertex MSE.
- These results are also observed in the displayed examples.

Quick Overview over "Implicit Neural Representations with Periodic Activation Functions"

- The problem of quickly encapsulating high frequencies and low frequencies at the same time, in a set number of parameters, is very difficult.
- Some approached this issue and attempted to solve it with positional encoding – essentially multiplying the data at input, with many different sine functions, with varying success

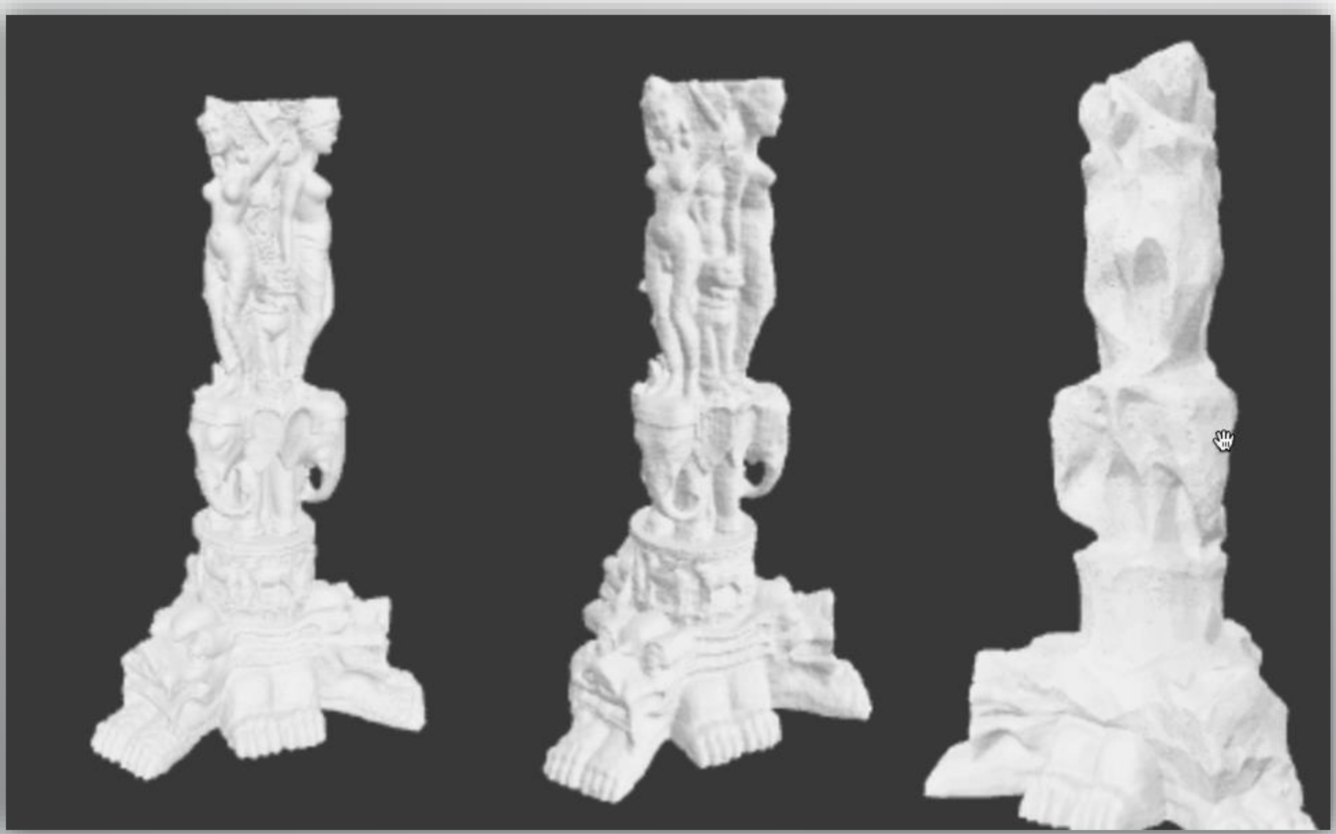
Quick Overview over "Implicit Neural Representations with Periodic Activation Functions"

- Instead, the paper suggests that simply replacing our activations with sinusoidal activations, helps the network learn, and encode high frequencies and low frequencies, with greater ease

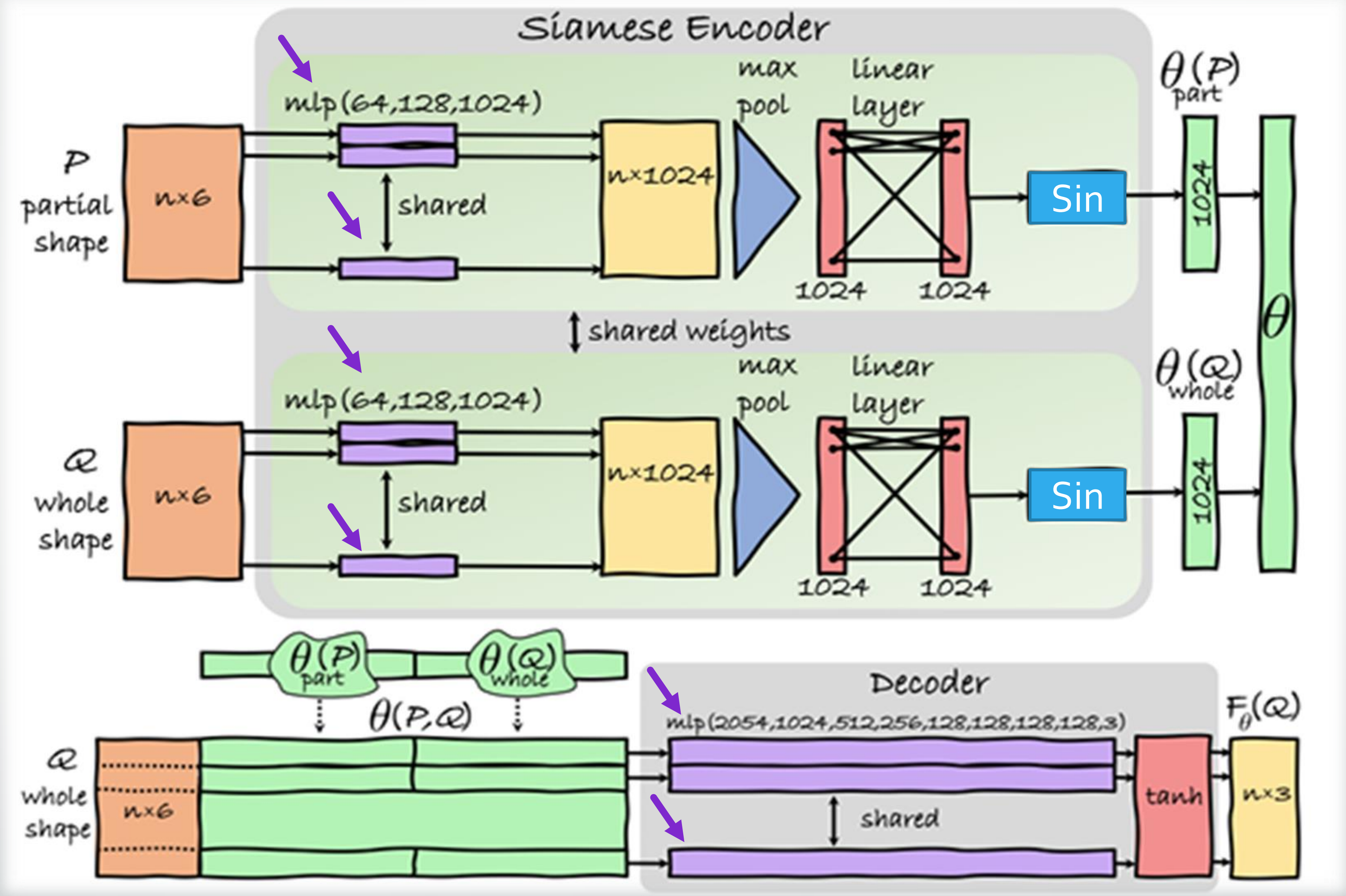


Quick Overview over "Implicit Neural Representations with Periodic Activation Functions"

Siren ReLU Pos. Encoding ReLU

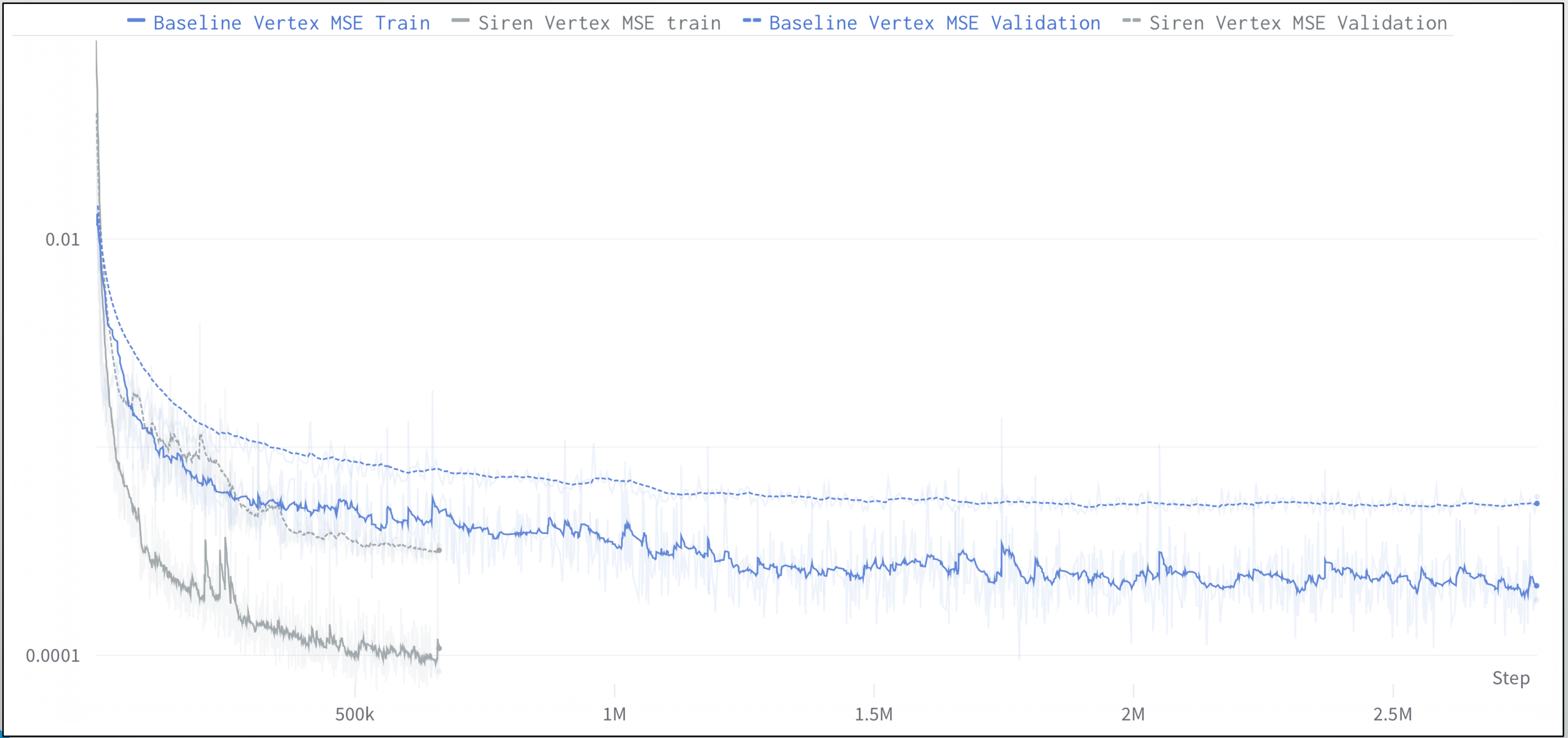


Siren Based Architecture



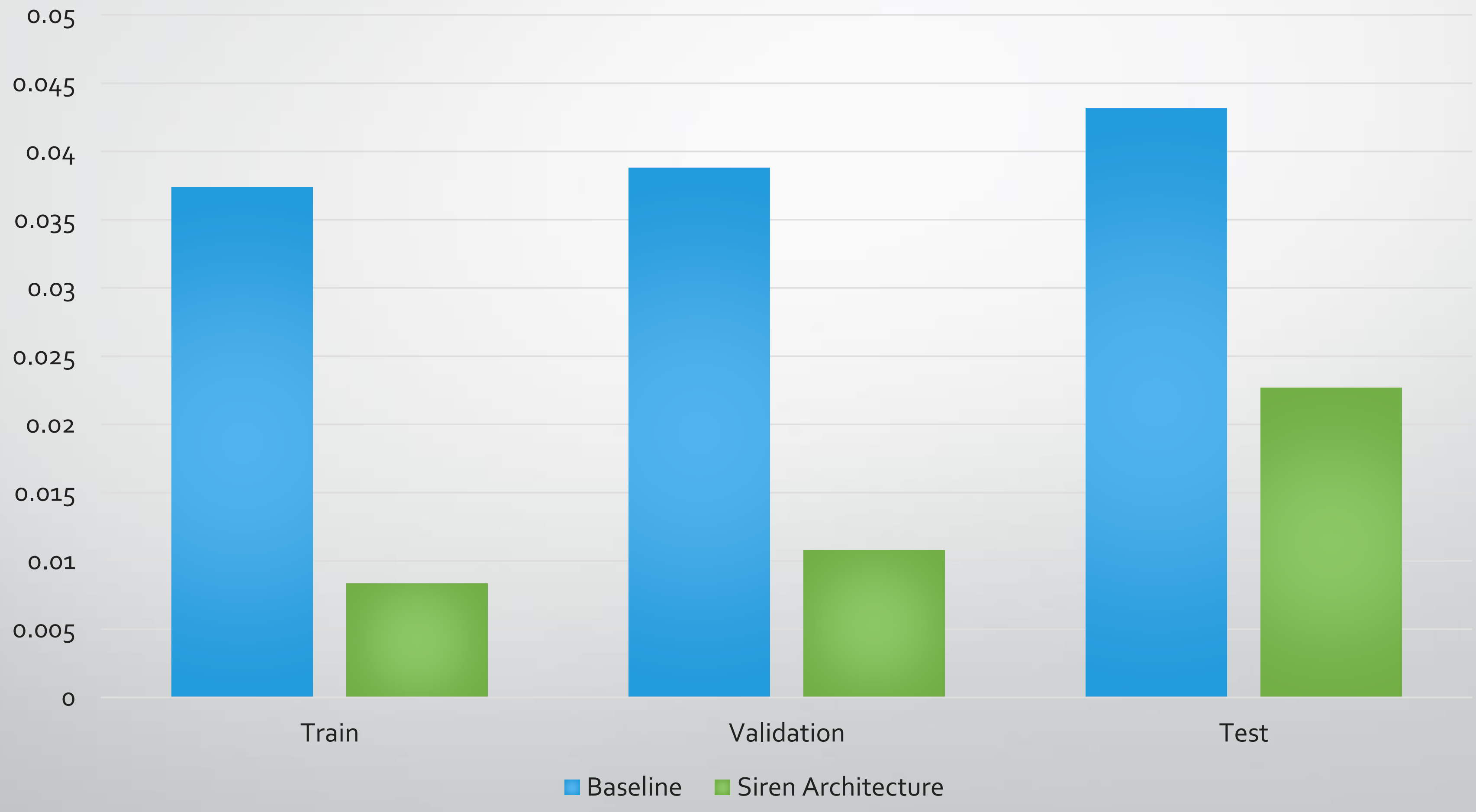
*Sinusoidal activations also go inside the MLP layers in the Decoder and Siamese Encoders

Vertex MSE

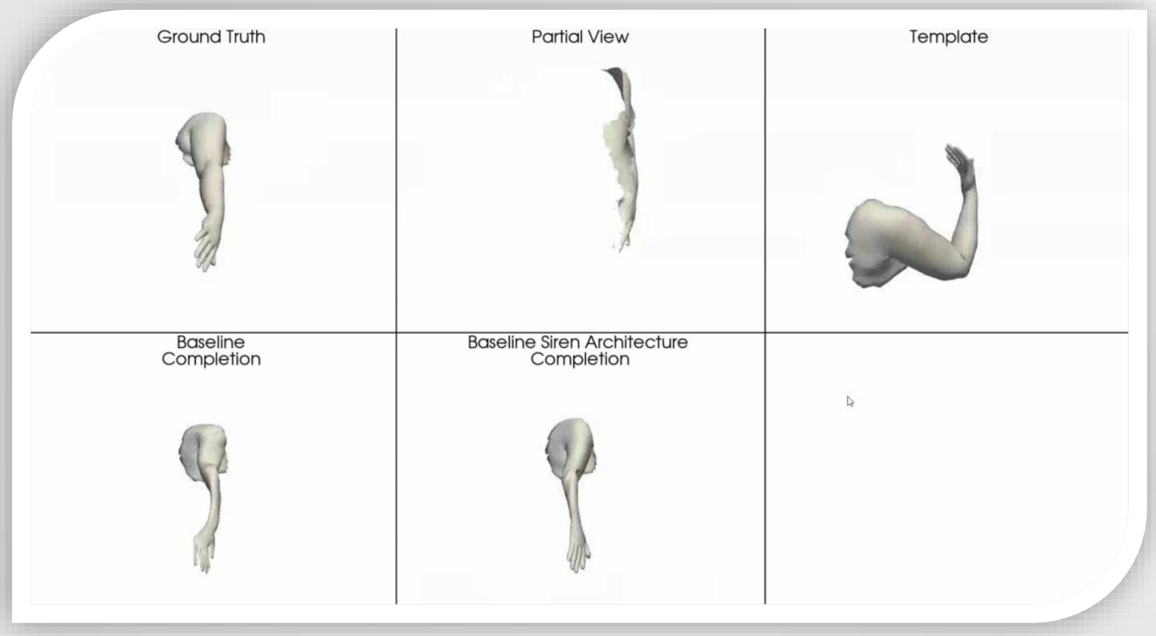


Baseline Vertex MSE Train
Siren Vertex MSE train
Baseline Vertex MSE Validation
Siren Vertex MSE Validation
0.01
0.0001
Step
500k
1M
1.5M
2M
2.5M

Siren Architecture – Vertex MSE Improvement



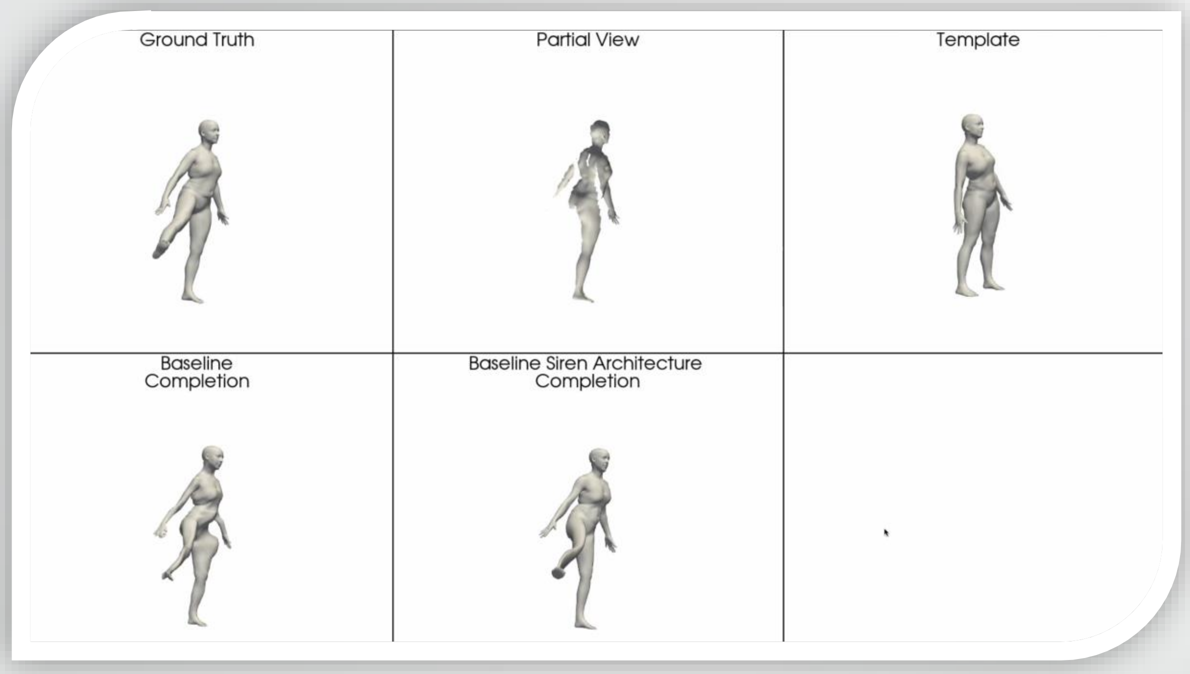
Deep Dive



GT Volume: 0.005207

	Baseline	Baseline +Siren Architecture
GT-comp Vertex MSE	0.000388	0.000281
Completion Volume	0.002719	0.003474
Volume Deformation [%]	47.7818	33.2898

Deep Dive



GT Volume: 0.06814

	Baseline	Baseline +Siren Architecture
GT-comp Vertex MSE	0.001797	0.001477
Completion Volume	0.058038	0.057155
Volume Deformation [%]	14.8224	16.1285

Deep Dive – High Resolution

Ground Truth | Baseline Completion | Baseline + Siren Completion

	Baseline	Baseline +Siren Architecture
GT-comp Vertex MSE	0.0009404	0.0004877

Additional Work

- We have attempted to improve the temporal smoothness by changing the architecture to LSTM
- As you will now see, this attempt succeeded far less then expected

LSTM – Long Short-Term Memory Network

An advanced RNN which can learn order dependence in sequence prediction problems

LSTMs have feedback connections, that allow processing sequences of data

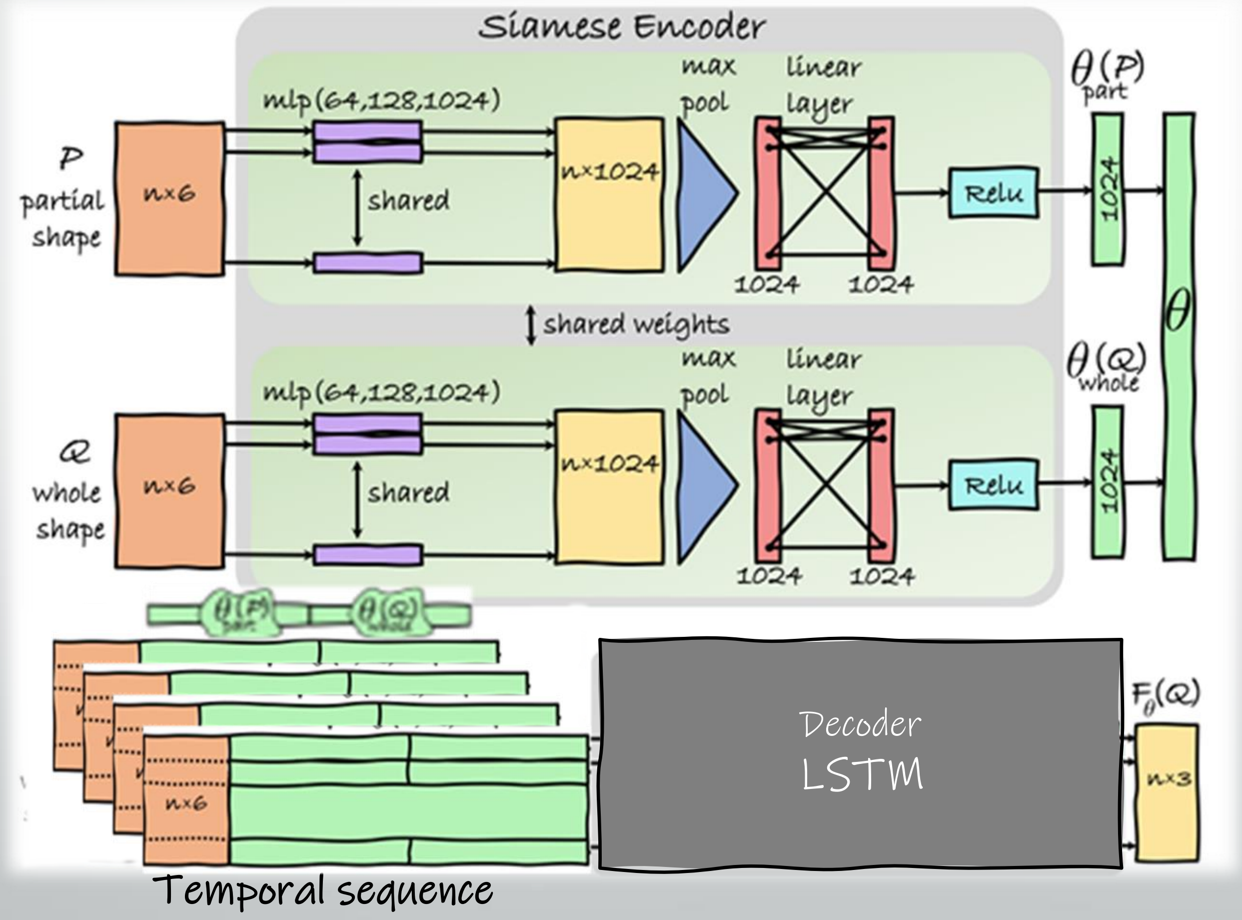
They manage to overcome the short-term memory problem of traditional RNNs

About Sequentiallity

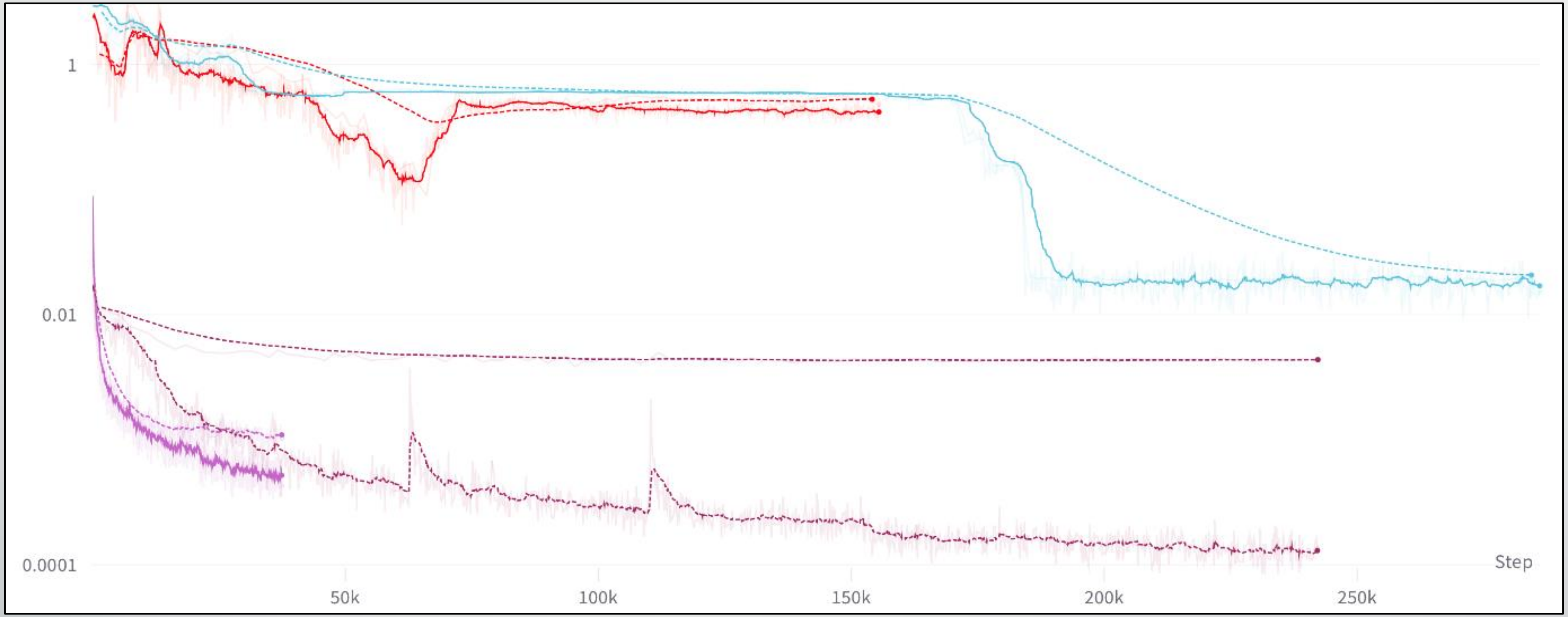
- Let's Define 2 Terms which we will use:
  - **Window Size** - How far in the past do we take each series
  - **Stride** - What size step do we take at each time step

LSTM - Network architectural Changes

- We propose a new network architecture, of the following form:



Vertex MSE



--- validation — train

Baseline

LSTM encoder window size 12, stride 8

LSTM encoder window size 12, stride 8, 0.1 coefficient for head volume loss

LSTM encoder window size 10, stride 8, 0.3 coefficient for head volume loss

Failures of the LSTM Architecture

Conclusions

- In this project, we have implemented various changes to both network architecture and loss function
- Sadly, the temporal "smoothness" architecture changes didn't succeed as expected
- However, we achieved significant improvements in the spatial domain, using various methods
  - Loss function terms and modifications
  - Architectural changes
- We performed experiments to validate our changes have the effect we desired, both in quantity and quality.

Benchmarking & Monitoring tools

experiment tracking, dataset versioning, and model management

Weights & Biases

Experiments

Reports

Artifacts

Tables

Sweeps

Challenges

- Resource allocation
- Online Servers logging issues
- Memory leaks
- Research Methodology
- Defining reliable monitoring methods

Questions?


mean squared error
Reviewer 2
Euclidean metric
L2 distance
log-likelihood with standard gaussian error
geodesic length on a Riemannian manifold with Euclidean topology