Vision-Based Camera Motion Tracking

sddec25-15:
Eric Wittrock, William Ernatt, Andrew Gooding, Isaac Kenyon

Client: Eric Wittrock
Faculty Advisor: Prof. Ashraf Gaffar

IOWA STATE UNIVERSITY

Background

- The VFX industry was $10.60 billion in 2024, expected to have grown to $11.19 billion in 2025.
- Camera Tracking is highly valuable to the VFX industry acting as a bound between live action footage and CGI.

The Problem

- VFX artists need to recreate camera movement for realistic effects
- A very tedious, time consuming process, especially for independent users.
- We want to streamline this process
- Save time and labor, while retaining track quality.

Blender Background

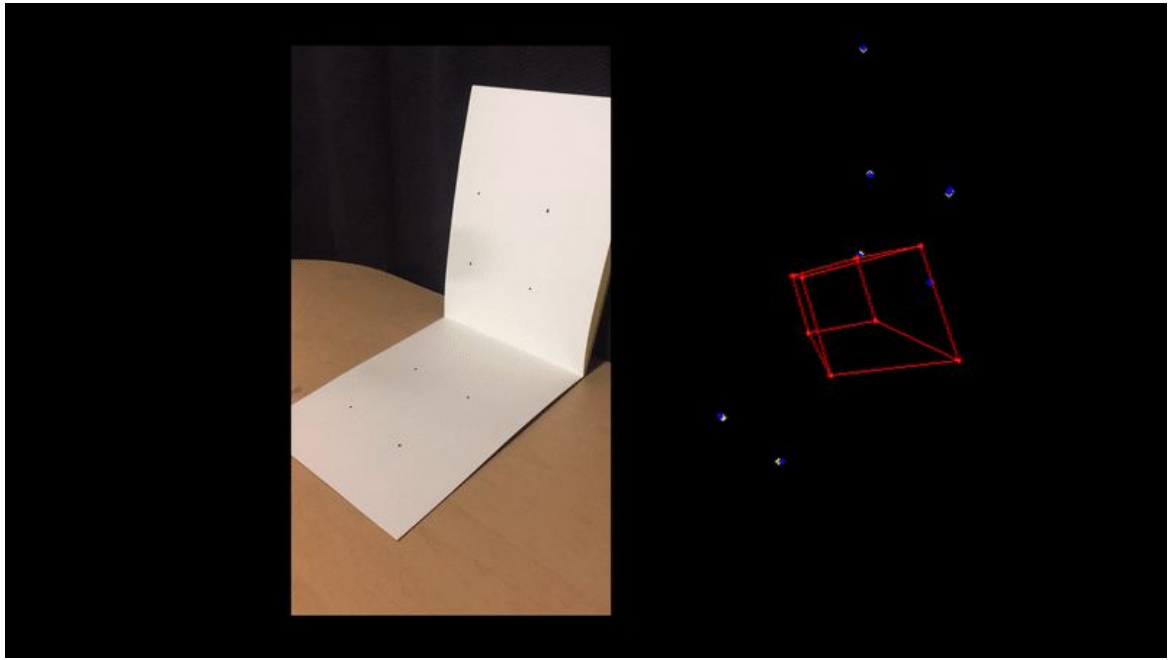
- Easy to use interface users will use to create with our product.

- With plugin scripting our product can be used in current and future development.
- Common among professional and amatuer VFX artist.

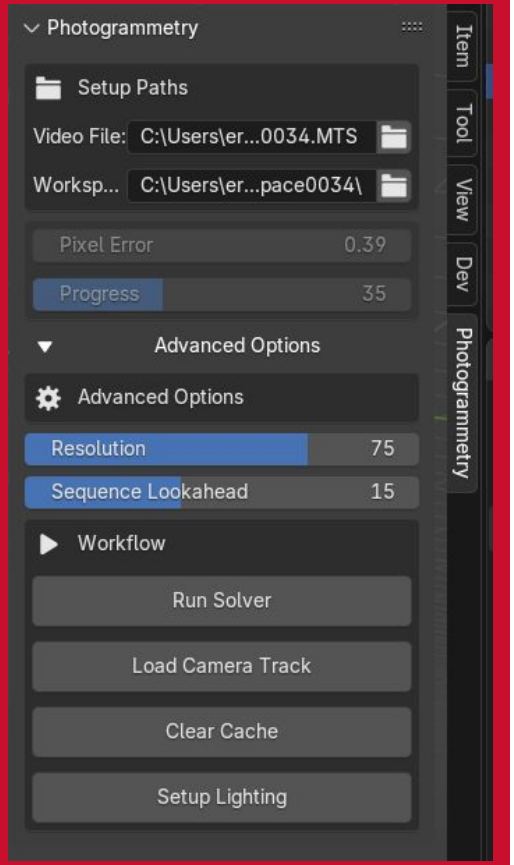
Our Solution

- A Blender extension for Camera Tracking
- Requires minimal effort for the user
- A couple clicks and you are done!
- Major reduction in time and effort for the same quality

Semester 1 results



- From-scratch Implementation
- Have since switched to COLMAP-based solution



Intended Users

- **The Professional User**
  - Quick testing of new ideas before final product
  - Benefits from quick integration, low processing time
- **The Independent User**
  - Our primary audience
  - Benefits from ease of use, high quality end result

Design Requirements

Our product should...

1. Solve for motion with no jitter

- Given physical camera footage, solve for motion and assign to virtual camera and blender.
- Jitter is unacceptable.

2. Require little to no user intervention

- After inputting the video and running the software, the rest of the process should be fully automated.

3. Have a Consistent UI Theme

- UI design of our software should integrate with existing Blender UI.

4. Work on any computer that runs Blender

- OS - Windows 8.1 (64-bit)
- CPU - 4 cores
- RAM - 8 GB
- GPU - 2 GB VRAM

Related Works

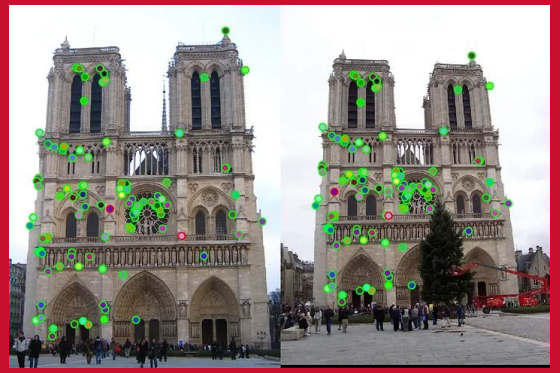
- **Blender Motion Tracking**
  - Free- included in Blender
  - Learning curve- less accessible for beginners
  - Tedious- can take up a lot of time!
- **SynthEyes**
  - Advanced suite of tools- users have almost complete control
  - Complex- can take weeks or months to learn
  - Expensive - $62/month

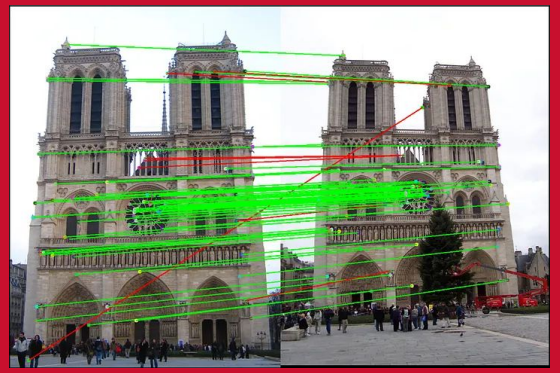
How it works- Image Extraction

- To prepare our video for the motion tracking pipeline, it must be broken into frames
- By default, each frame is rendered at half resolution
- Advanced feature- change frame resolution scale

Feature Extraction and Matching

- First step of COLMAP/GLOMAP pipeline
- Algorithm finds sparse features points in each image, assigns number
- Next, algorithm finds correlation between features across each image



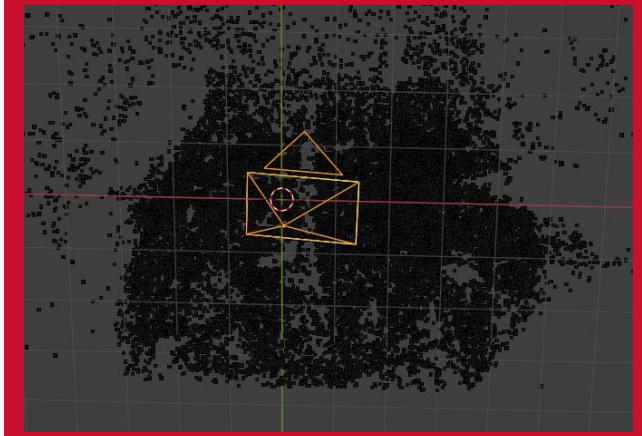


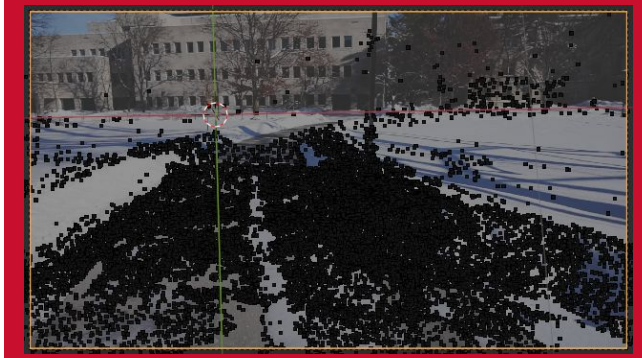
Structure-from-Motion Reconstruction

- SfM utilizes the output of feature matching to reconstruct a 3d structure.
- The projections of these features are used to calculate depth information and output a camera track and sparse point cloud.

Blender Point Cloud and Camera Track Import

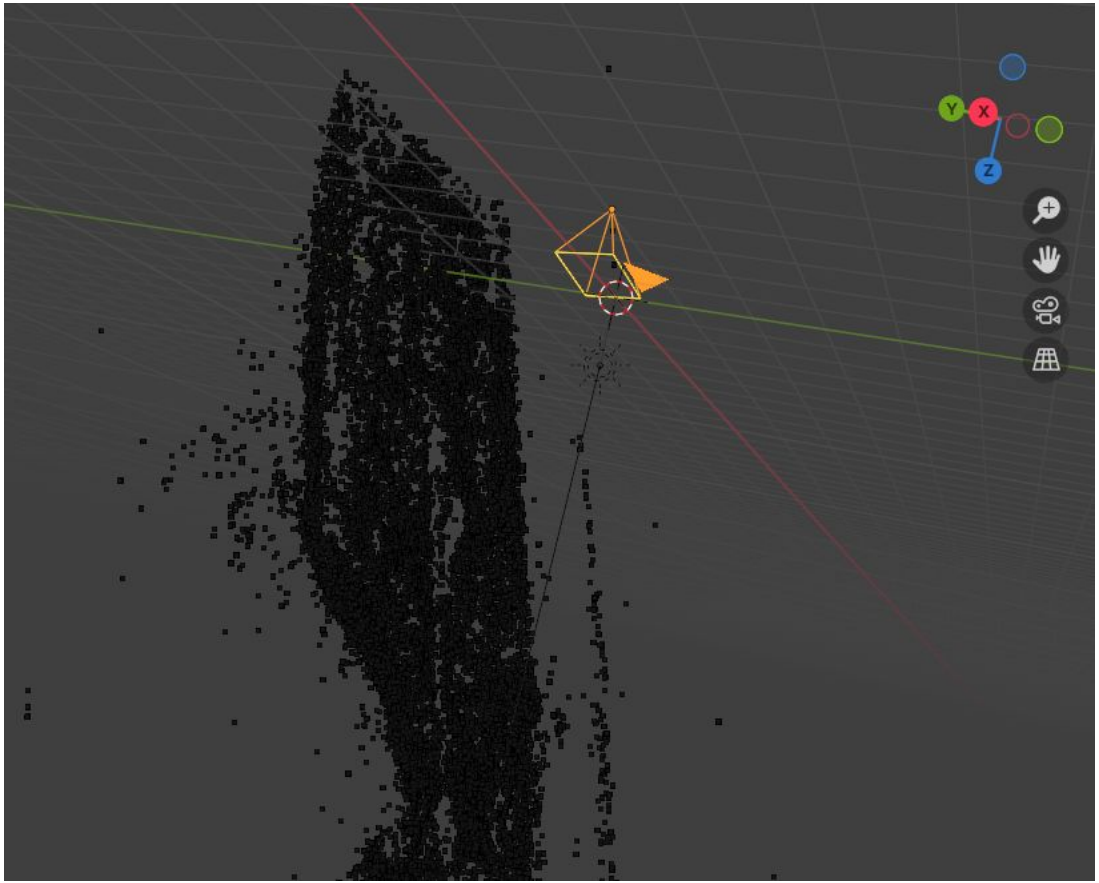
- Point cloud and camera track information is converted to usable data and imported into camera and point cloud object it Blender.
- Cache is stored, updated, and deleted automatically





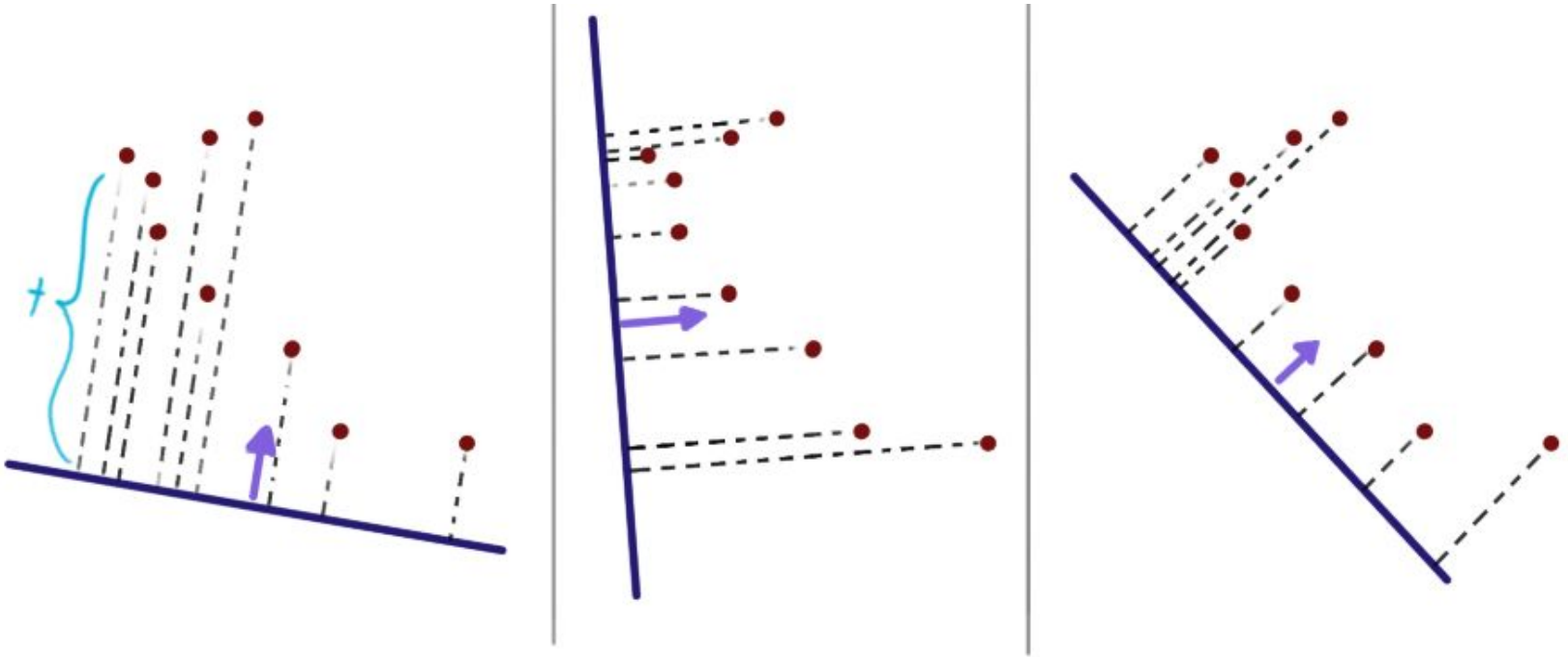
Floor Detection

- The points are loaded in at an arbitrary orientation
- Difficult to work with when not-axis aligned
- Detect floor plane when not all points are part of the floor structure

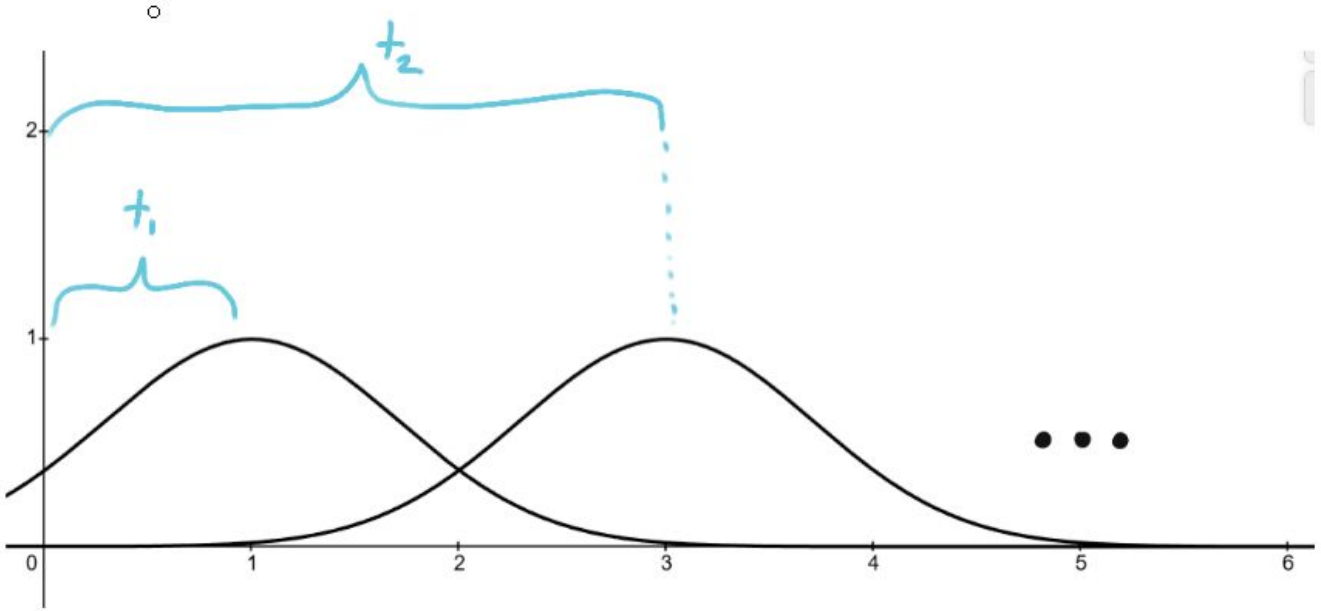


Floor Detection

- The floor usually is the largest plane of points
- Find a plane s.t. the the t-values for all points are close together

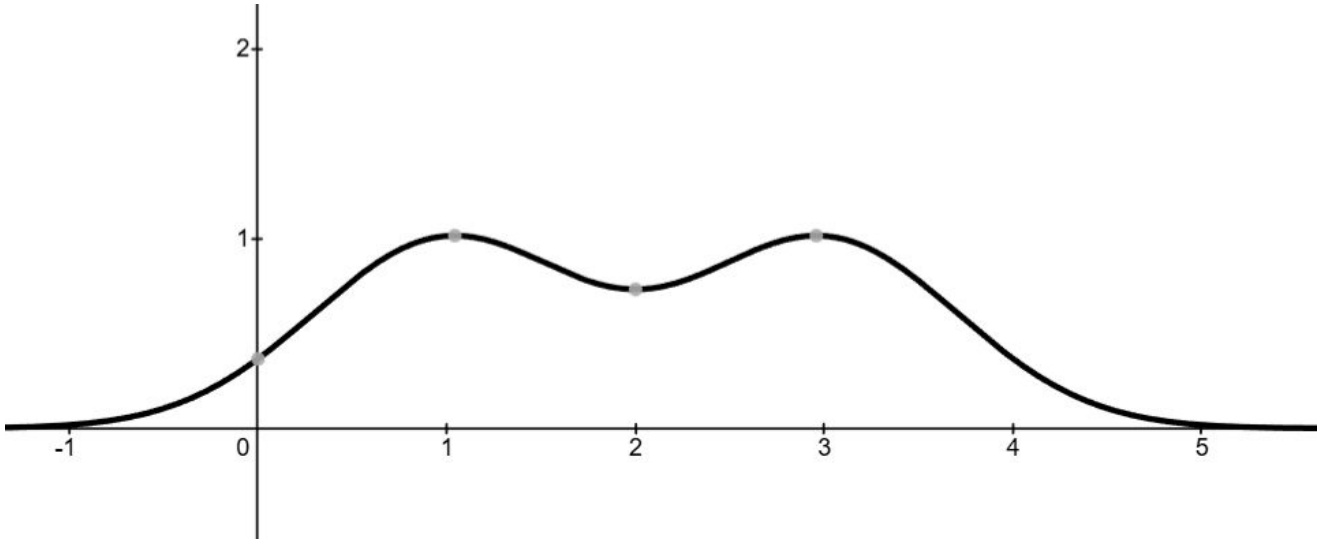


Floor Detection



Floor Detection

When we add the bell curves together...

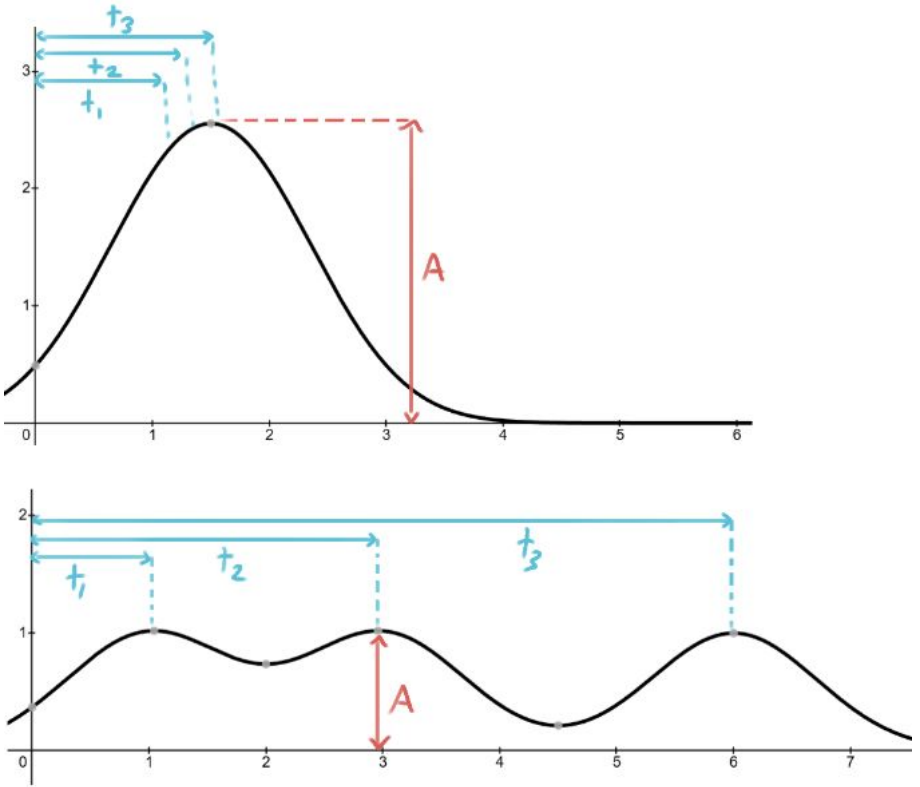


Floor Detection

The maximum height is greater when the t-values are close together

The maximum height is lesser when the t-values are far apart

Find a plane that maximizes A, the maximum height



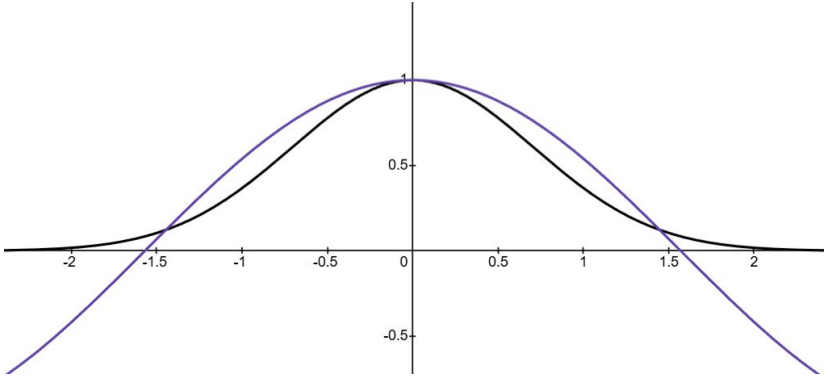
Floor Detection

This is difficult to solve

Find a plane orientation that maximizes

$$\max \sum_{i=1}^{N} e^{-(x-t_i)^2}$$

What if we substitute the bell curve for cosine?



Floor Detection

Adding two cosines of the same frequency gives you another cosine

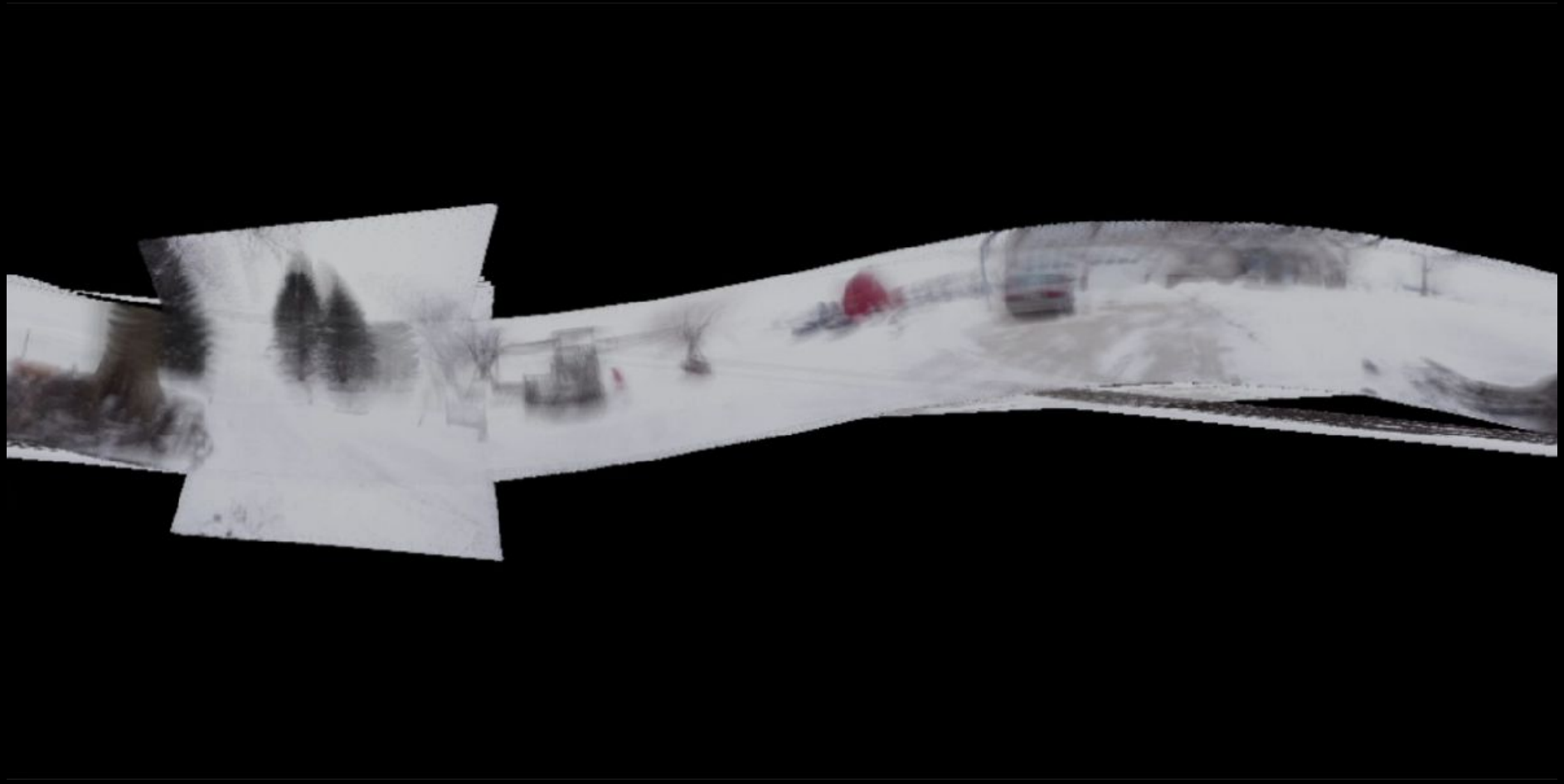
$$A = \max \sum_{i=1}^{N} cos(x - t_i) = \sqrt{(\sum cos(t_i))^2 + (\sum sin(t_i))^2}$$

Find a plane orientation that maximizes A (easy with optimization)

Lighting With Environment Maps

- 360 panoramic view
- Usually HDR (High Dynamic Range)

warped_overlay

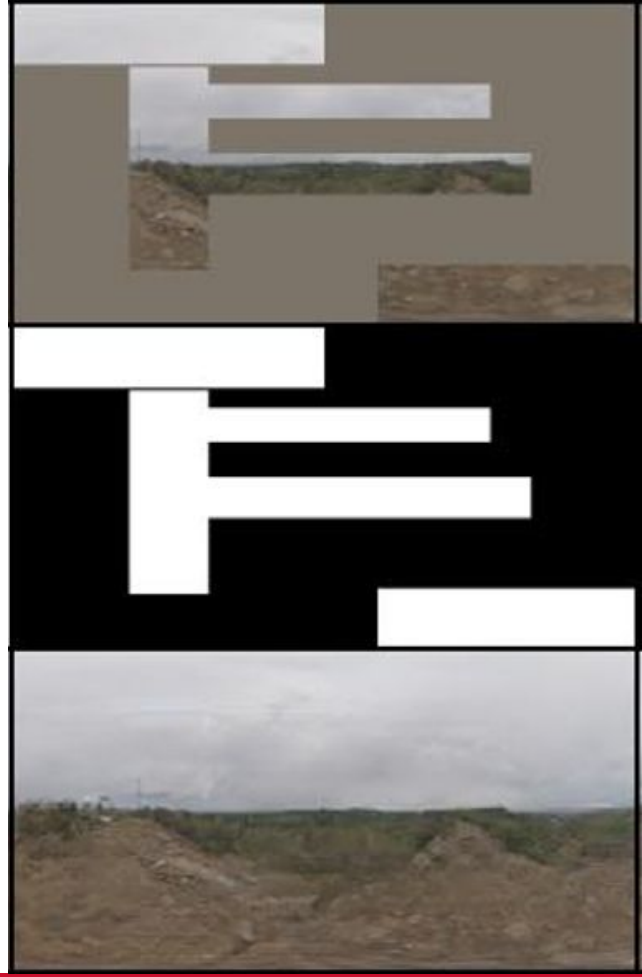


- ResNet 50 (26 million parameters)
- Fine tuned on an augmented dataset of 5000 environment images
- 8 hours on a single RTX 3070

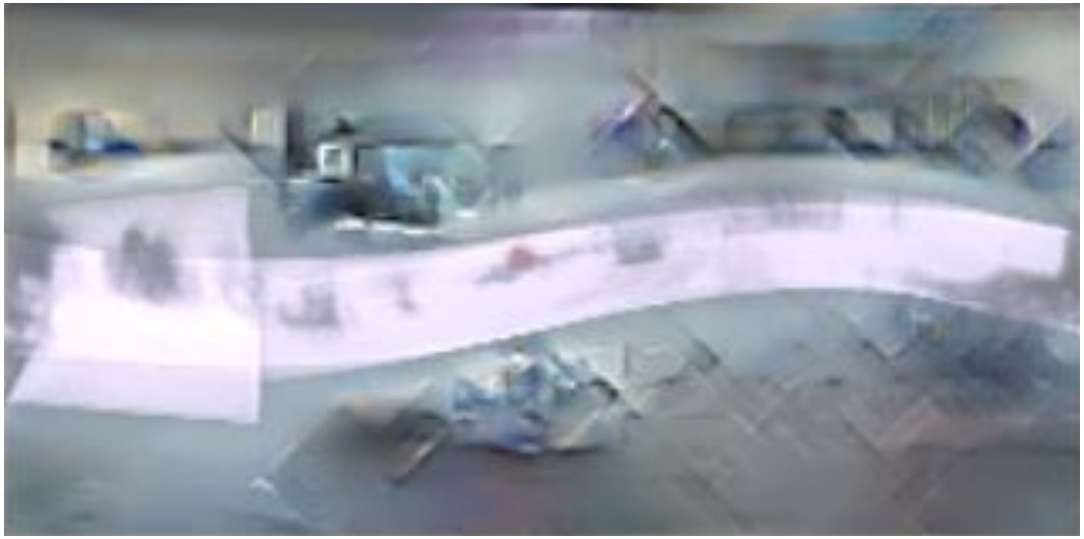
Input Image

Input Mask

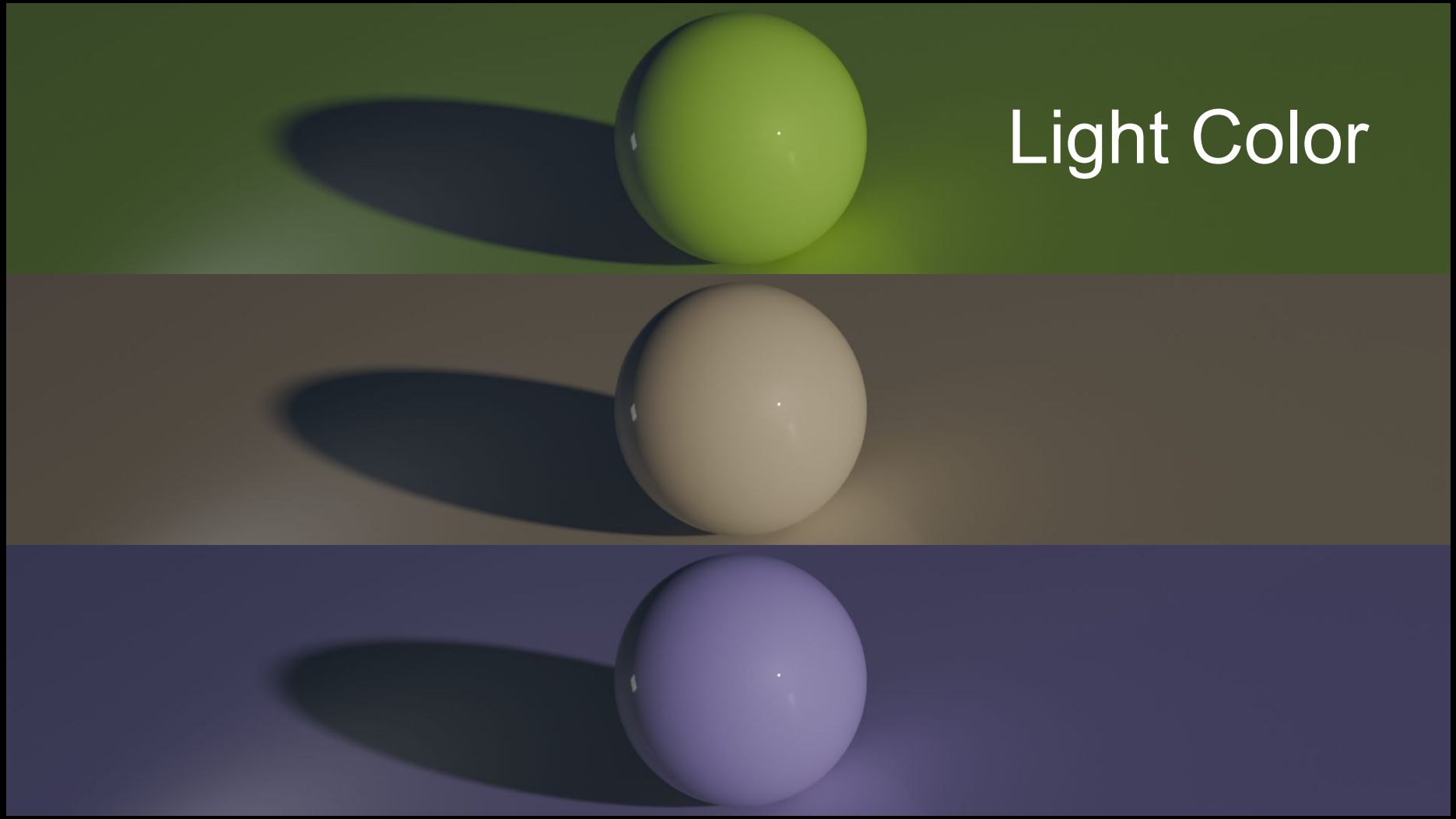
Output

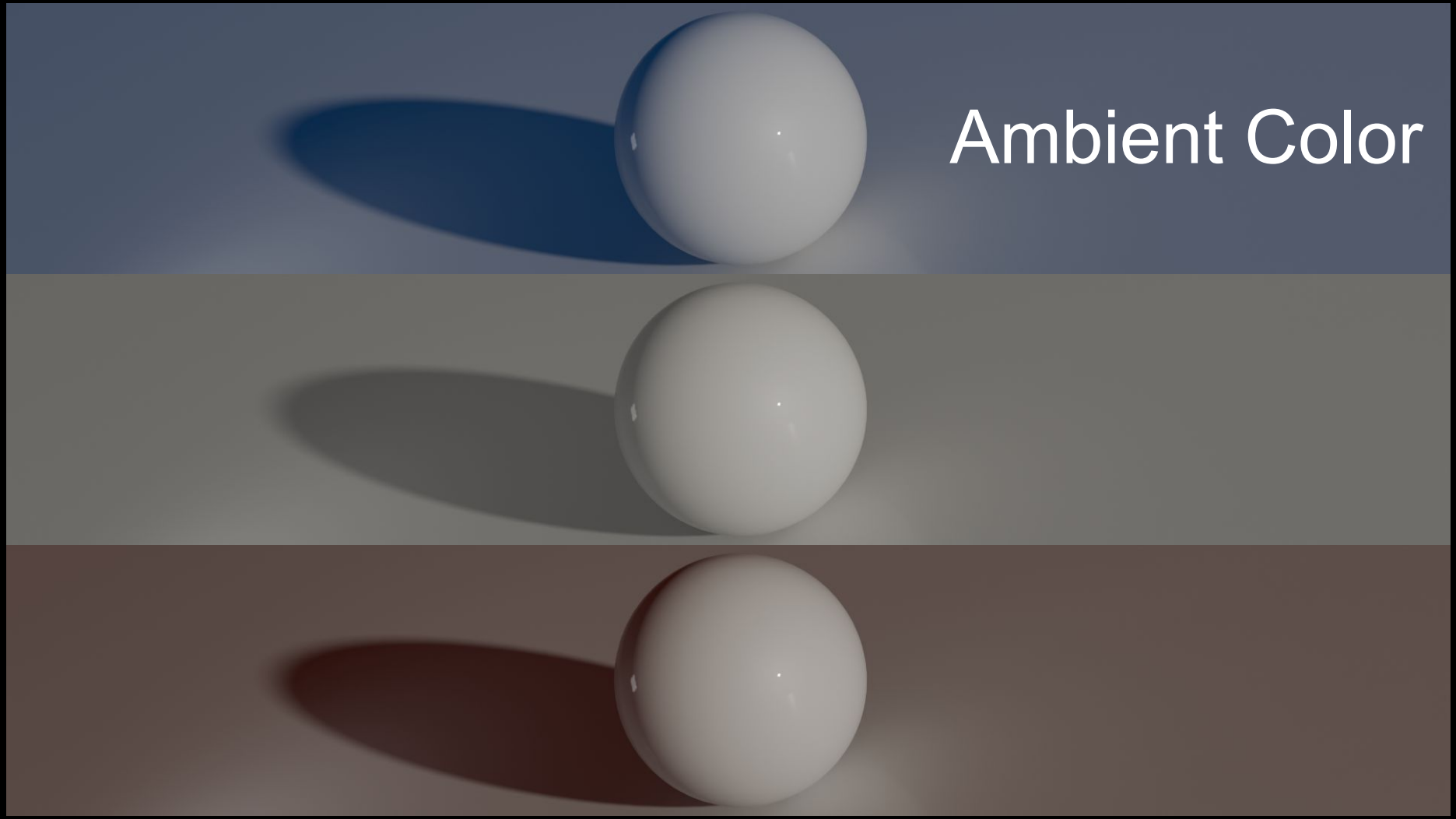


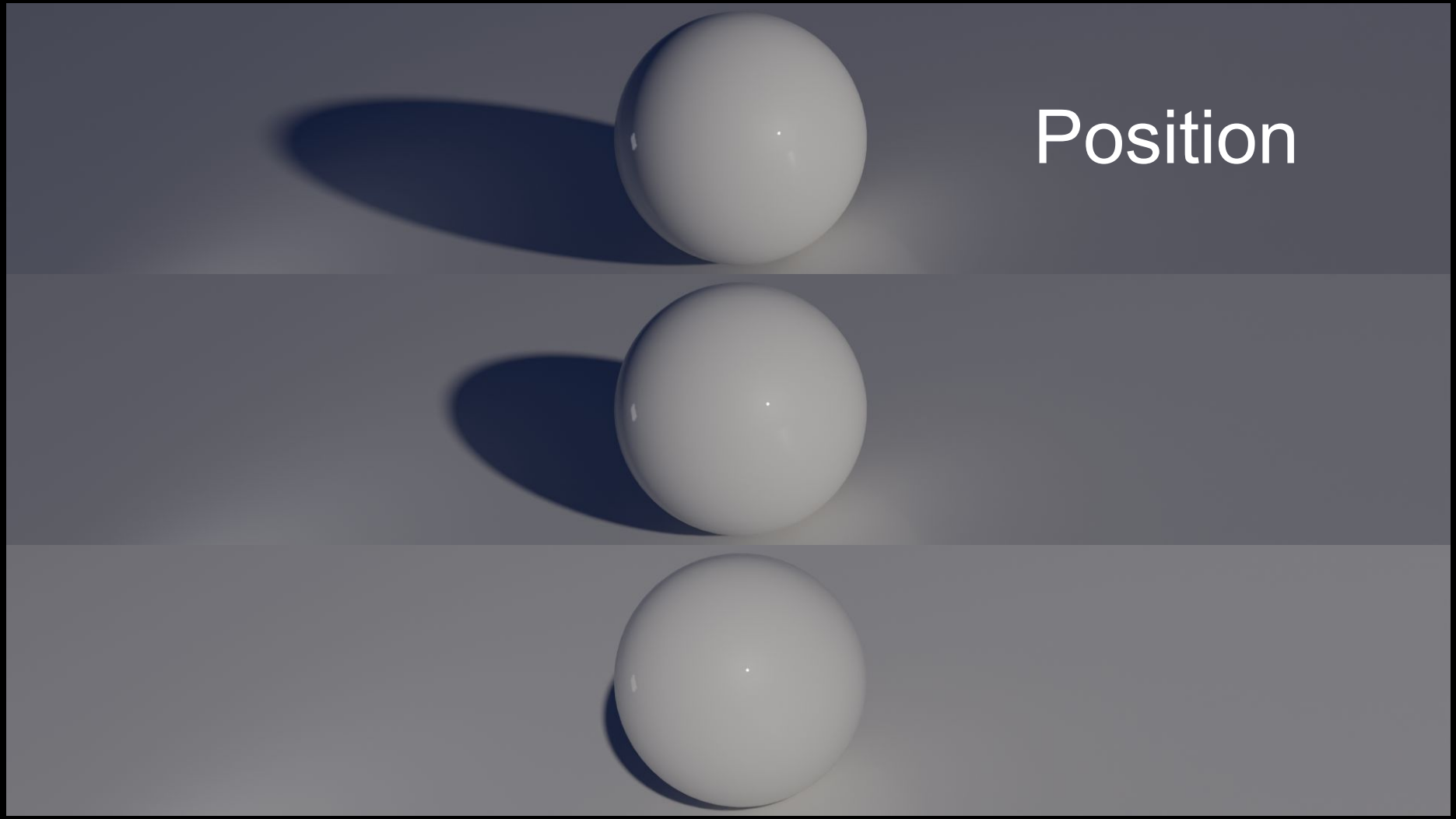
Results With Real Data

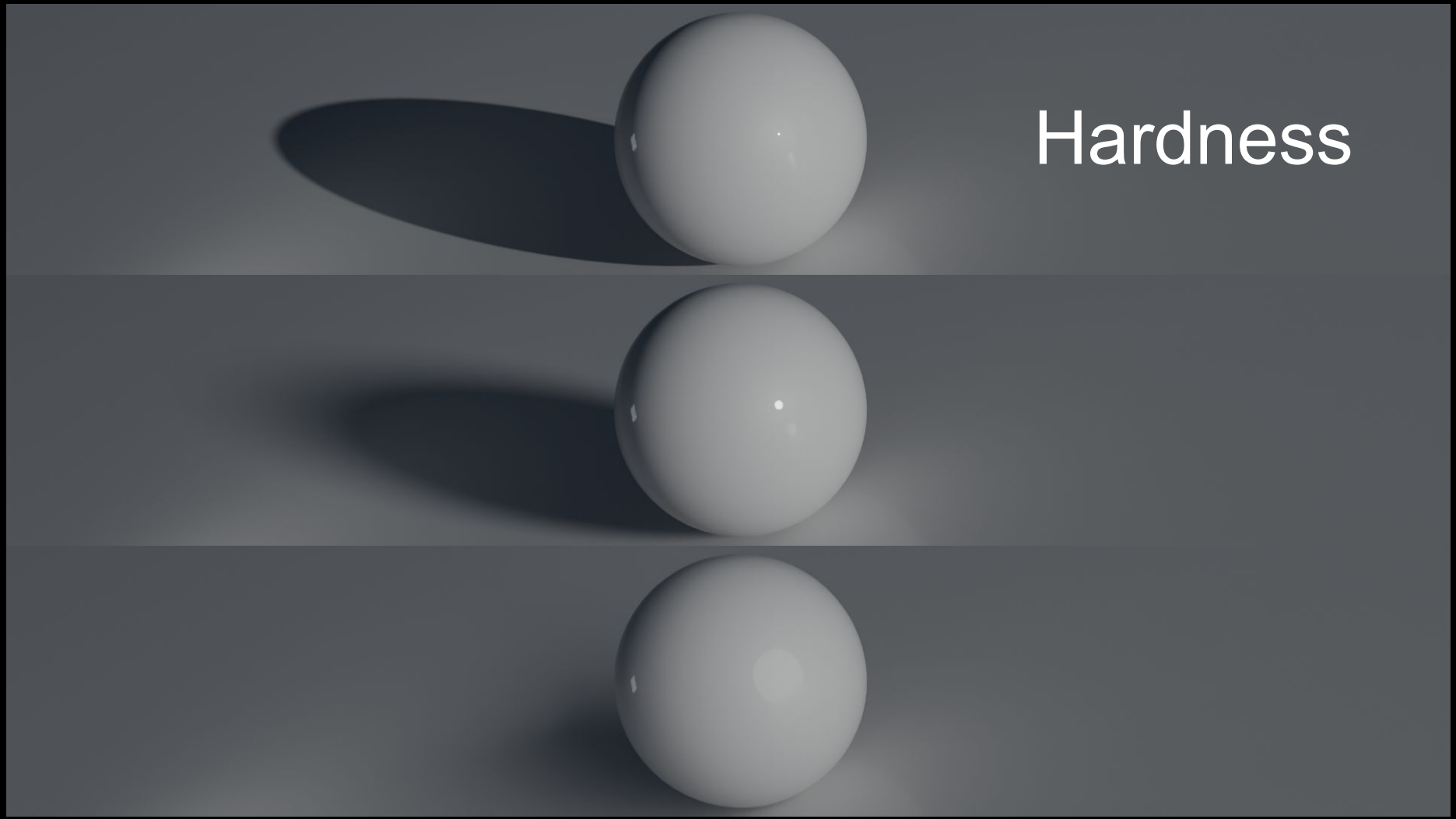


(Dismal)

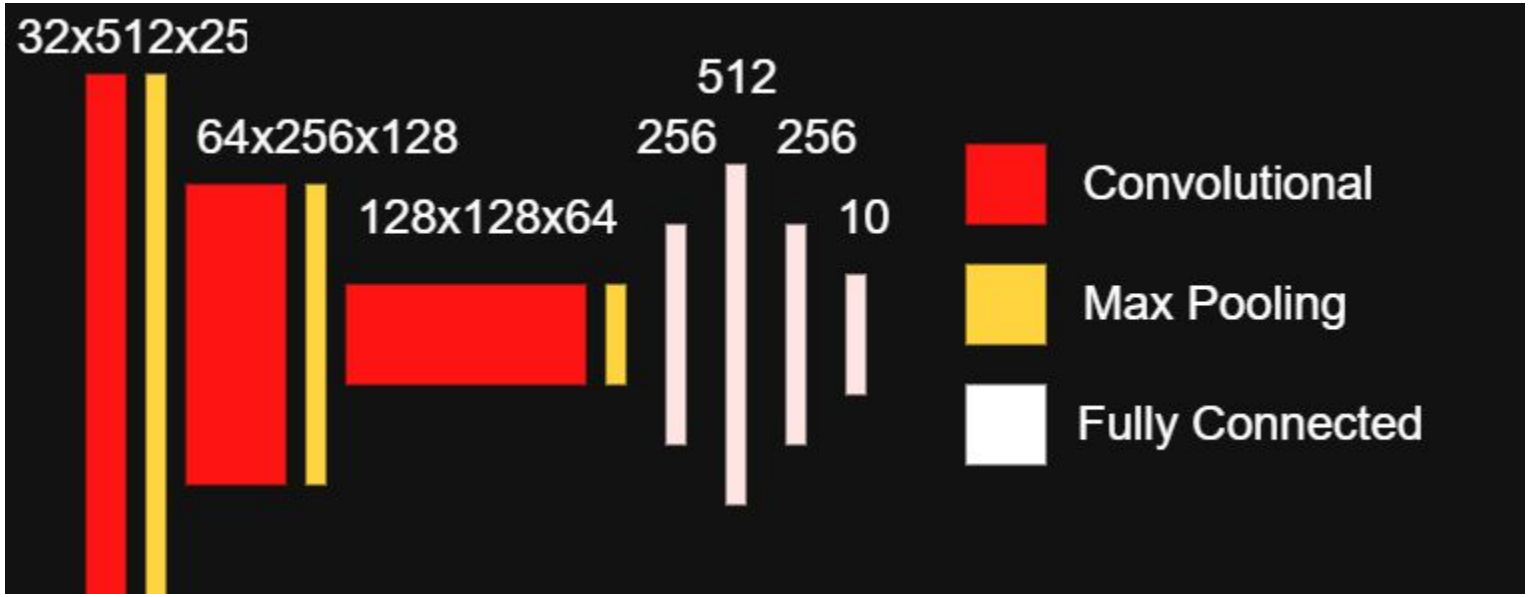

Light Color


Ambient Color


Position


Hardness

Convolutional Neural Network



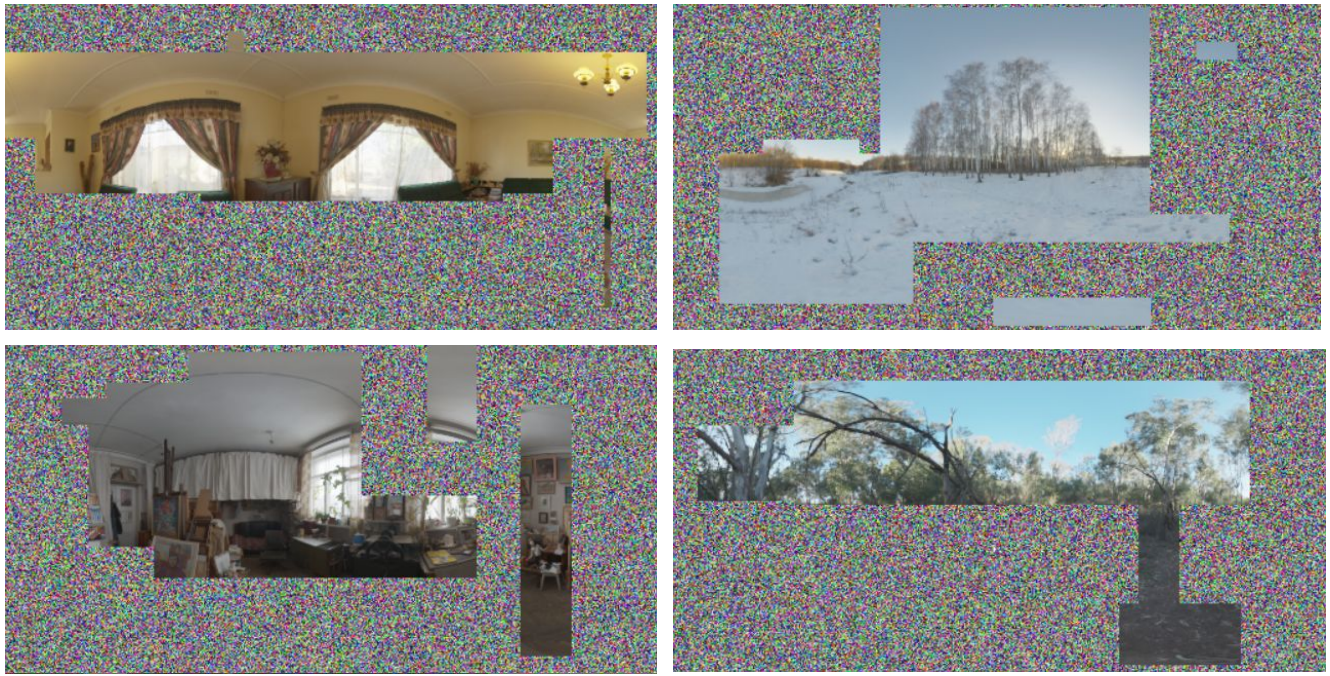
- 636,874 parameters (40x smaller)
- Trained 1 hr on augmented dataset size of 5000
- 10 outputs representing each lighting attribute

HDR images have more lighting information

- Analyze pixels in the 70th-98th percentile for ambient information
- Analyze pixels in the 98th percentile for direct lighting information

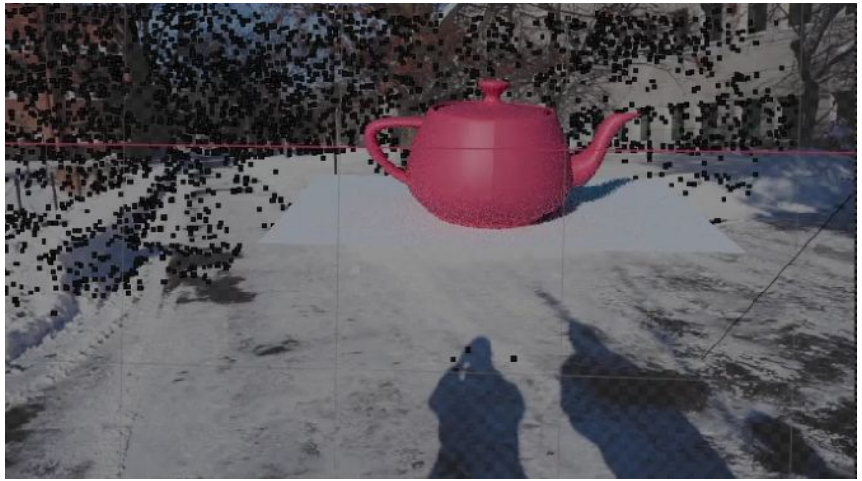
$$y \in \mathbb{R}^{10}$$

$$x \in$$



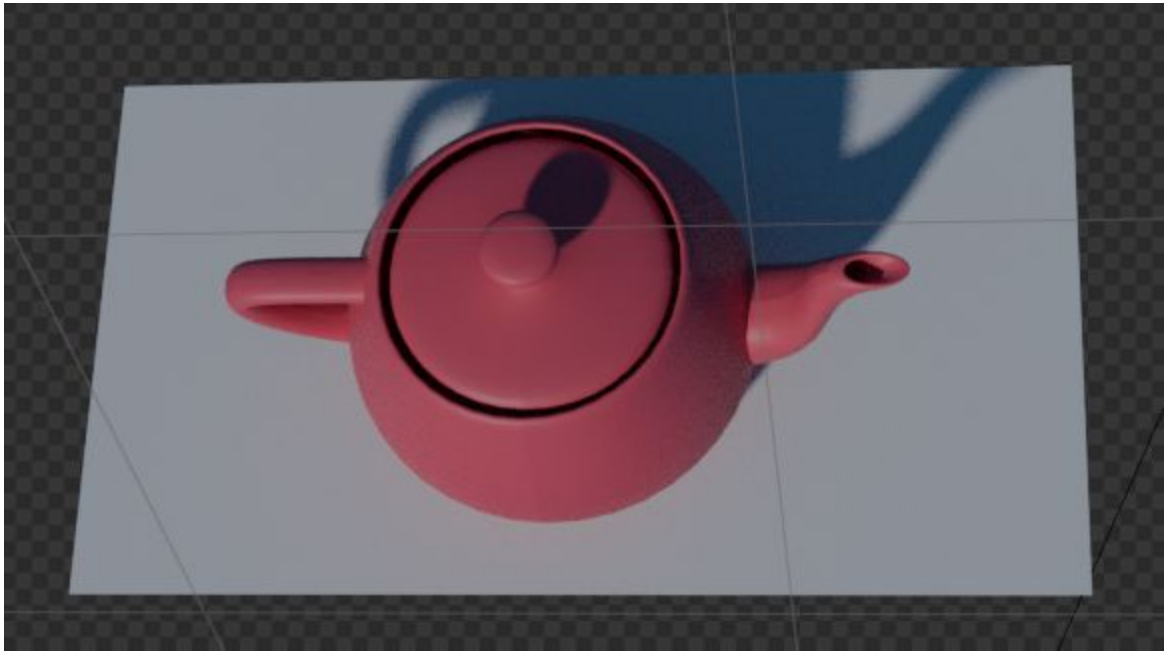
Results

Video Input



Lighting Output

Results



Results

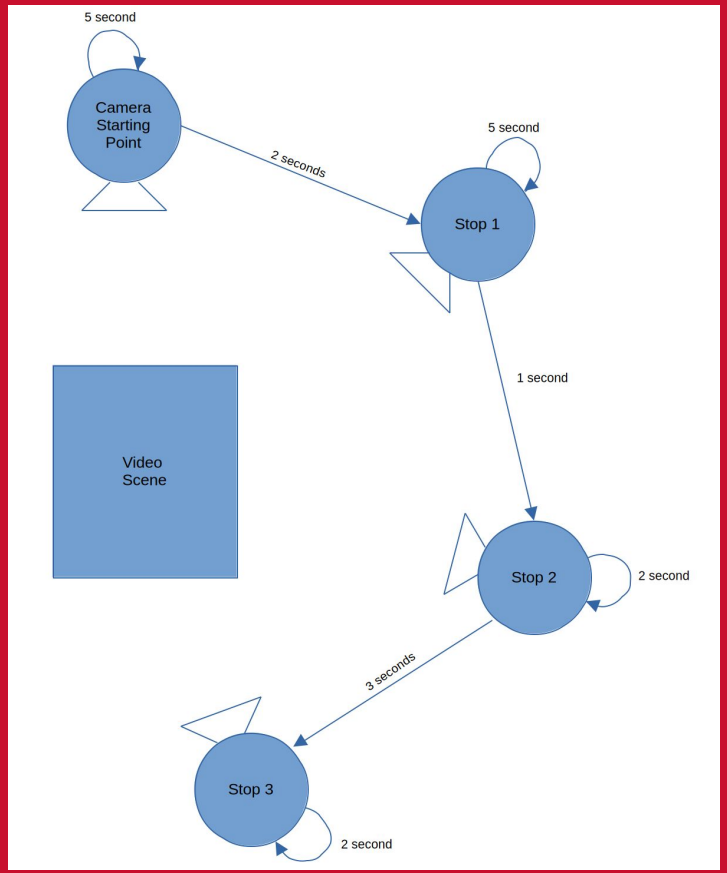
Testing

Values to Test

- Speed
  - Manual Camera Tracking in blender vs Plugin Camera Tracking
  - Length of Video affect on Plugin Performance
- Accuracy
  - Pixel Error (mean distance between 2d tracker points and the camera projection of their estimated 3d position)
- Efficiency
  - Speed and accuracy

How to Quantify?

- ➔ Does video size scale linearly?
  - Can a video size help estimate length of time needed to compute.
- ➔ Does video size affect pixel error?
  - Improve or make worse.
- ➔ How to quantify video content?
  - Shaking camera.
  - Number of objects.
  - Moving objects.
  - Contrasting areas.

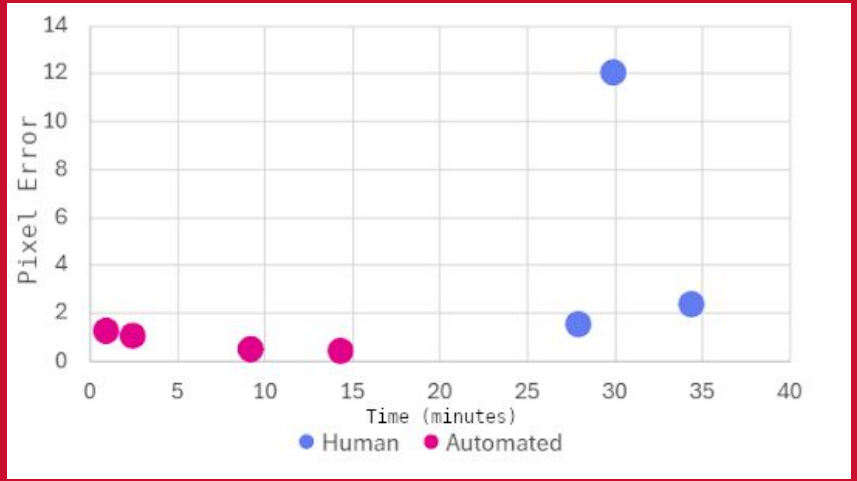


Duration Example

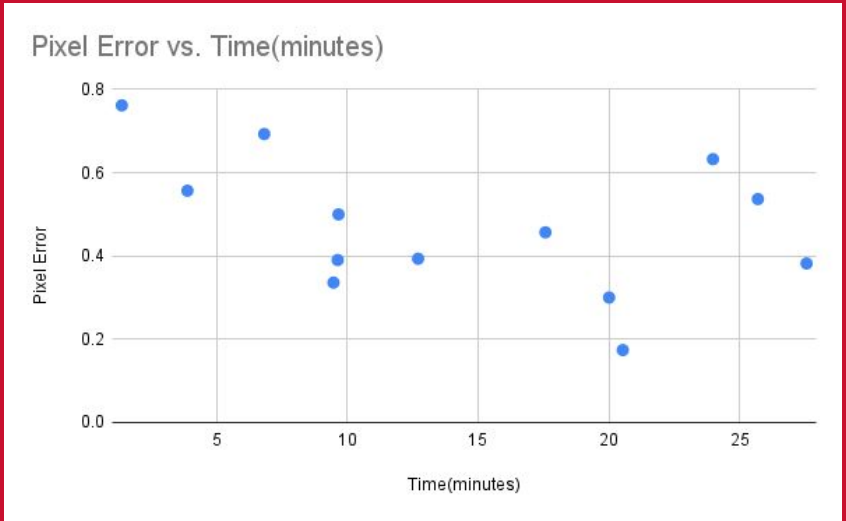
- The video takes 20 seconds
  - No movement in certain intervals
  - Movement between spots.
- 70% Static
  - Pipeline attempts to merge static frames.
- 30% Movement
  - Majority of the computation is done.

Speed - Human vs. Machine

- Did a manual tracking inside blender.
  - Non-professional user of blender.
  - Frame-by-frame changing values.
- Compared the Blender Plugin.
  - Same videos.
- Hard to quantify a user.



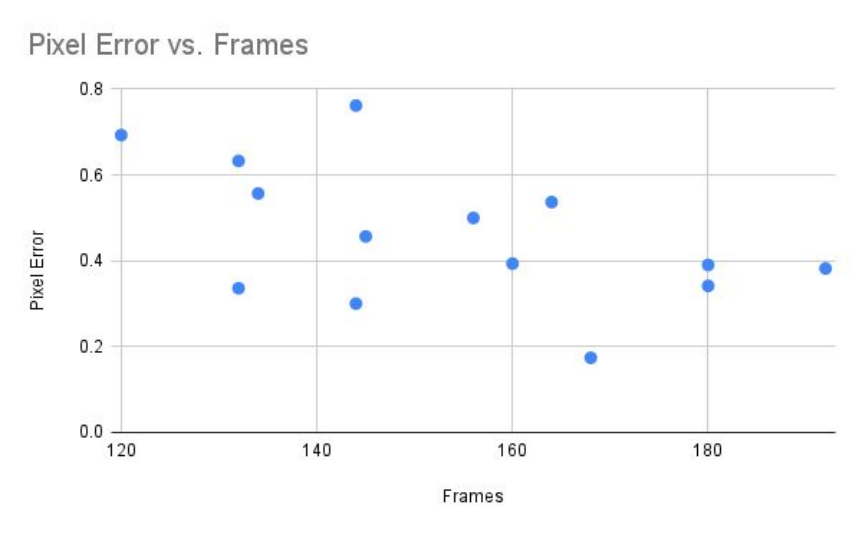
Speed and Accuracy



- Collected similar starting parameters.
  - Truncated to 120 frames
  - Same quality
  - Same file size
  - 50% resolution
- Variety of scenes and movement types.
- The 95% slope confidence interval contains zero.

Frames and Accuracy

- Collected different starting parameters.
  - Variety number of frames
  - Same quality
  - Different file size
  - 50% resolution
- Variety of scenes and movement types.
- The 95% slope confidence interval also contains zero.



Accuracy Tests

- Straight walk forward.
- No large moving objects in frame.

15.2MB	File Size
34.864 minutes	Total Time
0.341496	Pixel Error
180	Frames

Accuracy Tests

- Straight walking forward.
- Two objects moving in sequence with the camera movement.

11MB	File Size
26.381 minutes	Total Time
0.632551	Pixel Error
132	Frames

Accuracy Tests

- Panning video.
- More wobble than other videos.
- Minimal parallax

15.9MB	File Size
44.098 minutes	Total Time
0.382053	Pixel Error
192	Frames

Accuracy Tests

- Orbit shot
- Objects stay in frame
- Contrasting objects but featureless surfaces

12.1MB	File Size
1.658 minutes	Total Time
0.761707	Pixel Error
144	Frames

Results

- Scene complexity matters more than duration or file size.
- Videos with more features to match have a lower pixel error but take longer to compute.
- More frames can help reduce pixel error, but as stated before, the quality, motion, and feature distribution of those frames matter more than the count.

Conclusion

Lessons Learned

- Used in other tech fields like robotics.
- With COLMAP and GLOMAP pipelines being used it is important to be able to pivot.
- Communication is key!

Future Work

- More extensive testing for super small and large videos(Very resource intensive).
- Finish MacOS integration problems.
- Code has been open sourced and uploaded to the official blender extensions site for approval.

Team

Eric Wittrock

ejw3@iastate.edu

Andrew Gooding

drewgood@iastate.edu

Will Ernatt

wernatt@iastate.edu

Isaac Kenyon

ilpenzol@iastate.edu

Thank You

IOWA STATE UNIVERSITY