Microsoft 3D Reconstruction Supplement

v1 2025.05.09

Outline

- Schedule
- Data
- Evaluation
- Code Submission
- Report Submission
- Grading

Schedule

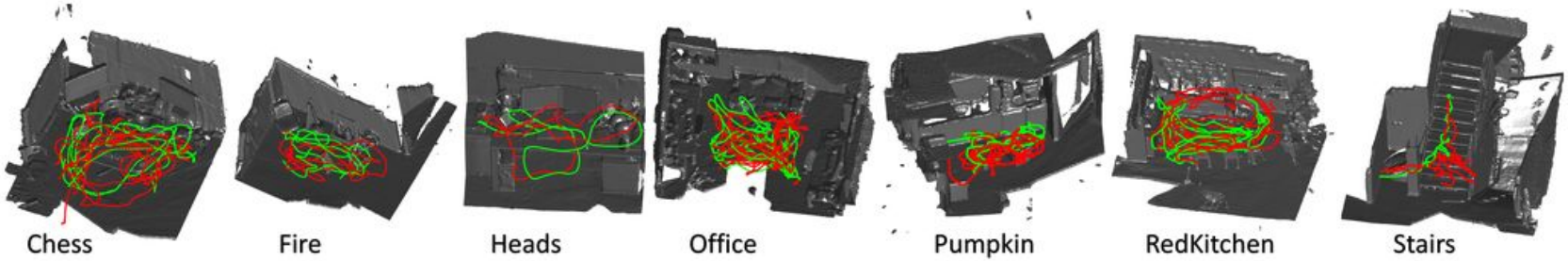
- Evaluation server open
  - 2025/05/11 00:00
- Evaluation server close
  - 2025/06/01 23:59
- Oral presentation
  - 2025/06/06 14:20~15:50 (Tentative)
- Code submission
  - 2025/06/8 23:59
- Report submission
  - 2025/06/8 23:59

Outline

- Schedule
- Data
- Evaluation
- Code Submission
- Report Submission
- Grading

Data

- 7Scenes contains video sequences of 7 indoor scenes.
- Please download the our preprocessed version. Link
- Each sequence contains:
  - RGB Image: XXX.color.png
  - Pose: XXX.pose.txt
  - Depth: XXX.depth.png
  - Depth Projection: frame-XXXXXX.depth.proj.png



Data

- RGB Image:
  - 24-bit RGB image, 640 x 480
- Pose:
  - 4 by 4 matrix (T) which represents the camera-to-world pose
  - P_world = T · P_camera
  - Pose is usually not available
- Depth:
  - 640 x 480 single channel png file
  - Each pixel is a 16-bit integer depth in millimeters
  - Invalid depth is set to 65535
- Depth Projection
  - Calibrate the depth information to the view of RGB camera
- Intrinsic:
  - fx = 525, fy = 525 , cx = 320, cy = 240

Data

- You are encouraged to
  - Use any 3D reconstruction method or pretrained model
  - Train from scratch or fine-tune on external datasets
- You are not allowed to
  - Directly using pretrained models already trained on the 7-Scenes dataset
  - Fine-tuning any models on the 7-Scenes testing set
- Here we provide you some reference work
  - DUSt3R: Geometric 3D Vision Made Easy
  - Fast3R: Towards 3D Reconstruction of 1000+ Images in One Forward Pass

Data

- There are two kinds of test sequences
  - Dense test sequences: 500 to 1000 frames
  - Bonus sparse sequences: 10 frames only
- We would build the ground truth data with this file
  - The unit of ground truth data is meter
  - But be careful the unit in raw depth file is millimeter
  - We would use kf_every=20 and voxel_grid_size=7.5e-3 to build the ground truth point cloud
- How to use the above data?
  - Training/Fine-tuning stage: You may use rgb, depth, or pose
  - Inference/Testing stage: You can only use rgb and depth information!!
- Calibrate the test sequence results using the pose of the first frame
  - Assume you have 3D coordinates P_c0 under the first frame's camera view
  - Transform them to world coordinates using $P_w = T0 \cdot P_c0$

Outline

- Schedule
- Data
- Evaluation
- Code Submission
- Report Submission
- Grading

Evaluation

- Metric1 - Accuracy:
  - For each predicted point…
    - Find its nearest neighbor in the ground-truth point cloud
    - Compute the Euclidean distance between the two points
  - Take the median of these distances as the Acc score.

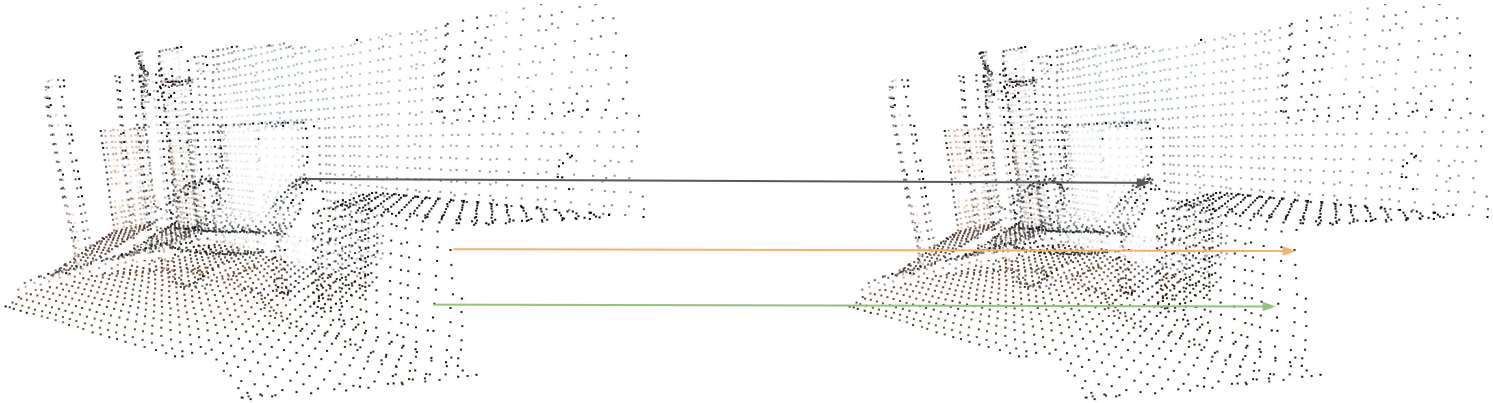
$P = \{\mathbf{p}_i\}_{i=1}^{N_P}$: Predicted Point Cloud

$G = \{\mathbf{g}_j\}_{j=1}^{N_G}$: Ground-Truth Point Cloud

$\|\cdot\|_2$: Euclidean Distance

$\mathrm{med}\{\cdot\}$: Median number of a set

$$\mathrm{Acc}(P, G) = \mathrm{med}_{\mathbf{p}_i \in P} \left[\min_{\mathbf{g}_j \in G} \|\mathbf{p}_i - \mathbf{g}_j\|_2 \right]$$



Predicted Point Cloud

Ground Truth Point Cloud

Evaluation

- Metric2 - Completeness:
  - For each ground-truth point
    - Find its nearest neighbor in the predicted point cloud
    - Compute the Euclidean distance between the two points
  - Take the median of these distances as the Comp score.

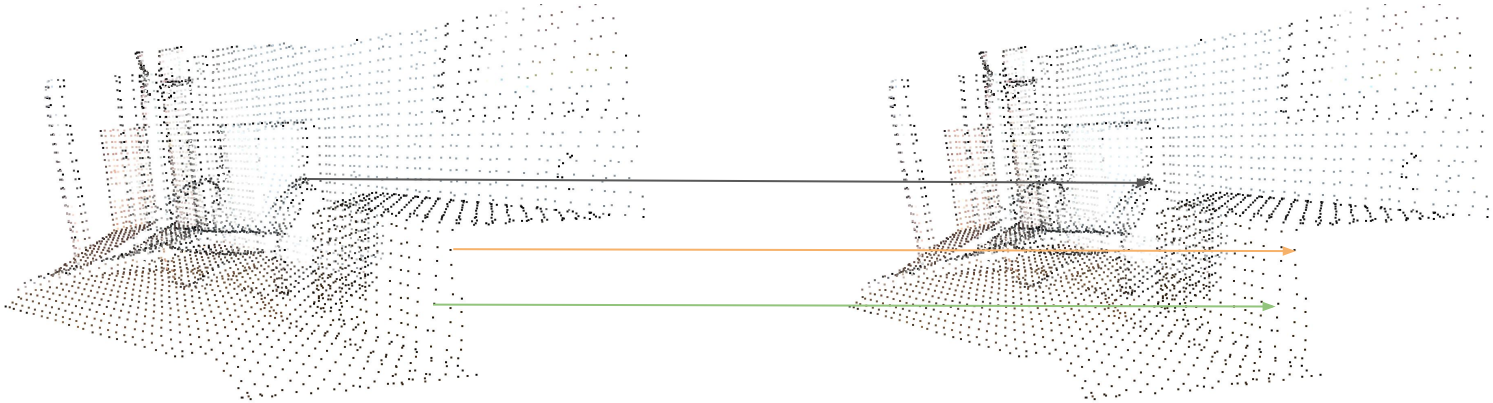
$P = \{\mathbf{p}_i\}_{i=1}^{N_P}$: Predicted Point Cloud

$G = \{\mathbf{g}_j\}_{j=1}^{N_G}$: Ground-Truth Point Cloud

$\|\cdot\|_2$: Euclidean Distance

$\mathrm{med}\{\cdot\}$: Median number of a set

$$\mathrm{Comp}(P, G) = \mathrm{med}_{\mathbf{g}_j \in G} \left[\min_{\mathbf{p}_i \in P} \|\mathbf{g}_j - \mathbf{p}_i\|_2 \right]$$



Ground Truth Point Cloud

Predicted Point Cloud

Evaluation

- Our project would be held on Codabench
- Competition Link
- Registration Flow
  - Sign up an account on Codabench
  - Email TA (jackmafan@media.ee.ntu.edu.tw) with the account name of your team
  - We would only approve registration request once you email us
- The competition is available from 05/11 0:00 to 06/01 23:59

Evaluation

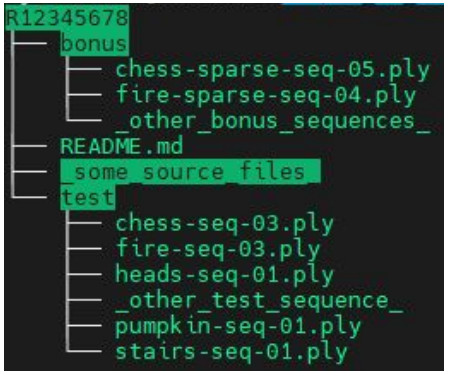
- Submit the reconstruction results of dense sequences to Codabench server
  - Store the result as {scene id}-{sequence id}.ply
  - Put all .ply files under a folder named “test”
  - Then zip the folder into “test.zip” and submit it to the codabench server
  - Please visit the competition link for more detailed imformation
- For each metric (accuracy and completeness)
  - First compute the average across all dense test sequences within each scene.
  - Then take the mean of per-scene scores for final score
- No need to submit sparse(bonus) sequences to Codabench
  - Pleas refers to p.15 of this slide

Code Submission

- Schedule
- Data
- Evaluation
- Code Submission
- Report Submission
- Grading

Code Submission

- R12345678/
  - README file
  - Source code (which can reproduce the result on the leaderboard)
  - Reconstruction result of test sequences
    - Name your result with the specified format
    - Put them under “test” folder
  - Reconstruction result of bonus sequences
    - Name your result with the specified format
    - Put them under “bonus” folder
  - Brief description of models and your method(pdf file; content is not restricted; serve just as a reference for the selection of teams for oral presentations)



- Compress all the files in a zip file named StudentID.zip (e.g. R12345678.zip)
  - Upon extraction, only one directory named R12345678 should be generated

Code Submission

- Only the team leader need to upload the code to NTU COOL
- Clearly describe how to set up the environment in the README file
  - Provide steps by steps instruction (ideally a bash script) to build the environment
  - So that TA can reproduce the result
- If we can not reproduce your result on the leaderboard….
  - You will receive 0 point in the performance part
  - However, minor errors are acceptable
- We will excute your code on Linux system
  - Make sure your code can be excuted on Linux system before submission
- Deadline: 2025/06/08 23:59

Report Submission

- Schedule
- Data
- Evaluation
- Code Submission
- Report Submission
- Grading

Report Submission

- Only the team leader need to upload the code to NTU COOL
- For presentation teams…
  - Upload your presentation slide in ppt format
- For other team…
  - Upload your report in pdf format
- Deadline: 2025/06/08 23:59

Grading

- Schedule
- Data
- Evaluation
- Code Submission
- Report Submission
- Grading

Grading

- Performance (60%)
  - Average **Acc** (30%)
  - Average **Comp** (30%)
- Report (40%) (For Top 10 Teams)
  - Novelty and technical contribution (15%)
  - Experiment completeness (15%)
  - Oral Presentation (10%)
- Report (40%) (For Others)
  - Novelty and technical contribution (20%)
  - Experiment completeness (20%)
- Bonus (10%)
  - Reconstruction with sparse sequence
  - The baseline would be announced later

Points (For each Metric)	# of Teams
30%	1
29%	2
28%	2
26%	The rest teams / 4
24%	The rest teams / 4
22%	The rest teams / 4
20%	The rest teams / 4

Reminder

- Please start working on the project as early as possible.
- Please read and follow the rules carefully.
- Taking any unfair advantages (e.g., plagiarism) over other class members is strictly prohibited.
  - Violating university policy would result in F for this course.
- If you have any problems on the project …
  - Issue it on the NTU COOL forum
  - Send email to jackmafan@media.ee.ntu.tw (范宇清)