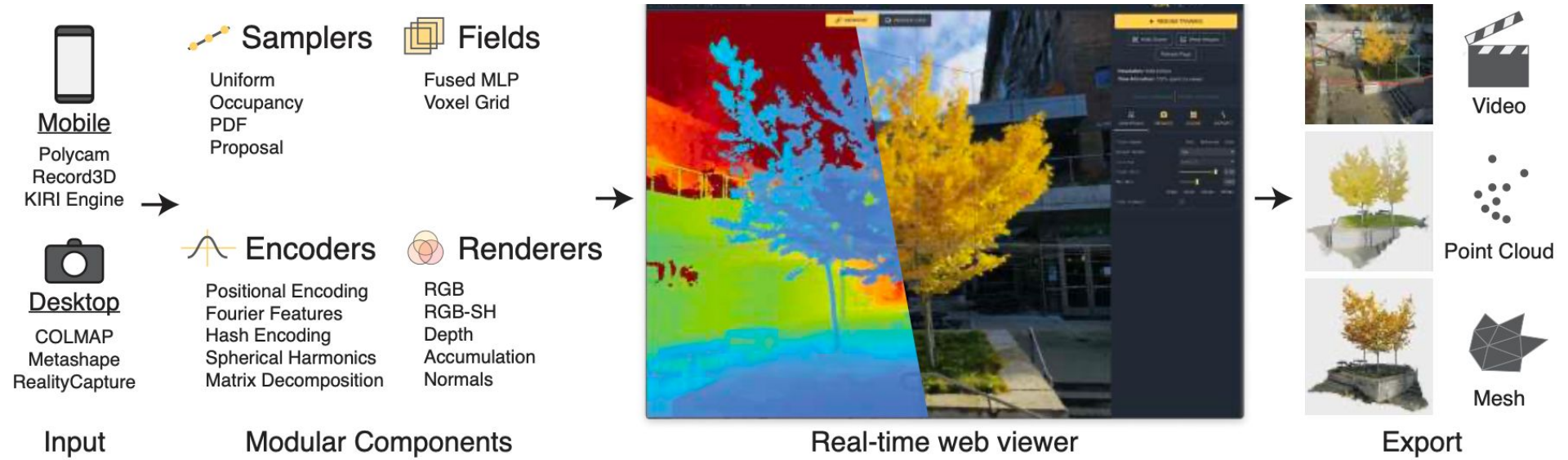
Nerf-Studio


Mobile
Polycam
Record3D
KIRI Engine
Desktop
COLMAP
Metashape
RealityCapture
Input
Samplers
Uniform
Occupancy
PDF
Proposal
Fields
Fused MLP
Voxel Grid
Encoders
Positional Encoding
Fourier Features
Hash Encoding
Spherical Harmonics
Matrix Decomposition
Renderers
RGB
RGB-SH
Depth
Accumulation
Normals
Modular Components
Real-time web viewer
Video
Point Cloud
Mesh
Export

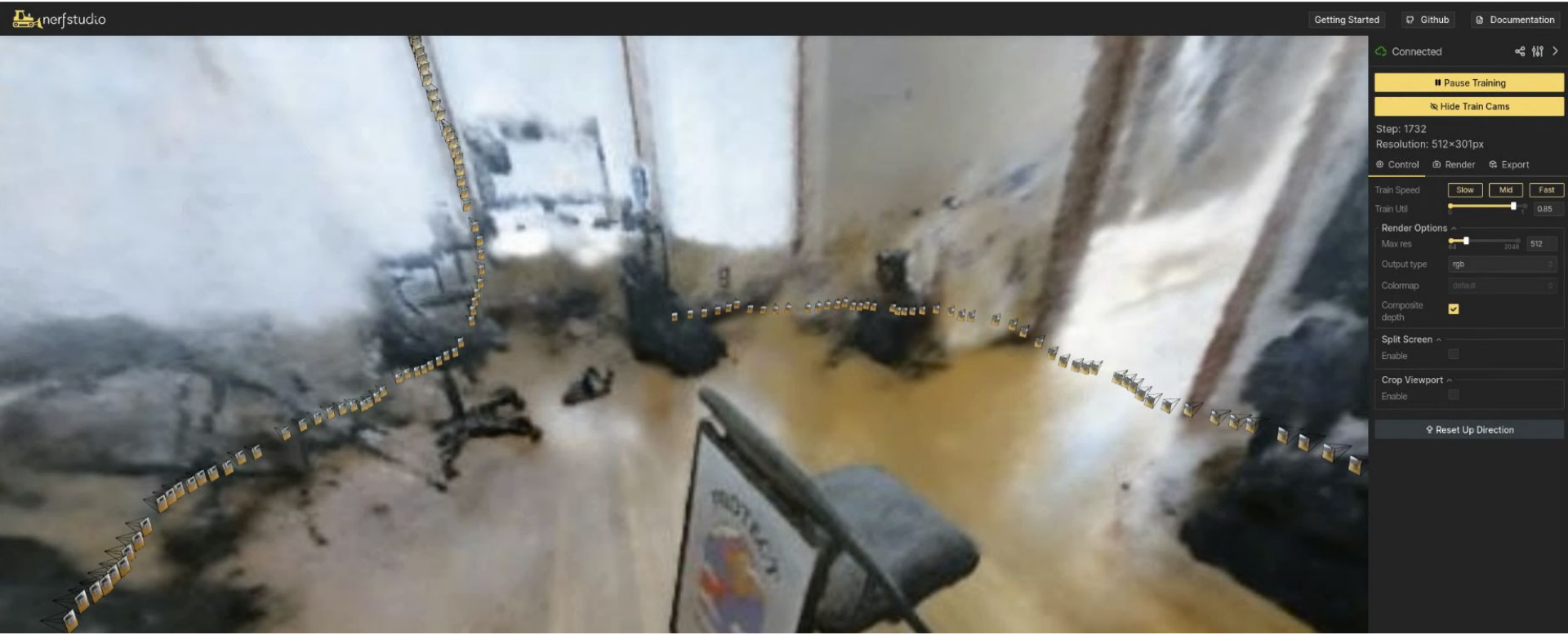
Overview

Methods:

- Nerfacto: Recommended method, integrates multiple methods into one.
- Instant-NGP: Instant Neural Graphics Primitives with a Multiresolution Hash Encoding
- NeRF: OG Neural Radiance Fields
- Mip-NeRF: A Multiscale Representation for Anti-Aliasing Neural Radiance Fields
- TensoRF: Tensorial Radiance Fields
- Splatfacto: Nerfstudio’s Gaussian Splatting implementation
- Many third party ones: https://docs.nerf.studio/#third-party-methods

Example 1: Poster

- Existing Dataset
- Default: AlexNet and nerfacto as default training method
- Training time on 1 T4 ~25 minutes
  - 4GB/16GB,60%utilization
- Render/Export scene from terminal/UI


nerfstudio
Getting Started
Github
Documentation
Connected
Pause Training
Hide Train Cams
Step: 1732
Resolution: 512×301px
Control
Render
Export
Train Speed
Slow
Mid
Fast
Train Util
0.85
Render Options
Max res
512
Output type
rgb
Colormap
default
Composite depth
Split Screen
Enable
Crop Viewport
Enable
Reset Up Direction


PROTECT
Your Mental
Health

Example 2: Lego Astronaut

- Polycam -> 32 images
- Convert:

```
ns-process-data polycam --data data.zip--output-dir astronaut
```

- Train:

```
ns-train nerfacto --data astronaut
```

  - Nerfacto method
- Train time 30min

Start

Example 3: Earth from Google Earth

- Video of google earth
- Roughly 2 hours of training
- Not really rotating around the globe