Fine-tuning 3D Pointmap Matching Foundation Models for Underwater Environments

Student: Aleksandra Novikova
Supervisor: Jonathan Sauder
Optional research project in ECEO Laboratory, EPFL
January 2025

I. Introduction

The degradation of coral reefs poses a significant threat to marine ecosystems, driven by global warming and other human activities. Coral reefs, often referred to as the "rainforests of the sea" [10], are among the most biodiverse and ecologically critical ecosystems on the planet, providing habitat for countless marine species and serving as natural barriers that protect coastlines from erosion and storm surges. Despite their importance, coral reefs are facing unprecedented threats due to climate change, pollution, and local human activity [7]. Rising global temperatures are a particularly severe driver of coral bleaching [18], leading to widespread die-offs that threaten the long-term survival of these ecosystems, making their monitoring and conservation an urgent global concern.

In recent years, advances in machine learning and computer vision have opened new avenues for monitoring coral reefs through underwater video analysis [2, 17, 16]. 3D reconstruction techniques, in particular, enable detailed mapping of coral reefs by integrating video data from different angles and timeframes. This integration provides researchers with valuable insights into coral health, growth, and damage over time.

Underwater environments, however, present unique challenges for 3D reconstruction [8]. Factors such as light scattering, water turbidity introduce noise and reduce the accuracy of standard techniques. Additionally, moving objects such as fish, sea turtles, swaying macroalgae, or other divers can interfere with reconstruction pipelines. These challenges necessitate methods that are not only robust but also specifically tailored to the underwater domain.

This semester project focuses on evaluating state-of-the-art 3D reconstruction methods on underwater video data, with a specific emphasis on videos from the Red Sea—a region known for its unique coral diversity. The goal is to help devise more effective and efficient 3D reconstruction approaches for this domain. Specifically, we compare traditional methods such as COLMAP [19, 20] and GLOMAP [15], alongside cutting-edge machine learning models such as DUSt3R [22] and MASt3R [9]. Furthermore, we fine-tune these models on underwater datasets to enhance their performance for this challenging application. By addressing the unique challenges posed by underwater environments, this research seeks to advance tools for the monitoring and conservation of coral reef ecosystems.

The results of this study will contribute to the growing field of underwater 3D reconstruction by providing insights into the relative strengths and limitations of different approaches. Moreover, the ability to integrate video data from different angles and timeframes will facilitate long-term monitoring efforts, allowing researchers to track changes in coral reefs more effectively and prioritize conservation efforts.

II. Related Work

3D reconstruction has emerged as a critical tool in marine science, providing researchers with high-resolution spatial models for analyzing underwater ecosystems. Traditional techniques for 3D reconstruction, such as Structure-from-Motion (SfM) [19, 13, 24] and Multi-View Stereo (MVS) [20, 5, 25], have demonstrated success in terrestrial and aerial applications but face unique challenges [6, 8] when applied to

underwater environments. Light refraction, scattering, and the dynamic nature of aquatic scenes introduce noise and errors, necessitating the development of robust reconstruction algorithms specifically tailored to address the challenges of underwater environments.

A. Classical Approaches

Traditional methods for 3D reconstruction are based on geometric principles, relying on algorithms such as SfM [19, 13, 24] and MVS [20, 5, 25]. One of the most widely used frameworks in this category is **COLMAP**, which is a comprehensive pipeline for SfM and MVS. COLMAP is renowned for its robustness in estimating camera poses and generating dense point clouds from multiple images. The pipeline consists of three main stages: first, the tool employs feature extraction and matching algorithms such as SIFT [11] (Scale-Invariant Feature Transform) to establish feature correspondences. Next comes the SfM stage: COLMAP uses feature correspondences to estimate camera poses and sparse 3D point clouds, along with bundle adjustment techniques for global optimization. The final stage of the pipeline is MVS: COLMAP generates dense 3D models by propagating depth estimates across images. Depth maps are fused into a final dense point cloud using a volumetric fusion algorithm.

COLMAP has been extensively used in various terrestrial and aerial applications, but its application to underwater environments is less common. Studies have shown that COLMAP can be adapted to underwater use cases by incorporating pre-processing techniques such as image dehazing, color correction, and light scattering compensation [6]. However, the computational requirements of this algorithm are very high, both in terms of memory and time, and combined with its sensitivity to noise and outliers, it poses challenges when dealing with dynamic underwater scenes, especially with long video recordings of underwater environments.

Recent work **GLOMAP** [15] represents an improvement of the SfM step in the COLMAP algorithm. COLMAP is an incremental SfM system, which iteratively adds new cameras and refines the reconstruction through repeated bundle adjustments. Therefore, its scalability is limited, as computational costs grow substantially with the size of the dataset.

In contrast, GLOMAP employs a global SfM paradigm, performing a unified estimation of camera poses and 3D structure. It delivers accuracy and robustness comparable to incremental methods while maintaining the efficiency characteristic of global approaches. The core innovation lies in its joint global positioning step, which simultaneously estimates both camera positions and 3D points. This approach contrasts with traditional global SfM pipelines that decouple these tasks, leading to inaccuracies during translation averaging and triangulation due to noise and scale ambiguities. Experimental results show that GLOMAP is orders of magnitude faster than COLMAP and comparable in runtime to other global methods while achieving a similar level of accuracy and robustness as state-of-the-art incremental SfM.

B. Machine Learning Approaches

Recent advancements in machine learning have significantly improved the field of 3D reconstruction, offering robust and scalable techniques to model complex scenes [12, 21, 1]. Unlike traditional geometric approaches, neural networks have several advantages, especially in challenging scenarios such as underwater videos: ML approaches can be trained to recognize patterns and learn domain-specific features even under conditions of light attenuation, scattering, and color distortion inherent to underwater videos.

The state-of-the-art model **DUSt3R** [22] is an innovative solution for 3D reconstruction tasks, demonstrating remarkable results across a wide range of scenarios. DUSt3R is an algorithm for Dense Unconstrained Stereo 3D Reconstruction. The core component is a neural network that can regress a dense and accurate scene representation solely from a pair of images, without any prior information about the scene or cameras. The resulting scene representation is based on 3D pointmaps ($X \in R^{W \times H \times 3}$), where each pixel in an image is mapped to a corresponding 3D point in a canonical reference frame. This approach eliminates the need for explicit camera parameter estimation.

A pair of images (I_1, I_2) is processed using a shared Transformer-based architecture to produce corresponding pointmaps $(X_{1,1}, X_{2,1})$. These are expressed in the same coordinate frame, enabling efficient triangulation and depth computation.

The architecture employs ViT encoders and Transformer decoders with cross-attention to ensure alignment between the views. A confidence-weighted regression loss ensures robustness to ill-defined 3D points, such as those in translucent or textureless regions:

$$\mathcal{L}_{\text{conf}} = \sum_{v\in\{1,2\}} \sum_{i\in D_v} C^i_{v,1} \cdot \ell_{\text{regr}}(v,i) - \alpha \log(C^i_{v,1}), \tag{1}$$

where where $v \in \{1, 2\}$ - represents the image index, D_v - is the set of pixels in view v, $C_{v,1}$ represents confidence scores (output of the model), and ℓ_{regr} minimizes the Euclidean distance between predicted and ground-truth normalized pointmaps, and α is a hyper-parameter controlling the regularization term.

The primary advantage of training on pointmaps is that it seamlessly integrates depth estimation with camera parameter estimation. The training data comprises a large collection of image pairs sourced from rendered scenes or successful SfM reconstructions, where ground truth depth and camera parameters are used to generate the pointmaps.

An even newer model, **MASt3R** [9] (Matching and Stereo 3D Reconstruction), represents a significant improvement over the DUSt3R algorithm by introducing a second head to the neural network that outputs dense local feature maps ($F \in R^{H\times W\times d}$). The main goal of this head is to encourage each local descriptor from one image to match with at most a single descriptor from the other image that represents the same 3D point in the scene. For this purpose, a novel matching loss (based on InfoNCE [14]) is introduced to explicitly train for dense correspondences, thereby increasing precision.

The final loss combines both heads by summing the original confidence loss from the DUSt3R model and the new matching loss, weighted by a coefficient β:

$$L_{\text{total}} = L_{\text{conf}} + \beta L_{\text{match}} \tag{2}$$

MASt3R introduces a fast iterative algorithm to extract reciprocal matches efficiently, reducing the complexity from $O(W^2H^2)$ to $O(kWH)$ while maintaining robustness. This algorithm iteratively refines matches by aligning pixels in coarse-to-fine scales, achieving dramatic speedups and enhanced accuracy. Moreover, the matches produced by MASt3R can be directly used to perform unconstrained structure-from-motion, as shown in their accompanying SfM study [4], enabling 3D reconstruction without prior constraints on camera poses or scene geometry.

III. Methodology: Data and GLOMAP

The primary goal of this study was to evaluate and test the combined performance of classical and machine learning-based 3D reconstruction models. The methodology involved processing diverse underwater datasets, testing classical methods, and fine-tuning state-of-the-art machine learning models on customized datasets. This section provides a detailed explanation of the main algorithms and methods used for this purpose.

A. Data Preparation

The first stage involved preparing the data for subsequent testing and model analysis. The original data consisted of the same underwater scene captured under varying conditions. Specifically:

1) **Parallel Shooting Paths:** The scene was recorded from three parallel paths - left, center, and right.
2) **Directionality:** For each path, the scene was shot in two opposite directions - forward and backward.
3) **Time:** For every combination of path and direction, the scene was captured at two different time steps: T_1 and T_2 - they were taken on different days and different times of the day (morning and afternoon).

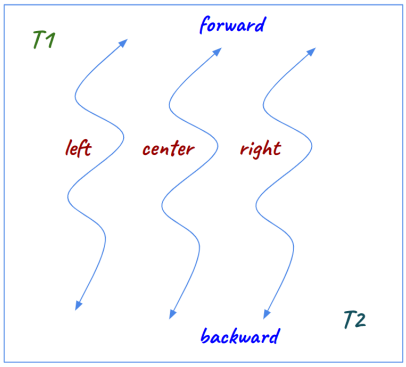

Fig. 1: Schematic representation of the original scenes

A schematic representation of the original scenes is shown in Figure 1.

To prepare these multifaceted data for experiments, the data was preprocessed to create scenarios with: different frame rates, varying image resolutions, and different combinations of input conditions.

As a result, a large and diverse dataset was created for testing.

Examples of test images are shown in Figure 2.

Fig. 2: Examples of test images from the dataset

B. Testing Classical Algorithms

As a classical computer vision algorithm, GLOMAP served as the baseline for the initial experiments. Therefore, the second stage involved testing this algorithm. It was applied to the preprocessed datasets. Its results for the scenario involving the left-center-right paths with a forward direction and one time interval T_1 can be seen in Figure 3.

While GLOMAP demonstrated strong reconstruction accuracy, its drawbacks included: long processing times, limited flexibility for improvement using the provided datasets, as it is a non-machine-learning approach, artifacts related to the specific characteristics of underwater videos.

C. Dataset Creation for Fine-tuning Machine Learning Models

The analysis and testing of the DUSt3R and MASt3R models will be described in detail in the **Experiments** section. Both models were trained on large volumes of diverse data and produced reasonable results on our test datasets. However, we still observed various artifacts in the underwater videos. Therefore, we aimed to investigate whether fine-tuning large pretrained models on a dataset of underwater images could improve their performance. Consequently, the next stage involved the development of such a dataset for fine-tuning.

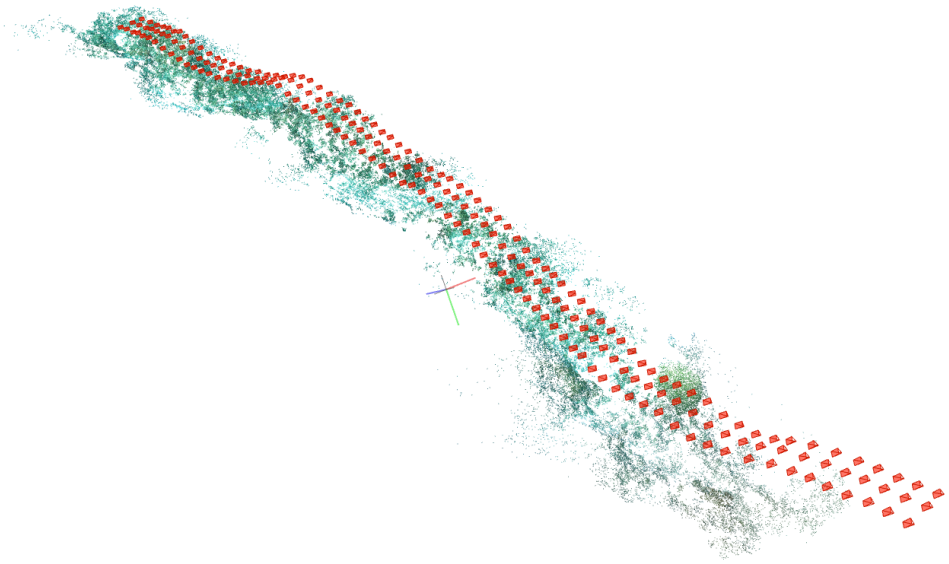

Fig. 3: Results of GLOMAP for the left-center-right scenario with forward direction and time interval T_1.

Using GLOMAP, ground-truth point clouds were generated for several training data scenes (which were distinct and did not overlap with the test datasets).

Following this, a dataset class was implemented to preprocess the output data from GLOMAP and pass it to the DUSt3R or MASt3R model for fine-tuning.

The preprocessing steps included:

- **Camera Parameter Processing:** Intrinsic parameters were converted into a standard matrix with translation and rotation, and extrinsic camera parameters were translated into world coordinates.
- **Depth Map Preprocessing:** Depth maps were prepared for each image to generate ground-truth 3D point maps.

D. Pair Selection Algorithm

Since both DUSt3R and MASt3R operate on image pairs, an algorithm was implemented to select diverse and representative image pairs. The implemented algorithm is very similar to the one described in Paper Weinzaepfel et al. [23].

The main idea is to select image pairs that overlap, but not too much or too little, while maximizing diversity among the pairs. The steps of the algorithm are as follows:

1) **IoU Calculation:** For each pair of images, their intersection-over-union (IoU) is calculated: IoU of points in the dense 3D point clouds. This was done using the nearest neighbors algorithm: for each point in the first point cloud, the nearest point in the second point cloud was found within a certain radius, which was a hyperparameter. All image pairs with an IoU that is too high or too low are discarded.
2) **Scoring Function:** For each remaining image pair, a target function (score) is calculated as in Weinzaepfel et al. [23]:

$$s(I_1, I_2) = \text{IoU}(I_1, I_2) \cdot 4 \cdot \cos(\alpha) \cdot (1 - \cos(\alpha)),$$

where I_1 and I_2 represent an image pair, and α is the angle between the two images. The meaning of this function is that it reaches its maximum when $\text{IoU}(I_1, I_2) = 1$ at an angle $\alpha = 60°$ and we "encourage" angles from $0°$ to $90°$.

3) **Pair Selection:** Finally, a greedy algorithm is used to select a set of image pairs with the highest scores.

To create the train dataset, experiments were conducted with various hyperparameter settings to identify the optimal parameters, ensuring the dataset was both large and diverse. The final score distribution for the dataset and its cumulative distribution function (CDF) is presented in Figure 4.

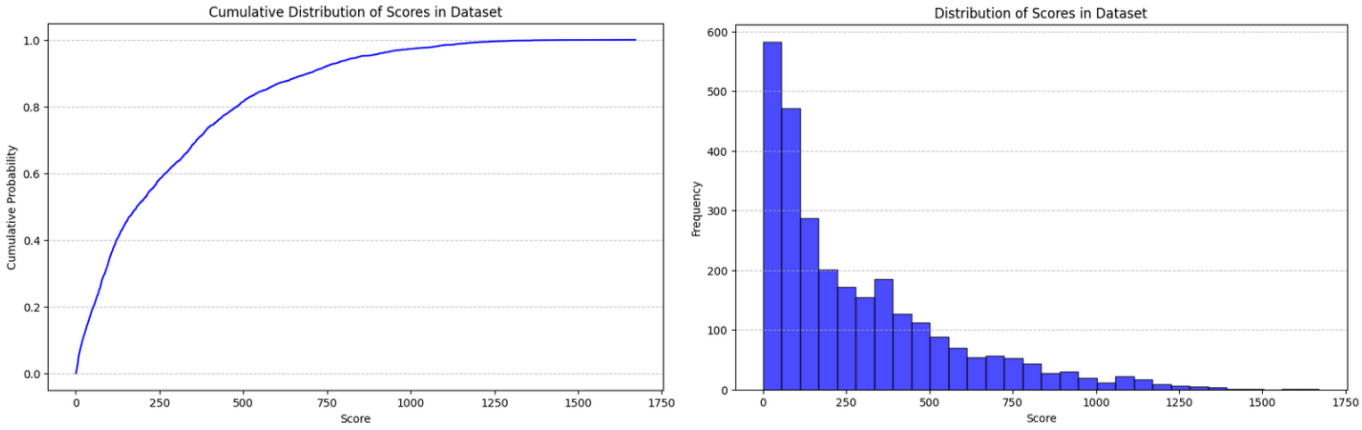

Fig. 4: Cumulative distribution function and Score distribution for the train dataset

In total, approximately 3000 image pairs were included in the train dataset for model fine-tuning, with around 500 pairs excluded from training and used for testing during training. The ends of the videos were used as test data, and the beginnings of the videos were used for train data.

The angle distribution is displayed in Figure 5. It shows that all angles are in the range of $0°$ to approximately $40°$ degrees, with the majority being small angles. This is because the primary movement in our dataset scene is along a straight line.

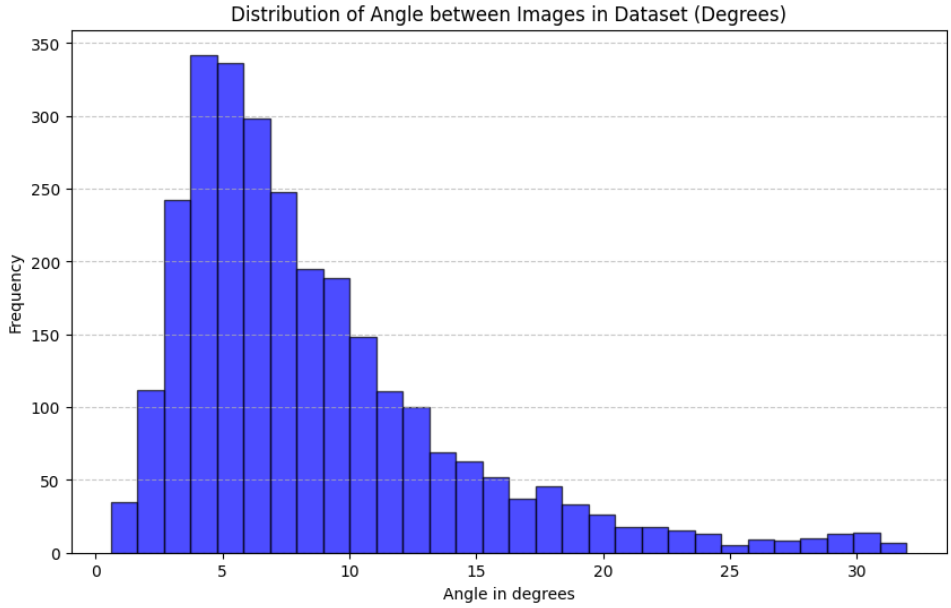

Fig. 5: Angle distribution for the train dataset

The distribution of frame distances (i.e. how many frames apart in the video an image pair is) for pairs with the highest and lowest scores is shown in Figure 6. We observe that the "best" image pairs have a difference of 5-20 frames (corresponding to approximately 5-20 seconds of video), while the "worst" image pairs have a difference greater than 15 frames (i.e., more than 15 seconds of video).

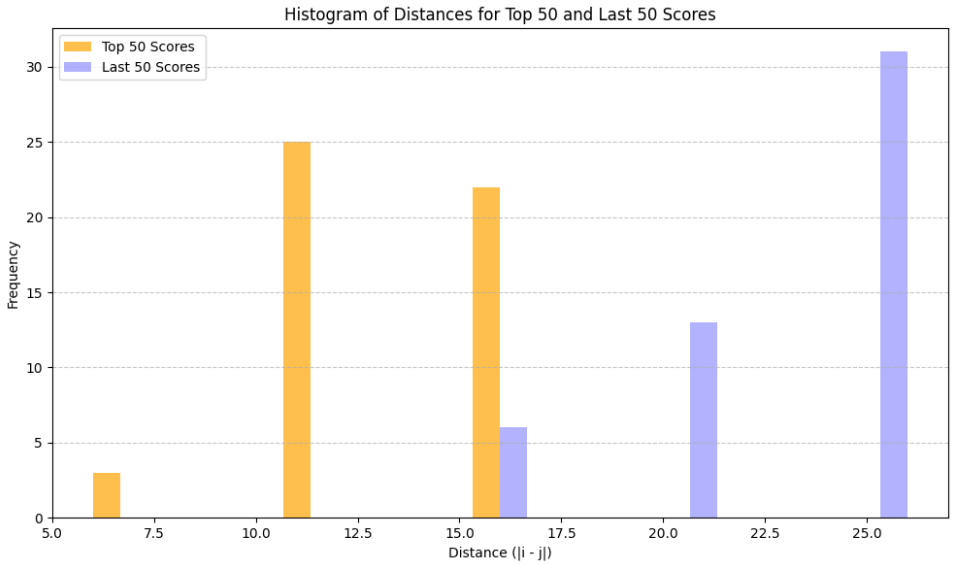

Fig. 6: Distributionof frame distances for pairs with the highest and lowest scores for the train dataset

An example of a “top-score” image pair is shown in Figure 7, while an example of a “lowest-score” image pair is shown in Figure 8.

Thus, a large and diverse dataset was prepared for fine-tuning.

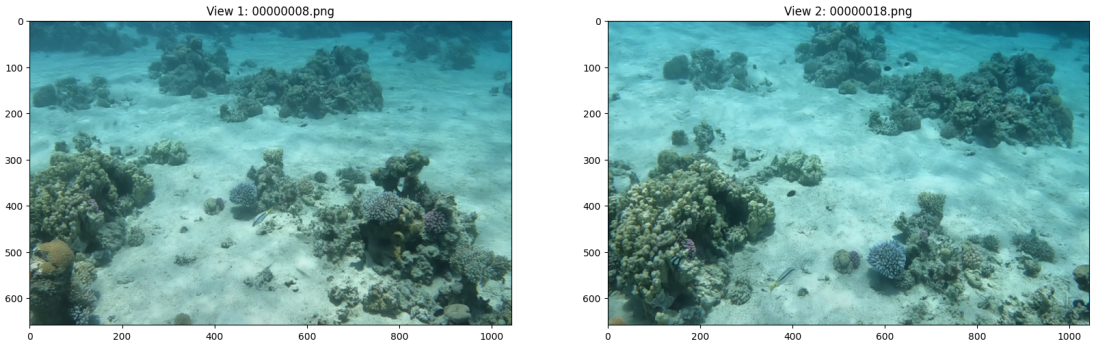

Fig. 7: Example of an image pair with the highest score in the train dataset

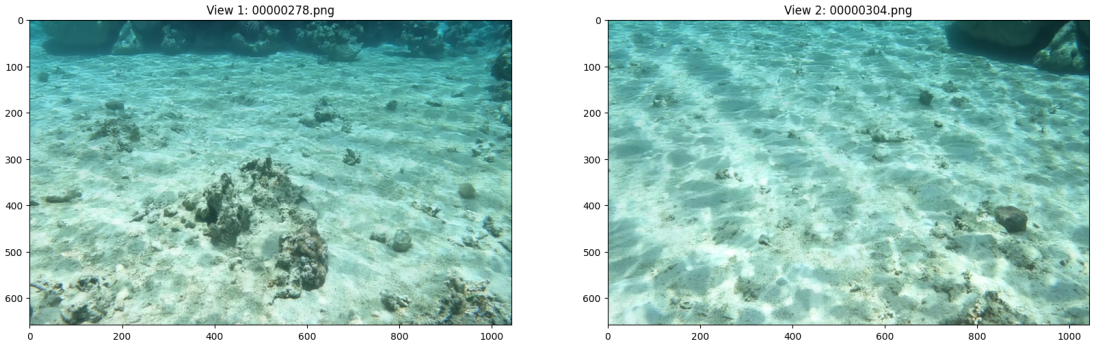

Fig. 8: Example of an image pair with the lowest score in the train dataset

E. Checkpointing

The pretrained DUSt3R and MASt3R models each have a size exceeding 2 GB. Due to limited computational resources (the fine-tuning process utilized a 32 GB GPU), a technique to reduce memory overhead during training was implemented to ensure the batch size could be greater than 1.

We leveraged PyTorch's mechanism for **gradient checkpointing** [3]. This technique strategically trades off computation for memory by recomputing intermediate activations during the backward pass instead of storing them. By wrapping parts of the model in checkpointed segments, memory usage was significantly reduced, enabling the fine-tuning process to proceed without exceeding memory limits.

F. Fine-tuning process

The fine-tuning process was conducted over 100 epochs, with the first 5 epochs serving as warmup epochs. The batch size was set to 8, and the initial learning rate was 0.0001.

The training progress on the train dataset is illustrated in Figure 9 with separate plots for each loss term as well as the total loss. The training progress on the test dataset is shown in Figure 10.

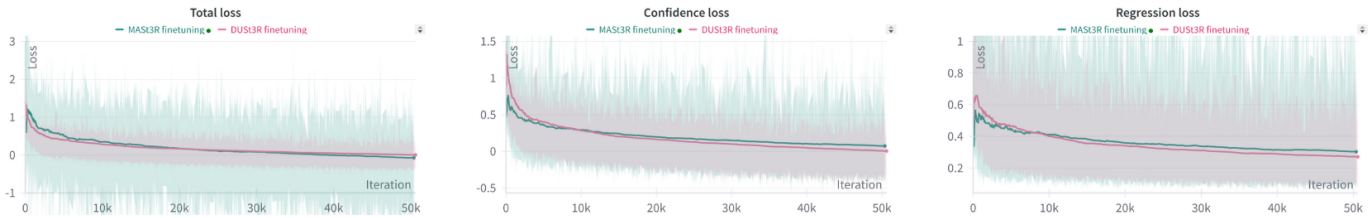

Fig. 9: Total, Confidence and Regression train losses during fine-tuning process

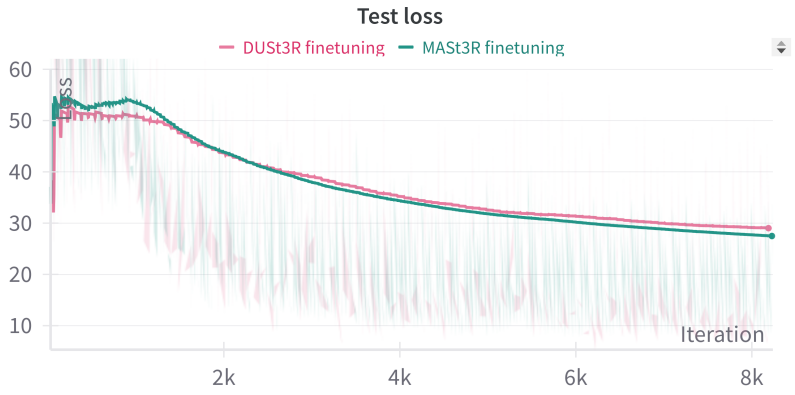

Fig. 10: Test loss during fine-tuning process

IV. Experiments

A. Test Scenarios

The evaluation focuses on qualitative reconstruction analysis, primarily distinguishing between successful and unsuccessful outcomes. Quantitative evaluation is beyond the scope of this report. The final 4 models (initial pretrained DUSt3R and MASt3R, and fine-tuned DUSt3R and MASt3R) were tested on several scenarios:

1) *Single video:* with the forward direction along the center path for time T_2 ("**T_2-center-forward**").
2) *Different paths:* 2 videos for DUSt3R and 3 videos for MASt3R with the forward direction for time T_1 with left, center and right paths. We used just left and center paths for DUSt3R because otherwise

the scene would be too short due to limitations on the number of frames ("**T_1-left-center-forward**" for DUSt3R and "**T_1-left-center-right-forward**" for MASt3R).

3) *Different times:* 2 videos with the forward direction along the center path for times T_1 and T_2 ("**T_1-T_2-center-forward**").
4) *Different directions:* 2 videos with forward and backward directions along the center path for time T_1 ("**T_1-center-forward-backward**").

The scenario with different times can be particularly useful in cases where it is necessary to compare the same scene over time, such as across different days, months, etc.

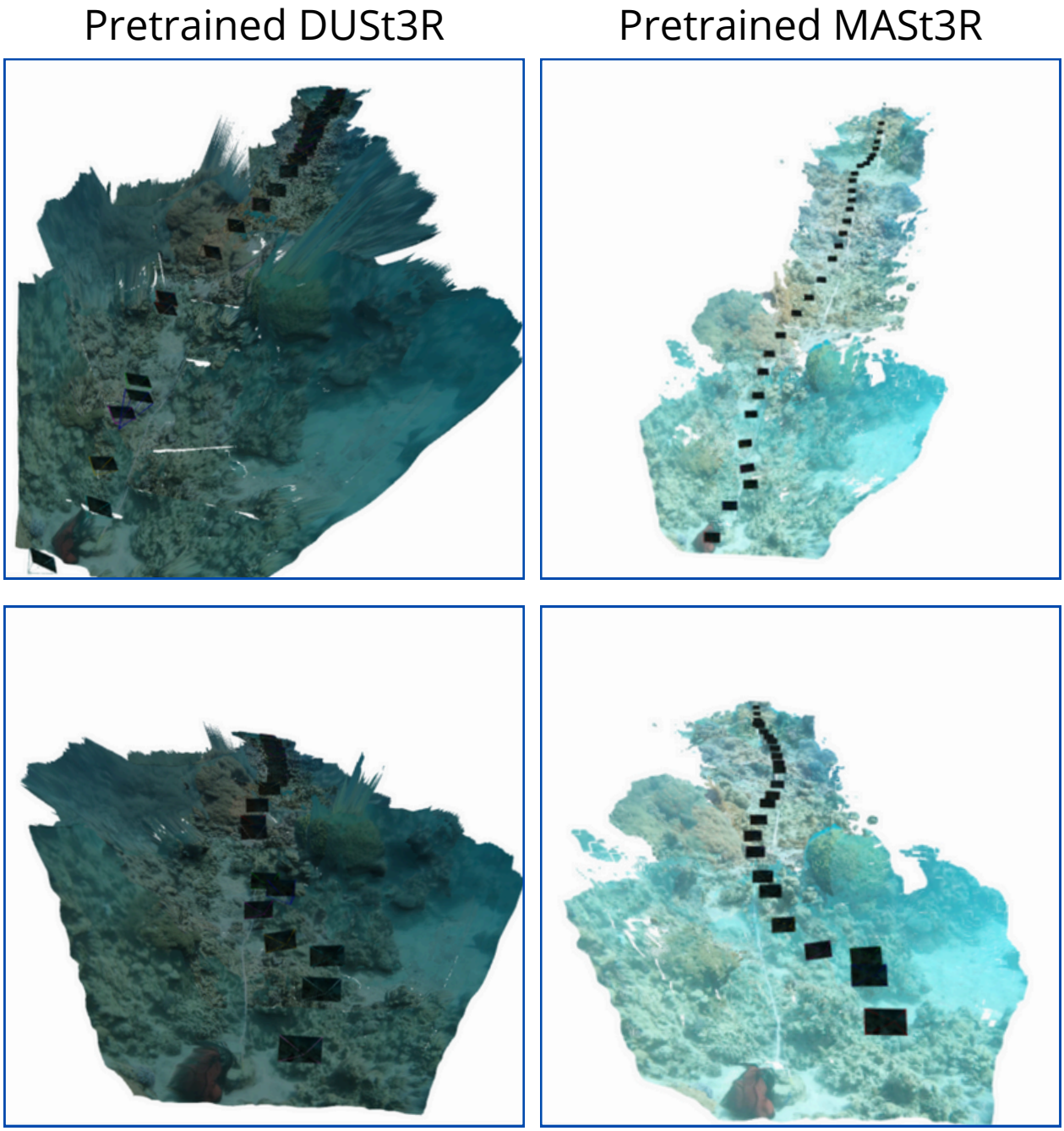

Fig. 11: **T_2-center-forward**: Results of the initial pretrained models for a single video with forward direction and time T_2.

B. Comparison of DUStR and MASt3R

We experimented with the following model hyperparameters:

- **DUSt3R:** Utilized the "all possible pairs" method, where the model processes every pair of images from the input set. This approach maximizes the potential feature matching but incurs a higher computational cost.
- **MASt3R:** Employed the sliding window method, processing images sequentially within a window of size 3. This method achieves a linear complexity in the number of pairs, making it computationally more efficient.

All other parameters were set to their default values for these experiments.

We also conducted several tests with different hyperparameters but settled on these as they provide a good balance between runtime efficiency and output quality for each model. The DUSt3R model fails to handle 3D reconstruction with the sliding window algorithm, producing very poor results. That's why the "all possible pairs" method was chosen for DUSt3R. The MASt3R model works well with any pair selection algorithm, so we chose the sliding window algorithm as it is the most time-efficient and suits our videos since video frames are processed sequentially.

Consequently, one of the main drawbacks of the DUSt3R model lies in its computational complexity: since "all possible pairs" are processed in quadratic time, as opposed to the sliding window algorithm — where the number of pairs is linear — the DUSt3R algorithm is significantly slower than MASt3R. The DUSt3R model takes approximately 10 minutes to process 20 frames, whereas the MASt3R model processes about 150 frames in the same time.

The results of the initial pretrained models for a single video with forward direction and time T_2 can be seen in Figure 11.

In addition to runtime, we observe a significant improvement in the quality of the point cloud in the MASt3R model compared to DUSt3R. This is particularly evident at the edges of the point cloud (Figure 14): in the DUSt3R model, the edges are highly distorted and extremely inaccurate, whereas this issue is absent in the MASt3R model. Furthermore, the output of DUSt3R shows noticeably more artifacts, distortions, and sharp lines (Figures 11, 14).

C. Comparison with Fine-tuned Models

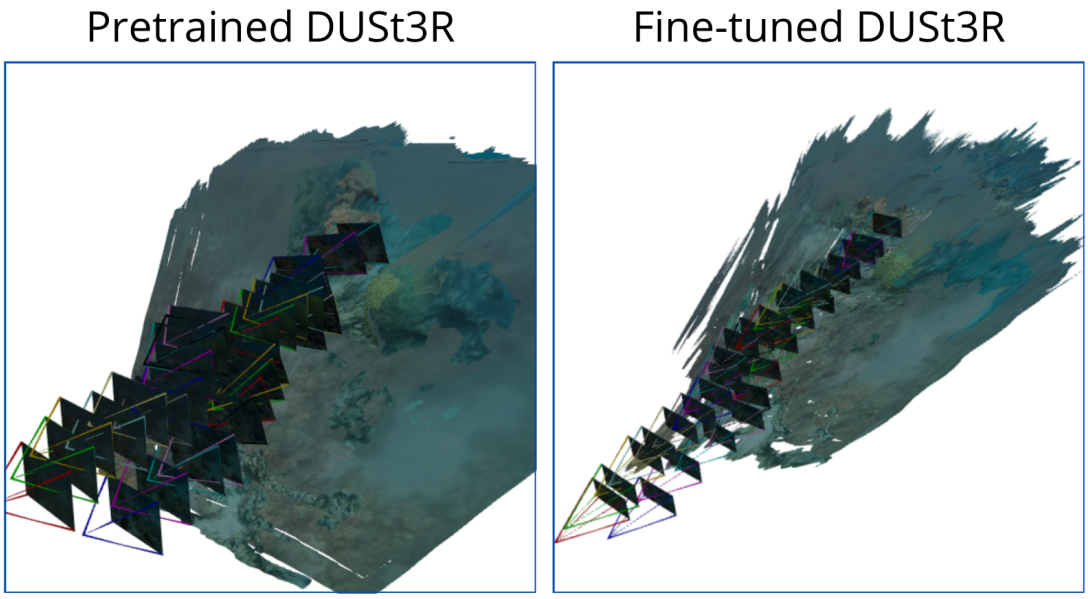

Fig. 12: **T_1-T_2-center-forward**: Results of the pretrained and fine-tuned DUSt3R models

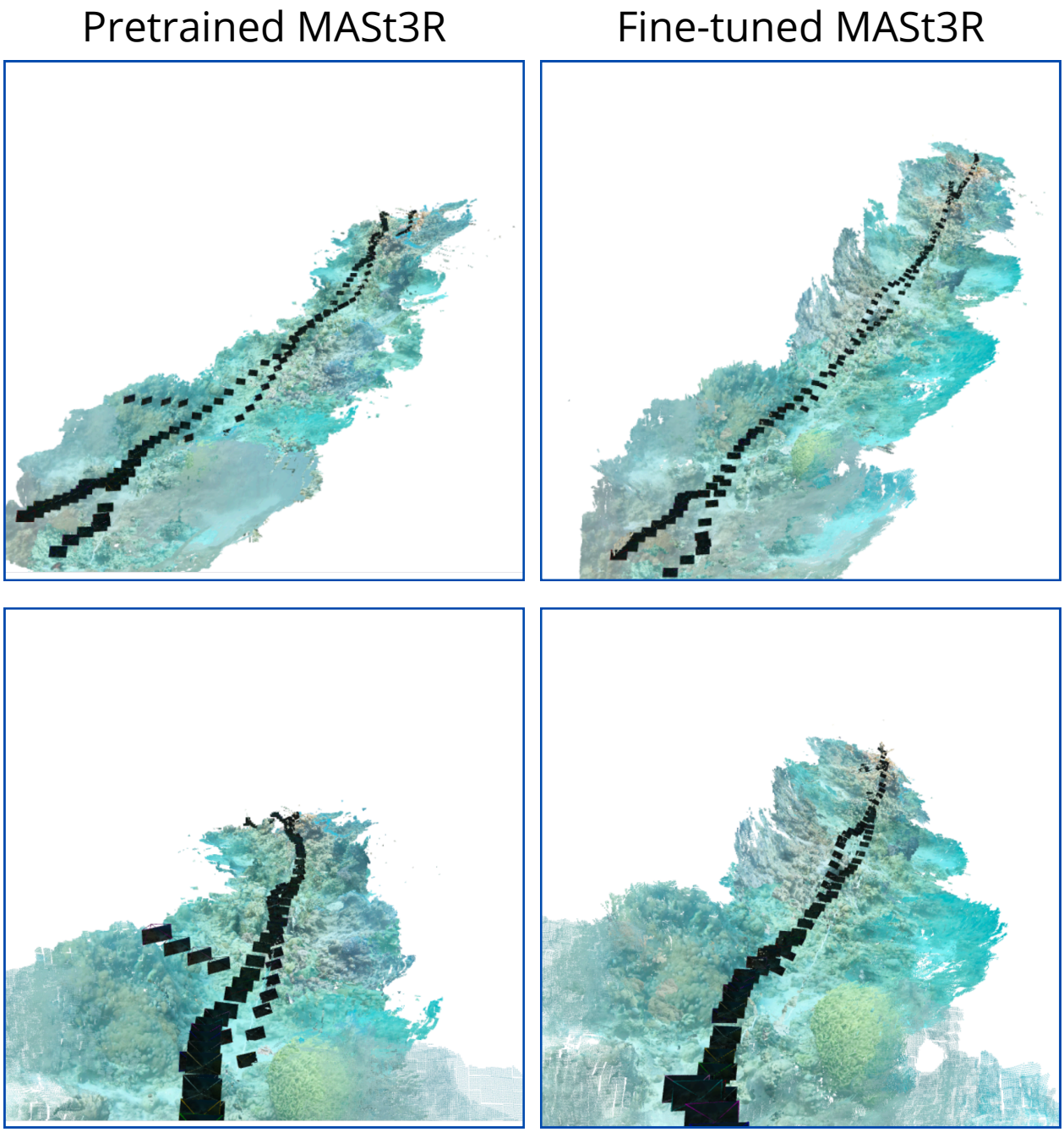

Fig. 13: **T_1-T_2-center-forward**: Results of the pretrained and fine-tuned MASt3R models

The results of comparing the initial pretrained models with our fine-tuned versions can be seen in Figures 12, 13. Additional results can be found in the Appendix in Figures 16, 15, 17, 20, 22.

For both models, fine-tuning introduces changes that are generally consistent in their impact on the results. Specifically, the fine-tuned models show both improvements in specific aspects and new challenges introduced by overfitting. While the initial and fine-tuned models yield similar results overall, fine-tuning appears to amplify certain artifacts. This is particularly evident in the case of DUSt3R, as shown in Figure 18: the fine-tuned model produces less accurate results. Some areas did not match at all or matched incorrectly, resulting in large gaps visible in the 3D cloud. Moreover, the artifacts themselves are clearly visible: For example at Figure 12 at the edges of the 3D cloud in the fine-tuned model, there are many

sharp lines. This can also be observed in MASt3R in Figure 19: the fine-tuned model exhibits more gaps, appearing to be sparser.

These artifacts may be related to GLOMAP and COLMAP artifacts. As mentioned earlier, these algorithms may perform poorly in underwater environments and produce many artifacts. For example, in some training data images, the output depth map was almost completely black, meaning the algorithm failed to map points from these images. Such outliers negatively affect the fine-tuning. Our fine-tuned models might overfit to these inaccuracies and incorrect point coordinate predictions.

There is also good news:

The fine-tuned models align frames better and therefore predict trajectories more accurately, which is especially noticeable in camera movement. For example, in Figures 12, the camera positions in the fine-tuned DUSt3R model form an almost straight line, whereas in the initial pretrained model, this is not the case.

This is even more evident in the MASt3R model. In Figure 13, results are presented for two videos of the same path but recorded at different times. In the fine-tuned model, the camera positions align much more closely, resembling a straight line, whereas in the initial pretrained model, there are many deviations from the main path, indicating that the same frames at different times did not match properly.

Thus, after fine-tuning with our data, we observe both improvements and drawbacks. The models have become significantly better at predicting movement trajectories but now introduce new artifacts: some parts of the 3D model are predicted with low accuracy and high sparsity, and additional ”sharp lines” and outliers have appeared. Our ground-truth training data is far from ideal, and the models overfit to the deficiencies and artifacts of the specific algorithm. Moreover, our training dataset included only three different videos with different scenes. Naturally, increasing the dataset size could lead to improved fine-tuning results.

D. General results

The overall results are summarized in Table I.

	GLOMAP	**Dust3r**	**Mast3r**	**Dust3r-Finetuned**	**Mast3r-Finetuned**
Single video	✓(sparse only)	✓(limited to 30 images)	✓	✓	✓
Different paths	✓(sparse only)	✓(limited to 30 images)	✓	✓	✓
Different times	x	x	x	✓	✓
Different directions	x	x	x	x	x

TABLE I: Comparison of Models on Various Scenarios, where x - means that the algorithm works poorly enough, and ✓- means well enough

The DUSt3R and MASt3R models perform well in simple cases, such as a single forward path (Figures 15, 20) or a few parallel paths (Figures 16, 22). They also handle matching videos recorded at different times relatively well (Figures 12, 13), though with noticeable inaccuracies. However, the fine-tuned models significantly improve trajectory predictions in such cases. Nevertheless, despite the original authors claiming these models work well for matching frames shot from very different angles, we observe that in our case, the models struggle to predict the 3D model correctly for forward-backward directions (Figures 17, 21). Instead of matching frames directly opposite each other, they predict them as ”floor-ceiling,” (Figure 21) or just not a match at all (Figure 17).

V. Conclusion

This study explored the use of 3D computer vision techniques to reconstruct underwater environments, focusing on their application to coral reef monitoring in the Red Sea. The research aimed to address the unique challenges posed by underwater scenes, including light scattering, water turbidity, and dynamic marine ecosystems, through a comparative evaluation of classical and machine learning-based 3D reconstruction methods.

In this study:

- Test dataset was collected.
- Training dataset was collected: a set of videos was aligned with GLOMAP to obtain camera parameters and depth maps, which was in turn transformed into a dataset of image pairs with appropriate overlap to fine-tune DUSt3R and MASt3R.
- The DUSt3R and MASt3R models were fine-tuned on underwater data.
- A comparative analysis was conducted between the DUSt3R and MASt3R models, as well as their fine-tuned versions.

The classical approach, GLOMAP, demonstrated strong reconstruction accuracy but was hindered by long processing times and limited adaptability to underwater-specific conditions. In contrast, machine learning models, DUSt3R and MASt3R, leveraged neural architectures to overcome underwater challenges. MASt3R outperformed DUSt3R in accuracy and computational efficiency, thanks to its dual-head architecture for feature matching and correspondence generation. The fine-tuning of these models on a curated underwater dataset highlighted both opportunities and limitations. While fine-tuned versions improved frame-to-frame alignment and trajectory estimation, they also introduced new artifacts, likely stemming from the imperfections in the GLOMAP-generated training data. Furthermore, the small dataset size limited the generalization of the fine-tuned models.

Future research can take several directions. First, improving the dataset: a larger quantity of data will help make the models more generalizable. Additionally, the images themselves can be enhanced, for example, through color correction and light scattering compensation. Other models that perform better specifically in underwater environments could also be used as ground truth instead of GLOMAP and COLMAP.

Second, adding metrics would be a very useful enhancement. In our study, all model evaluations were done ”by eye,” which is often very challenging and, of course, imprecise. At a minimum, the size of the point clouds could be analyzed, which might reveal sparser clouds. Outliers could also potentially be identified (e.g., as points far from the ground truth).

Overall, this work provides insights into the relative strengths and weaknesses of different 3D reconstruction approaches for underwater environments. Further research can help overcome the limitations of this work and improve the final model’s results.

REFERENCES

[1] Eric Brachmann, Jamie Wynn, Shuai Chen, Tommaso Cavallari, Áron Monszpart, Daniyar Turmukhambetov, and Victor Adrian Prisacariu. Scene coordinate reconstruction: Posing of image collections via incremental learning of a relocalizer. In *ECCV*, 2024.

[2] JHR Burns, D Delparte, RD Gates, and M Takabayashi. Integrating structure-from-motion photogrammetry with geospatial software as a novel technique for quantifying 3d ecological characteristics of coral reefs. *PeerJ*, 3:e1077, 2015.

[3] Tianqi Chen, Bing Xu, Chiyuan Zhang, and Carlos Guestrin. Training deep nets with sublinear memory cost. *arXiv preprint arXiv:1604.06174*, 2016.

[4] Bardienus Duisterhof, Lojze Zust, Philippe Weinzaepfel, Vincent Leroy, Yohann Cabon, and Jerome Revaud. Mast3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. *arXiv preprint arXiv:2409.19152*, 2024.

[5] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1362–1376, 2009.

[6] Michele Grimaldi, David Nakath, Mengkun She, and Kevin Köser. Investigation of the challenges of underwater-visual-monocular-slam. *arXiv preprint arXiv:2306.08738*, 2023.

[7] Ove Hoegh Guldberg, Daniela Jacob, Michael Taylor, Marco Bindi, Sally Brown, Ines Angela Camilloni, Arona Diedhiou, Riyanti Djalante, Kristie L Ebi, Francois Engelbrecht, et al. Impacts of 1.5 c global warming on natural and human systems. 2018.

[8] Kai Hu, Tianyan Wang, Chaowen Shen, Chenghang Weng, Fenghua Zhou, Min Xia, and Liguo Weng. Overview of underwater 3d reconstruction technology based on optical images. *Journal of Marine Science and Engineering*, 11(5):949, 2023.

[9] Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with mast3r, 2024.

[10] Baolin Liao, Baohua Xiao, and Zhiyong Li. Coral reef ecosystem. *Symbiotic microbiomes of coral reefs sponges and corals*, pages 1–15, 2019.

[11] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.

[12] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[13] Pierre Moulon, Pascal Monasse, and Renaud Marlet. Adaptive structure from motion with a contrario model estimation. In *Computer Vision–ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part IV 11*, pages 257–270. Springer, 2013.

[14] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[15] Linfei Pan, Daniel Barath, Marc Pollefeys, and Johannes Lutz Schönberger. Global Structure-from-Motion Revisited. In *European Conference on Computer Vision (ECCV)*, 2024.

[16] Kailey H Pascoe, Atsuko Fukunaga, Randall K Kosaki, and John HR Burns. 3d assessment of a coral reef at lalo atoll reveals varying responses of habitat metrics following a catastrophic hurricane. *Scientific reports*, 11(1):12050, 2021.

[17] Tiny Remmers, Nader Boutros, Mathew Wyatt, Sophie Gordon, Maren Toor, Chris Roelfsema, Katharina Fabricius, Alana Grech, Marine Lechene, and Renata Ferrari. Rapidbenthos: Automated segmentation and multi-view classification of coral reef communities from photogrammetric reconstruction. *Methods in Ecology and Evolution*, 2024.

[18] Johan Rockström, Will Steffen, Kevin Noone, Åsa Persson, F Stuart Chapin III, Eric Lambin, Timothy M Lenton, Marten Scheffer, Carl Folke, Hans Joachim Schellnhuber, et al. Planetary boundaries: exploring the safe operating space for humanity. *Ecology and society*, 14(2), 2009.

[19] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[20] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.

[21] Cameron Smith, David Charatan, Ayush Tewari, and Vincent Sitzmann. Flowmap: High-quality camera poses, intrinsics, and depth via gradient descent. *arXiv preprint arXiv:2404.15259*, 2024.

[22] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024.

[23] Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. Croco v2: Improved cross-view completion pre-training for stereo matching and optical flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17969–17980, 2023.

[24] Changchang Wu. Visualsfm: A visual structure from motion system. *http://www. cs. washington. edu/homes/ccwu/vsfm*, 2011.

[25] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018.

APPENDIX

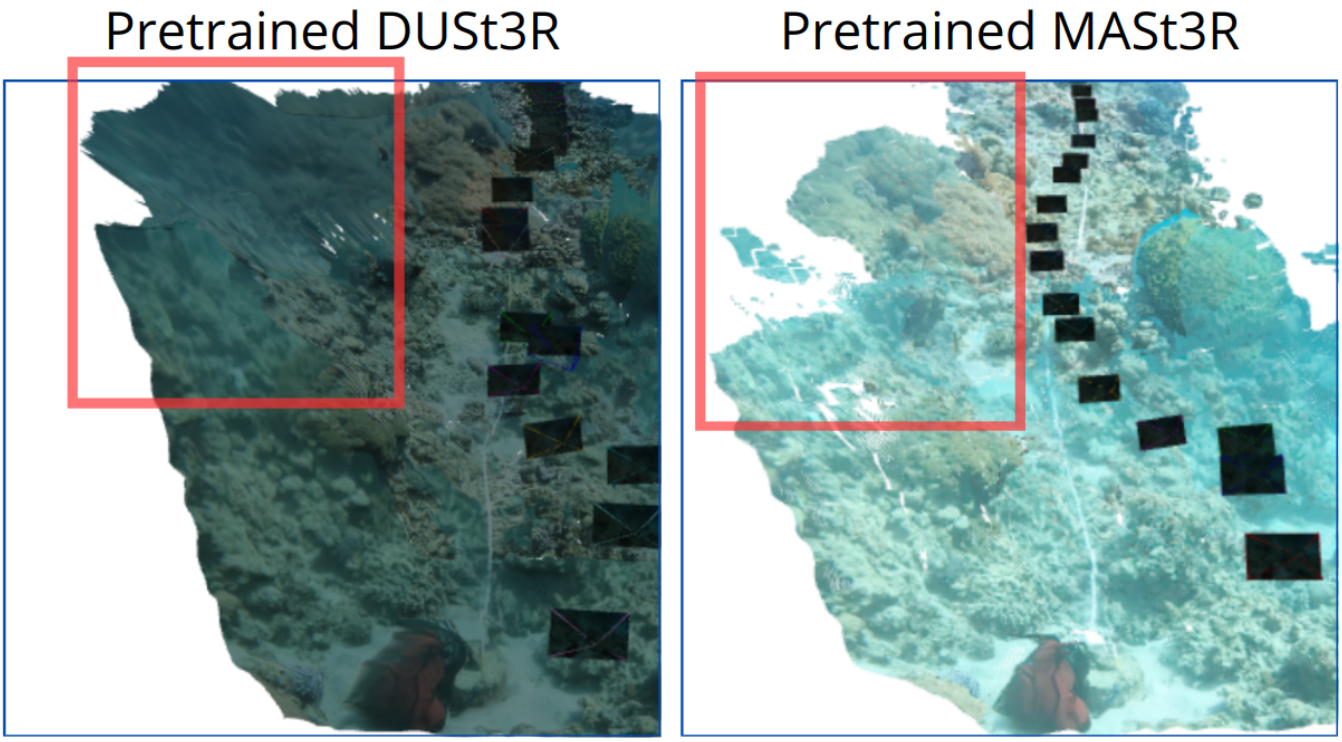

Fig. 14: **T_2-center-forward**: Results of the initial pretrained models for a single video with forward direction and time T_2 with clear differences indicated

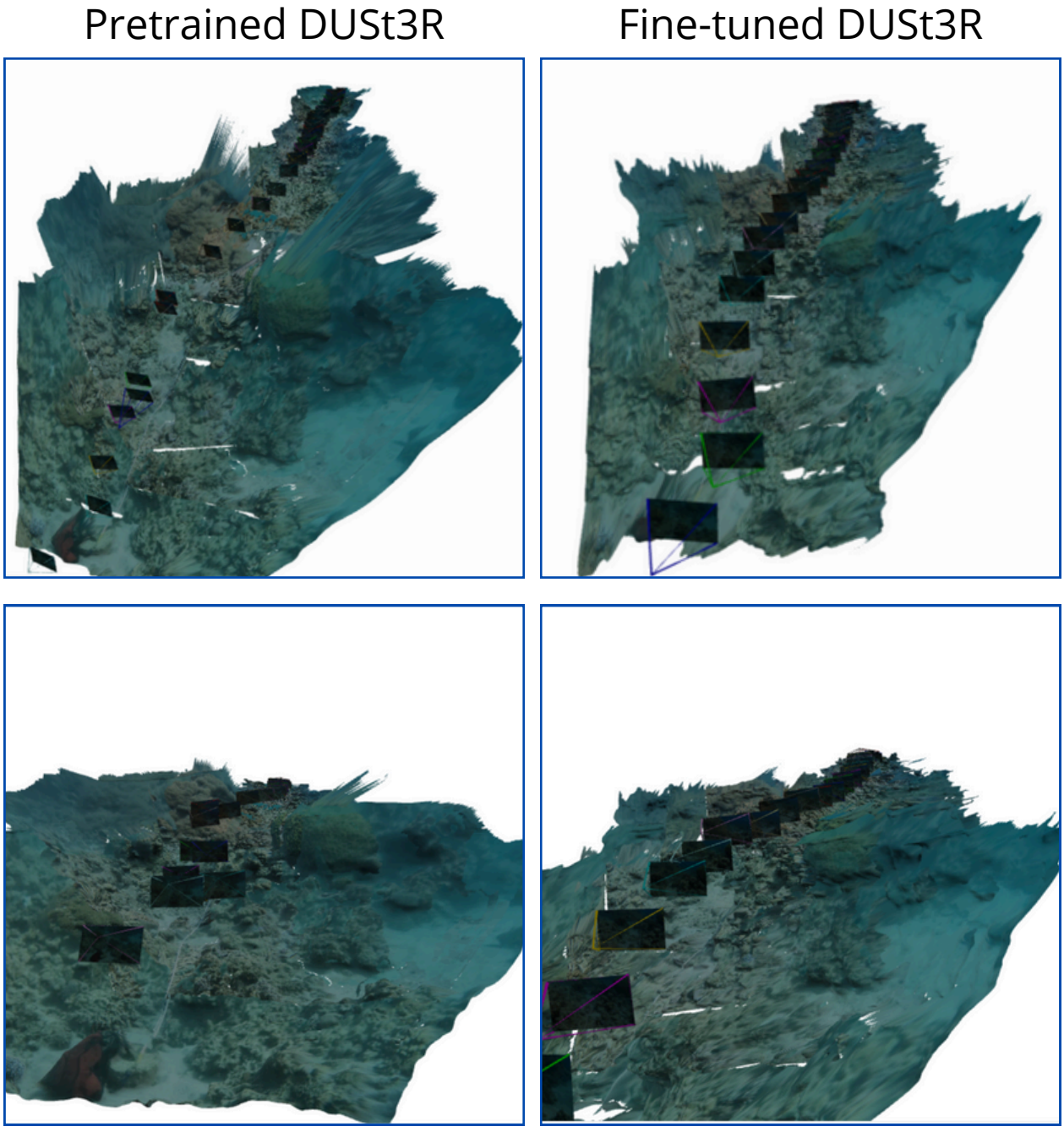

Fig. 15: **T_2-center-forward**: Results of the pretrained and fine-tuned DUSt3R models

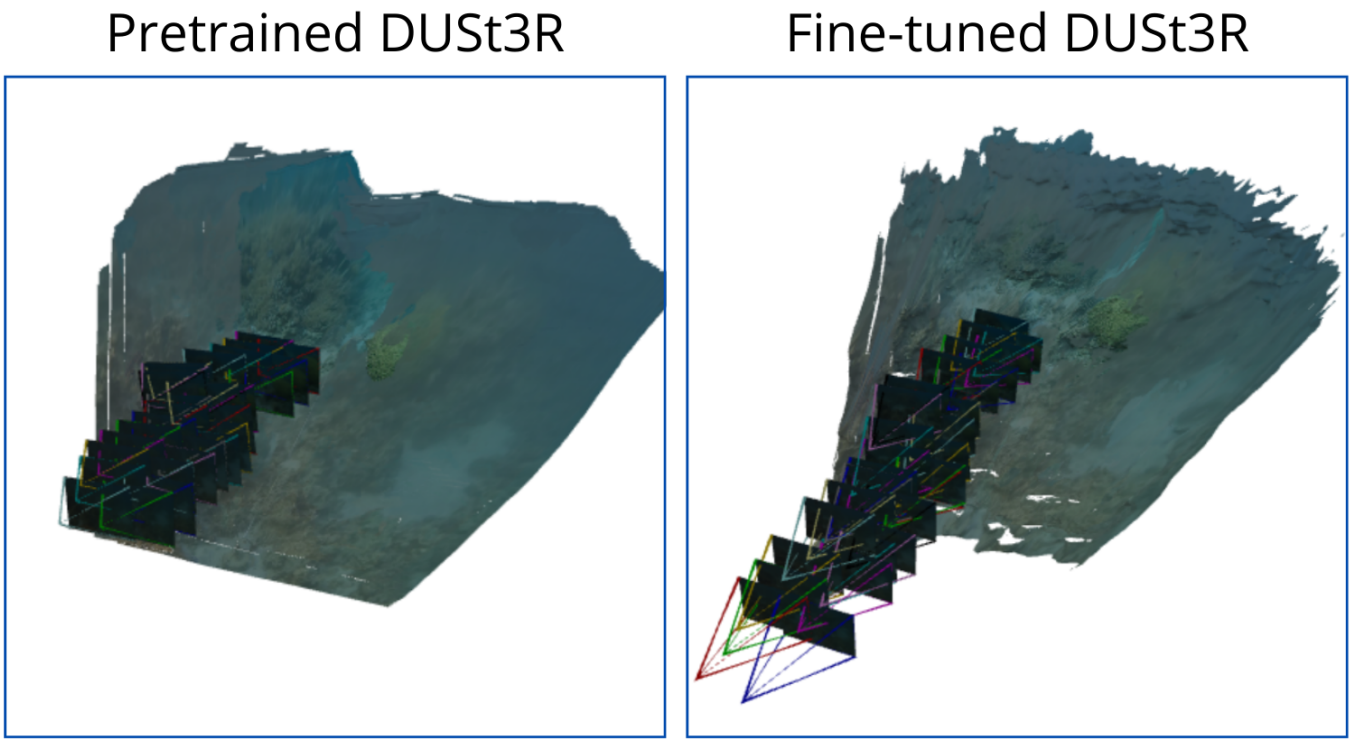

Fig. 16: **T_1-left-center-forward**: Results of the pretrained and fine-tuned DUSt3R models

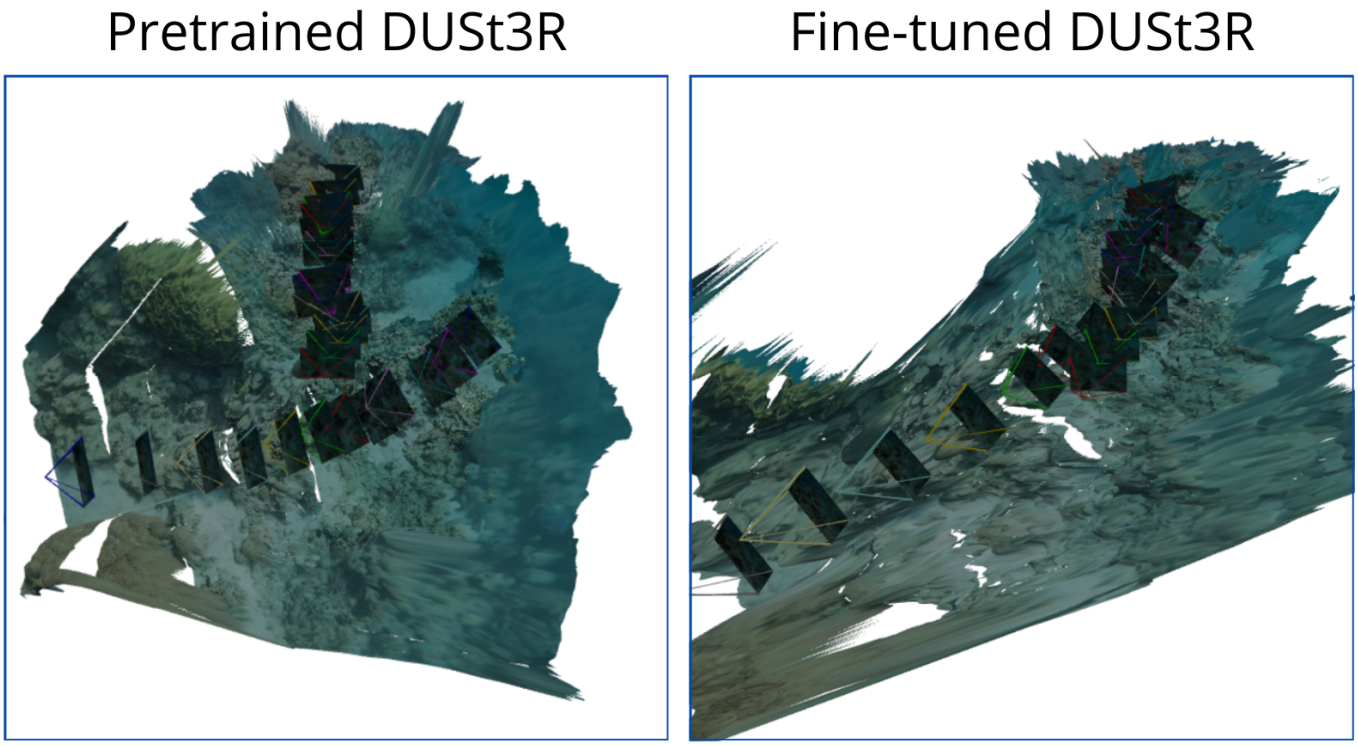

Fig. 17: **T_2-center-forward-backward**: Results of the pretrained and fine-tuned DUSt3R models

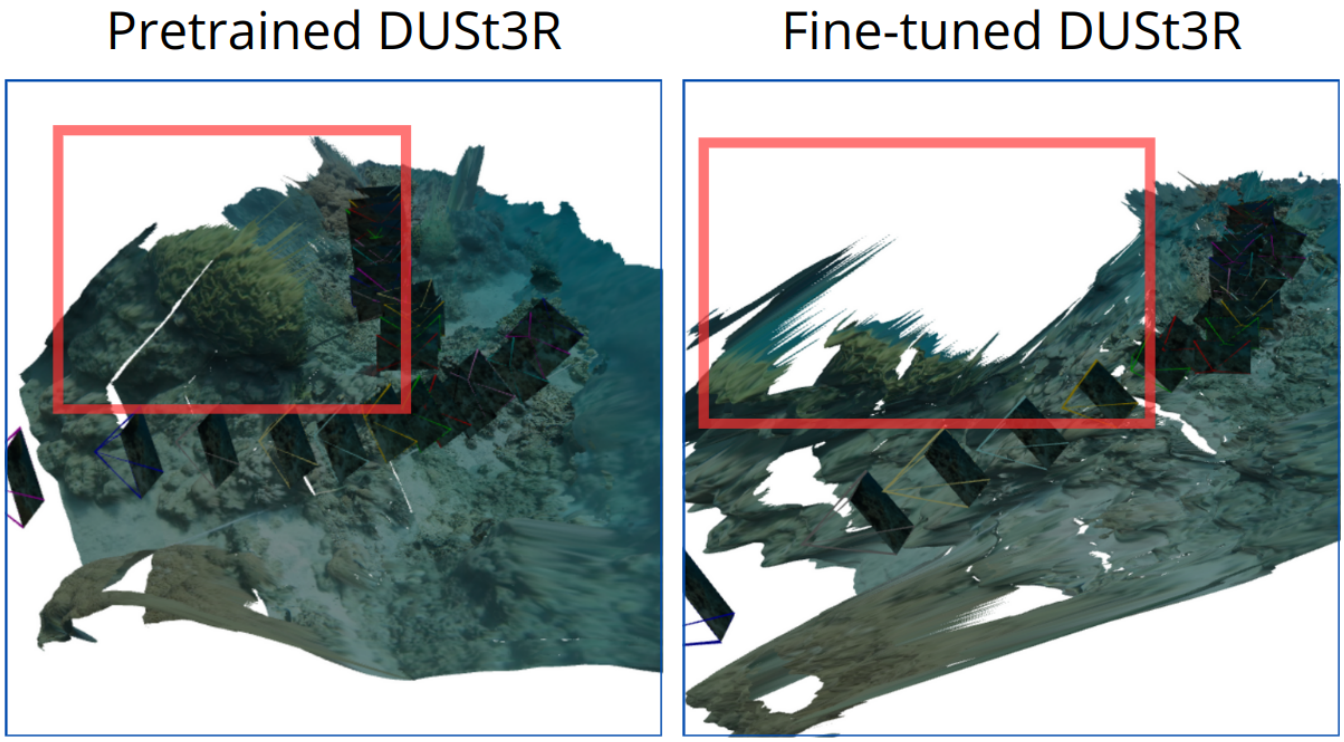

Fig. 18: **T_2-center-forward-backward**: Results of the pretrained and fine-tuned DUSt3R models with clear differences indicated

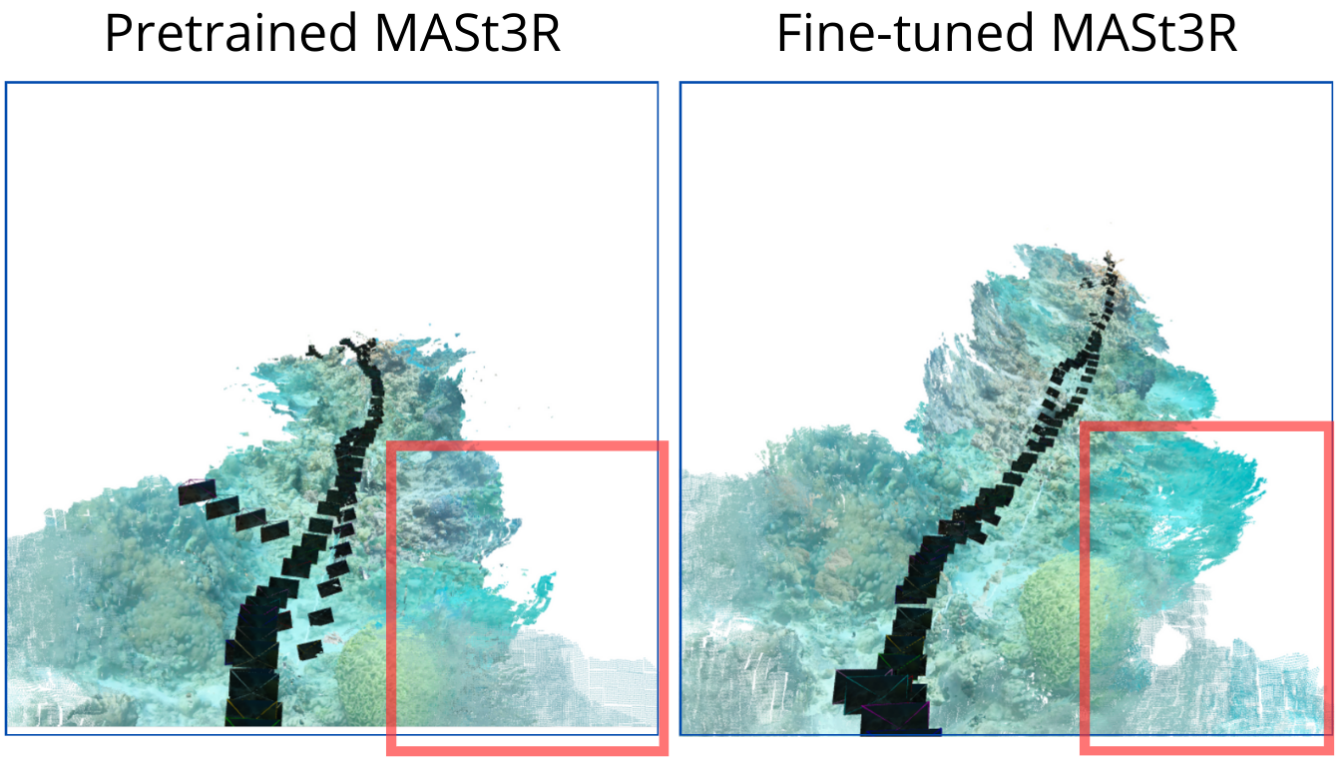

Fig. 19: **T_1-T_2-center-forward**: Results of the pretrained and fine-tuned MASt3R models with clear differences indicated

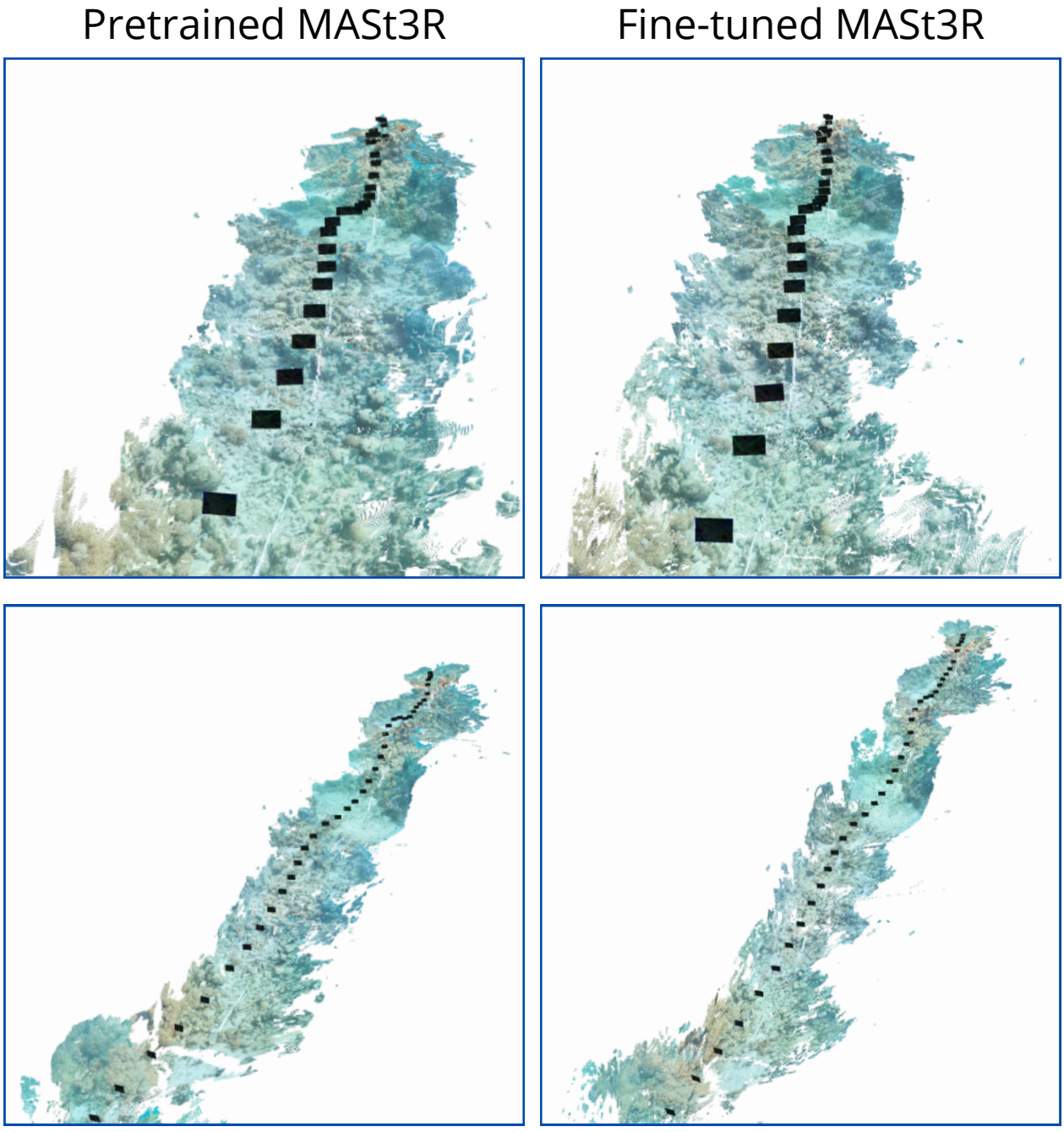

Fig. 20: **T_2-center-forward**: Results of the pretrained and fine-tuned MASt3R models

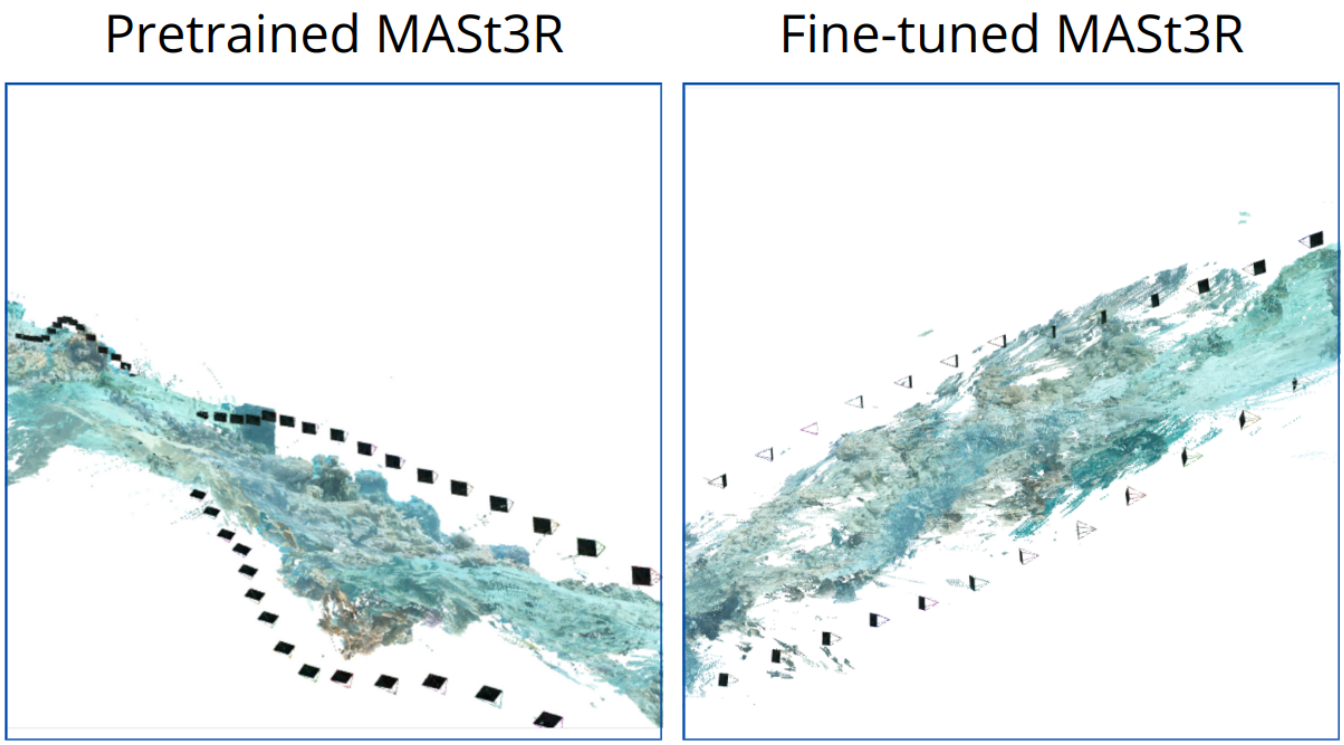

Fig. 21: **T_2-center-forward-backward**: Results of the pretrained and fine-tuned MASt3R models

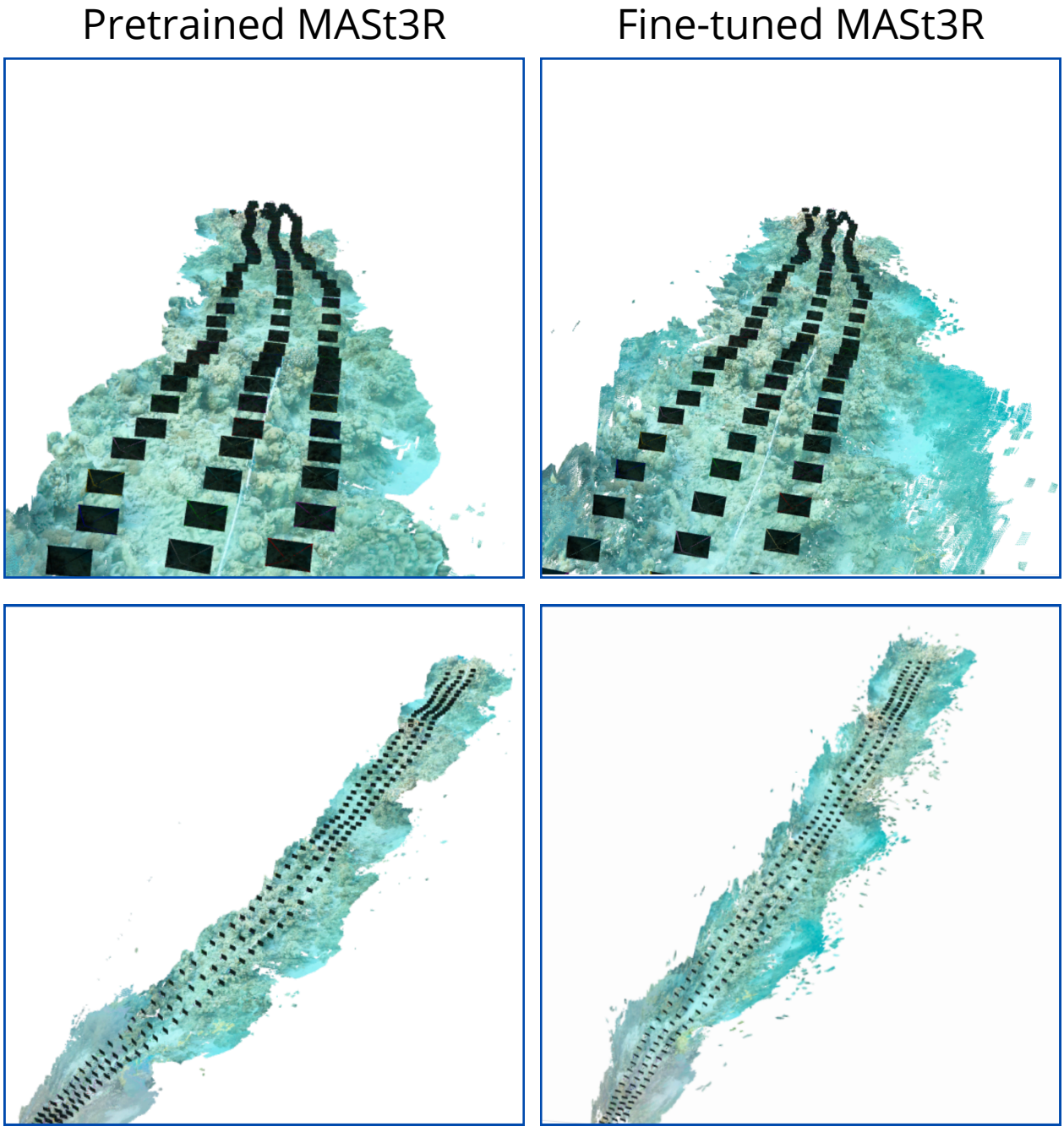

Fig. 22: **T_1-left-center-right-forward**: Results of the pretrained and fine-tuned MASt3R models