3D Reconstruction in the wild

Niranjan Thanikachalam
Stanford Center for Professional Development
Stanford University
niranjnt@stanford.edu

Abstract

Creating a digital replica of real world scenes and objects in 3D has many applications spanning digital archival, virtual tourism, extended reality or can simply serve as a better representation of the real world than 2D images. Several highly accurate solutions exist, most of which require a precise knowledge of camera parameters and scene illumination. In this project the problem of Phototourism - i.e recreating a 3D model of the real world from unstructured set of photographs is considered from a deep learning perspective. The work explores the use of deep learning components in the classical structure from motion pipeline. It also explores the replacement of the optimization component - bundle adjustment using the recently proposed DBARF, a generalized NERF inspired neural rendering method that simultaneously optimizes camera pose and image rendering. It is seen that while deep-learning components are in general successful in improving the feature description and matching stage, even neural rendering methods that "optimize" instead of learn, fail to achieve the accuracy of bundle adjustment.

1. Introduction

We consider the problem of reconstructing a scene in 3D given a collection of unstructered images obtained in the wild - i.e sourced from vastly different cameras, illuminations, angles etc.

Digitizing a real world object or scene in 3D is a fundamental problem in computer vision. Even if we restrict ourselves to vision only systems, an extremely wide variety of methods focusing on various scenarios exist and can be broadly classified into methods that include posed cameras with known illumination and those that use cameras or illumination at unknown locations.

1.1. Problem Statement

The goal of the project is to develop a framework which takes a set of unconstrained images as the input and estimates the camera extrinsics i.e $[\mathbf{R}|\mathbf{t}]$ - the rotation matrix and the translation vector of each camera in the image set. The problem is directly specified by the image matching challenge 2024[5]. As defined in the competition, the problem is different from traditional SfM in that, the dataset can be affected by the following aspects.

- Different sensor types and inconsistent occlusions.
- Night vs day temporal changes including poor lighting, different weather and climate conditions.
- Mixture of aerial and ground images.
- Repetetive structures.
- Highly non regular structures such foliage.
- Transparencies and reflections.

2. Related Work

When the camera intrinsics and extrinsics are known either fully or partially, the class of algorithms that are based on multi-view stereo[16] can reconstruct a scene in 3D with a high degree of accuracy. Image based methods like lightfields[26] produce dense representations of a scene by sampling the scene at enough known locations, without the need for generating parameterizations like meshes. More recently, neural rendering frameworks[11] have emerged, which leverage simple multilayer perceptrons as scene radiance and density approximators, which are then used to achieve photorealistic rendering of the scene through volume rendering.

Structure from motion (SfM) [12][15] methods on the other hand employ cameras at unknown locations of a scene. They then create a sparse representation - a point cloud of the scene, while simultaneously estimating the positions of the original cameras. In a typical SfM setup, while the camera positions are unknown, they are still consistent in illumination and camera intrinsic parameters, within the dataset being considered. Deep learning based methods have attempted at directly regressing on camera

locations[24][28][21][20], however have seen little success compared to traditional bundle adjustment based methods. DeepSFM[25] is a relatively successful method that is modelled after bundle adjustment - using a 2D CNN as a backbone for feature extraction, it then iteratively refines the depth and pose cost volumes by using a series of 3D convolutional layers for each cost branch. A bundle adjusting extension to NeRFs - BARF[7] poses a joint optimization problem per scene, where in addition to learning a coordinate based image representation, they also solve for the warp between image pairs i.e the camera pose transformations. Deep-BARF[2] extends this approach to Generalized NeRFs. NERF in the Wild [10] proposes modifications to the NERF photometric cost in order to account for transient objects and appearance variations, inorder to create NERFs from unstructured image sets.

A particularly difficult as well as interesting case of SfM is the case of using unstructured image collections as inputs [17]. Crowd sourced and internet datasets of frequently photographed scenes like landmarks can contain several thousand images of the same scene in different angles, different illuminations, and vastly different camera intrinsics. Inconsistent occlusions are another major hurdle to the application of vanialla SfM to this scenario. NeRF in the Wild [10] extend NeRFs by adding two additional MLPs, one that uses appearance embeddings to capture the variations in color due to illumination, sensor differences etc and another MLP that uses transient embeddings to account for inconsistent occlusions that exists in the image collection. Alternately, detector free methods[6] utilize "denser" detector free features[18] that are more robust to texture less surfaces to build a point cloud and estimate camera locations using a course to fine iterative refinement procedure.

3. Models

Traditional structure from motion involves three stages - feature detection and description, feature matching and outlier detection and finally reconstruction. First, for each image in a given scene, salient feature points like corners and descriptors that describe the area around these points are estimated. Then a list of image pairs that have an overlap is computed if scene contains too many images, else all combinations of image pairs are listed. Feature matches are then computed for the keypoints in each image pair using the nearest neighbours or other more sophisticated feature matching algorithms. These matches are filtered for outliers using RANSAC. The cleaned feature points and matches are then fed to COLMAP[15], which performs an incremental reconstruction by solving the classic bundle adjustment problem where the 3D reprojection error of the matched feature points is minimized by adjusting the camera positions and the 3D point positions. This results in the required camera poses.

For the baseline, we experimented with the following modern deep models for the feature detection and description :

- ALIKED: A Lighter Keypoint and Descriptor Extraction Network via Deformable Transformation [27]
- SuperPoint: Self-Supervised Interest Point Detection and Description [3]
- DISK: Learning local features with policy gradient [19]
- D2-Net: A Trainable CNN for Joint Detection and Description of Local Features [4]

For feature matching, we used a recent state of the art model - LightGlue: Local Feature Matching at Light Speed[9]

The proposed pipeline for structure from motion exploits two new paradigms introduced recently - detector free matching and bundle adjusting variants of neural rendering methods.

3.1. Detector Free Matching

Detector free matching methods like ASpanFormer[1], MatchFormer[23] and LoFTR[18] propose a new feature matching approach that removes the requirement for a separate feature detector phase and provide densely matched descriptors directly. These methods are particularly robust when dealing with images that have poor texture, repeatable patterns, scaling, varying illumination and viewpoint variations. We now provide a brief overview of LoFTR : Detector-Free Local Feature Matching with Transformers, which we choose to use in our SfM pipeline.

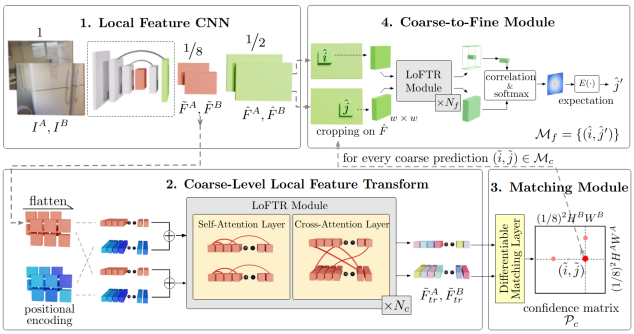


Figure 1: LoFTR Schematic, reproduced from the LoFTR paper.[18]

LoFTR has four sequential blocks. First, given a pair of images A,B, a Feature Pyramid Network (FPN) [8] generates a pair of feature maps at a course and fine level. In the second stage, the course level features are flattened, positionally encoded and then passed through a Local Feature Transformer Module, with self and cross attention. In-

stead of the conventional dot product attention, they propose the use of linear attention to reduce the computational complexity to O(N). The output of this block is then fed to a matching layer which establishes matches between the transformer output features in the coarse level. For each coarse level match, a pair of patches are extracted around the corresponding patches in the fine level feature. These are then accumulated and fed to a smaller LoFTR transformer module. The feature outputs of this stage are then correlated with each other, which is then used to produce a final match j with subpixel accuracy in B for a pixel i in A.

The use of detector free methods like LoFTR thus poses a unique challenge during bundle adjustment - feature points are defined between image pairs with sub-pixel accuracy on one image, and are therefore not shared across pairs. In SfM, feature matches are computed across multiple image pairs, which eventually need to be stacked to create "tracks". To solve for this, we pool the detections to a grid before redefining matches across these features across pairs.

During the course of this project Detector Free Structure From Motion [6] has been proposed with state of the art results. Their proposed approach is to also use LoFTR for matching, but they quantize and downsample the match locations to obtain matches on a coarser pair. These matches are then used with traditional SfM to get a coarse reconstruction. The second stage of their reconstruction pipeline involves iterative refinement of "feature tracks". At each iteration, they begin by using the current best reconstruction to create feature tracks, the patches in which are transformed using a second multi-view feature transformer. These transformed patches are then used to refine the feature track. These refined feature tracks are used in bundle adjustment to obtain a refined reconstruction.

3.2. Refining Bundle Adjustment Results

The final stage in Structure from Motion is the sparse reconstruction. This involves solving for the essential matrix to estimate relative camera positions between a pair of images to obtain a first sparse 3D model by triangulation of the points. New cameras are then added iteratively and at the end of each such step, bundle adjustment is carried out to jointly optimize for camera poses and recovered 3D points by minimizing the reprojection loss. Bundle adjustment is a notoriously large scale problem solved traditionally by using non-linear least squares optimization methods like Levenberg Marquardt.

The sparse nature of feature inputs to the reconstruction phase can be a source of errors since images with fewer matches no longer contribute to the reconstruction. Infact this is the inspiration behind the proposal to use semi-dense matches for SfM in the first proposed modification.

Recently in the counterpart to SfM - reconstruction with posed cameras, Neural rendering methods [11] have shown immense success in achieving photorealistic rendering of arbitrary scenes using posed cameras. They are implicitly dense reconstructors. We would like to investigate if the bundle adjusting variants of neural rendering can be used to further refine the pose estimation outputs of traditional structure from motion. BARF - Bundle Adjusting Radiance Fields[7] were an ideal starting point since they are optimized per scene resulting in the neural network equivalent of a joint pose optimizer and dense reconstruction. However due to limitations in computational capacity and time frame, I decided to investigate a generalized version of BARF, which can be fine tuned per scene.

We briefly present the core idea behind D-BARF: Deep Bundle Adjusting Radiance Fields[2]. DBARF takes a set of images and a a scene graph as its input. Initial camera poses are also an optional input. Image features are first extracted by a FPN[8] backbone. A feature metric consistency cost is constructed based on these features for feature consistency across the target and nearby views.

$$\mathcal{C} = \frac{1}{\mathcal{N}(i)} \sum_{j \in \mathcal{N}(i)} ||\mathcal{X}(\mathbf{K}_j \mathbf{P_{ij}} \mathbf{X}_{\mathcal{S}(\mathbf{u_i})}, \mathbf{F}_j) - \mathcal{X}(\mathcal{S}(\mathbf{u}_i), \mathbf{F}_i)|| \tag{1}$$

where, $\mathbf{K}_j, \mathbf{P}_{ij}$ are the camera intrinsics and relative pose matrix respectively. $\mathbf{X}_{\mathcal{S}(\mathbf{u}_i)}$ is the projection of the patch of 3d points which is computed from the predicted depth map $\mathbf{D}$ for the target image. $\mathbf{F}_j, \mathbf{F}_i$ represent the feature maps, $\mathcal{X}$ is the interpolation function, and $\mathcal{N}$ is the set of nearby views obtained from the scene graph. DBARF proposes the use of a recurrent GRU block based architecture for the depth and pose estimators which are used to update the proposed cost map. For the deep generalizable NERF, DBARF extends IBRNet Learning Multi-View Image-Based Rendering[22] which is optimized on a photometric consistency loss function. The depth network is optimized on a modified version of photometric loss.

4. Dataset

The dataset consists of unconstrained image collections of seven scenes. For each image in each collection, the corresponding camera extrinsic parameters has also been provided. In addition to this, a sparse representation of the scene as a point cloud is also provided that can be used by the popular SfM library COLMAP[15]. The number of available images in each scene is listed in Table 1.

Due to the nature of the problem at hand, this relatively small dataset cannot be used for training - neither from scratch nor for fine tuning. For this reason, I will use this dataset of seven scenes as my validation set, for which I am able to compute the mean Average Accuracy metric defined in the previous section.

For Test data, the competition provides a hidden/blind

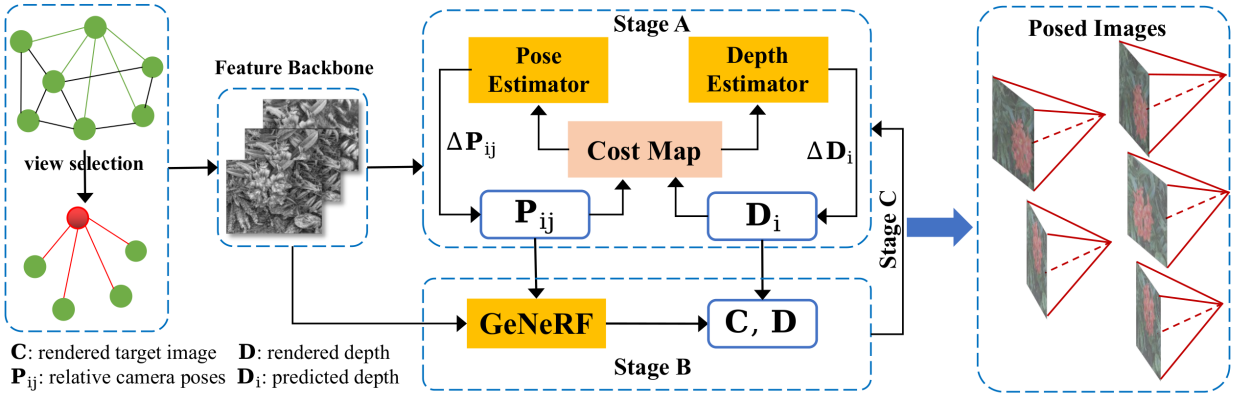


Figure 2: Network Architecture of D-BARF, reproduced from [2]

Name	#images	Sample
Church	111	
Dioscuri	70	
Lizard	727	
Multi-temporal-temple-baalshamin	75	
Pond	1148	
transp obj glass cup	36	
transp obj glass cylinder	36	

Table 1: Validation Dataset Summary

test, which can only be tested against by providing a submission to the competition.

5. Results and Experiments

5.1. Detector Free Matching

The accuracy of reconstruction of the baseline - feature detector + feature matching methods as well as the LoFTR based method and DFSFM are first evaluated using a custom metric defined by the Image Matching Challenge [5] and is refered as mean Average Accuracy (mAA) of camera centers, $\mathbf{C} = -\mathbf{R}^T\mathbf{t}$. Given the ground truth and the estimated $\mathbf{R}|\mathbf{t}$ values, the best similarity transformation $\mathcal{T}$ is first computed. A camera is considered to be registered if $||\mathbf{C}_g - \mathcal{T}(\mathbf{C})||$ where $\mathbf{C}_g$ is the ground truth camera center and $\mathbf{C}$ the estimated center. Assuming r_i percent of cameras in the scene that are registered by $\mathcal{T}_i$ the mAA is computed by averaging r_i for several thresholds t_i which are scene dependent and set by the competition.

In Table 2 we list the scenewise mAA scores for all the methods considered. The camera poses are the result of the optimization based sparse reconstruction stage and do not include the DBARF refinement proposed in the previous section. All DFSFM reconstructions underwent three rounds of iterative refinement of feature tracks and geometry. The two transparent object scenes as seen on 1 are rather unusual in that, camera moves around the object in a perfect circle, while the optical axis of the camera is always pointed towards the object. Given the cylindrical nature of the objects, the circular revolution of the camera around the axis of the cylinder and the absence of texture, they account for a very difficult dataset, that none of these methods perform well against.

The Dioscuri dataset has a lot of rotated images. This seems to adversely affect the number of inliers in the feature matches obtained with LoFTR since both detector less strategies perform poorly where the feature detector based methods perform better. Figure 3 shows the reconstructions from DFSFM as well as the best performing model for this

Dataset	ALIKED + LG	DISK + LG	Superpoint + LG	DFSFM	LoFTR
Church	0.2975	0.2897	0.1698	0.200935	0.2445
Dioscuri	0.1692	0.3010	0.1542	0.057214	0.0174
Lizard	0.8039	0.6950	0.7698	0.612245	0.6213
Multi-temporal-temple-baalshamin	0.2169	0.2540	0.2169	0.457672	0.1667
Pond	0.6497	0.5034	0.3810	0.416100	0.3662
transp obj glass cup	0.0152	0.0354	0.0152	0.00000	0.0000
transp obj glass cylinder	0.0303	0.0000	0.0000	0.00000	0.0000
Hidden test scene	0.13	0.13	0.11	NA	NA

Table 2: mean Average Accuracy: mAA for all the test scenes for all the proposed methods. The does not include the DBARF based refining strategy proposed in the previous section.

dataset - DISK + Lightglue.

Also of interest is the Multi-temporal-temple data, which has several images obtained with illumination variations. While DFSFM is the best performing model for this dataset, the other detector free method - LoFTR performs poorly. As seen in figure 5, the coarse reconstruction with DFSFM already outperfoms LoFTR interms of reconstructed matches. This could be attributed to the increased robustness obtained by quantizing the feature points into a coarser grid. The improved reconstruction in the DFSFM refined is also evidence of the improvement in performance attributed to feature track refinement.

Figure 4 shows an example pair from both these datasets with features matched with ALIKED + LightGlue and LoFTR

It is to be noted that except for the multi-temporal-temple dataset, the two detector free methods perform comparably on the remaining datasets, with the simple LoFTR method even outperforming the DFSFM pipeline in the Church and Lizard dataset by a small factor. Figure 6 shows the reconstructions from DFSFM pipeline for all the remaining datasets.

5.2. DBARF based pose refinement

DBARF assumes that all the images are obtained from a single camera. The pond dataset seems to be a SLAM like dataset obtained from a single vertical free moving camera. No other non-transparent dataset satisfies the single camera criterion. We only considered the pond dataset while attempting to finetune DBARF. For pose refinement with DBARF, we first used Hierarchical Localization [14] to prepare the scene graph. Followed by this, with the initial camera positions set to the output of ALIKED + LG (the best performing model for this dataset), we attempted finetuning the pretrained model. In figure 7, the loss function values and the total absolute error on the rotation matrix are plotted when the model was trained with a learning rate of 1e-4 for the features network and 2e-4 for the MLP for 30000 steps. The pretrained model checkpoint was already at 200000 steps. As can be seen, the model never began converging. We attempted training with several different LRs but with little success.

While it is possible that there is an error in my procedure, it is also to be observed that the generalized NERF model in DBARF was pretrained mainly on indoor scenes or on scenes with fronto-parallel camera motion. The model was validated with similar scenes as well. This might mean that the pretrained model does not generalize well with other outdoor data as seen in our case.

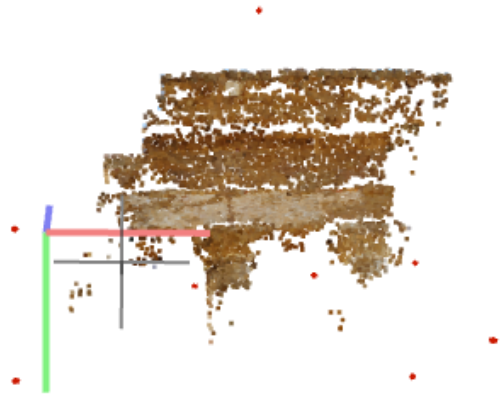
The DBARF model is complex and has a rather non-sequential computational graph. It is also possible that the finetuning of the model is sensitive to the hyperparameters.

6. Discussion

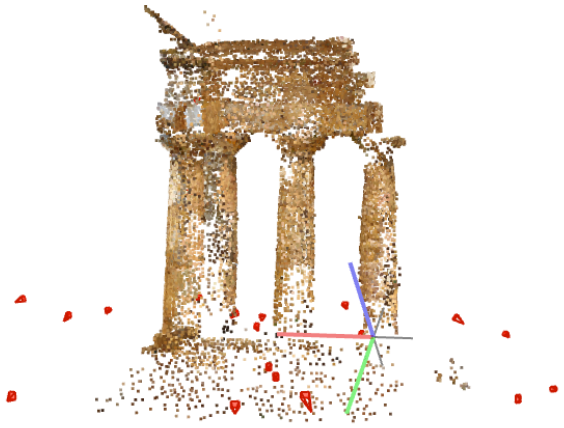
For the course project, we considered an end to end pipeline for SfM and explored modules in which deep learning methods can be utilized. The fact that several other methods have already been proposed [28][24][20][21] where a deep network attempts to directly regress on the SfM outputs with limited success dissuaded me from pursuing a similar approach. Given the success of "optimizer" like networks such as NERF and RAFT from which DBARF heavily borrows, we invested a lot of time into fine tuning DBARF for our data with little success. I briefly also tried BARF, although BARF was computationally too expensive for the time frame of this project.

We briefly also tried formulating the sparse reconstruction optimization using pytorch with the hope of using ADAM like optimizers to solve for bundle adjustment. However I soon realized this was also a much more demanding effort.

Finally, D2Net[4] is a feature detector that was specifically trained on the Aachen Day-Night dataset. Inorder to use it in a way comparable to the other feature detectors, I



(a) DFSFM : 13641 pts

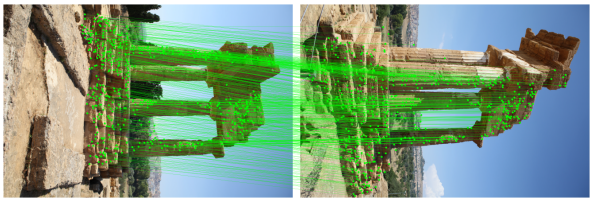


(b) DISK + LightGlue : 16985 pts

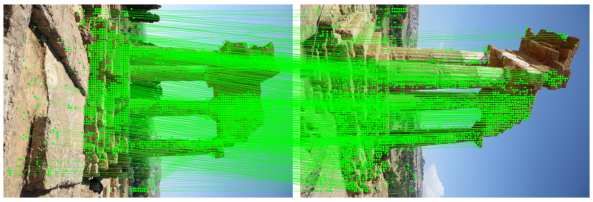
Figure 3: Comparison of the Dioscuri reconstruction between DISK + LightGlue and DFSFM. It is to be noted that this dataset had a lot of flipped and rotated images, which affected the accuracy of matching with LoFTR and consequently the DFSFM pipeline

also started training the LightGlue feature matcher using the recently proposes Glue Factory[13]. However, at the time of the deadline, the pretraining of the lightglue matcher with D2Net was still ongoing with the homography dataset. The training of the Megadepth dataset was yet to commence.

Given a longer time frame or a bigger frame, we might have pivoted to the above project or would have pursued the implementation of the PyTorch optimizer for sparse implementation with more confidence.

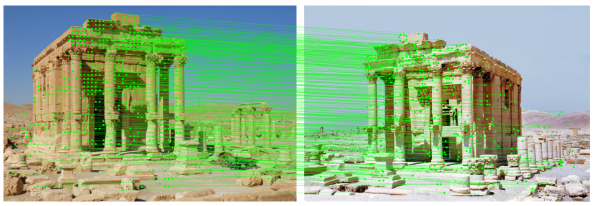


(a) ALIKED + LightGlue in Dioscuri dataset



(b) LoFTR in Dioscuri dataset

(c) ALIKED + LightGlue in Temple dataset



(d) LoFTR in the Temple dataset

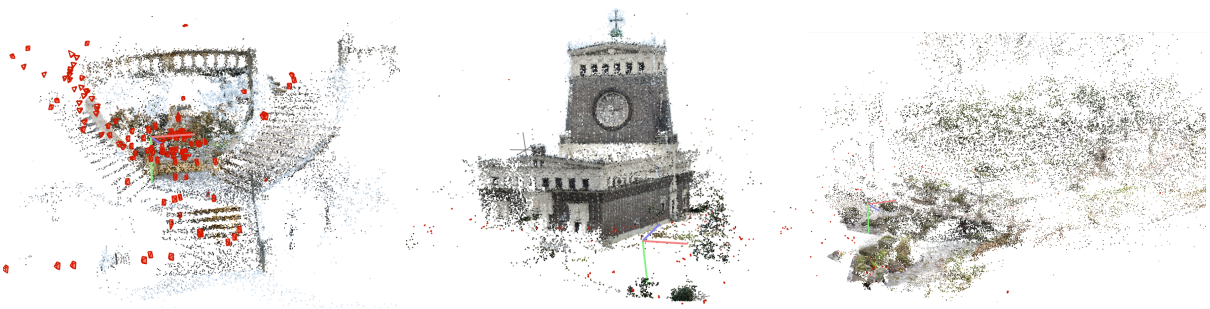
Figure 4: Comparison of matched feature points using ALIKED + LightGlue in two datasets with different peculiarities. The Dioscuri dataset includes a lot of rotations and flips of the image which seem to adversely affect the number of inliers in the LoFTR match as compared to the ALIKED + LightGlue matching. The temple dataset has images with temporal illumination changes. The LoFTR based DFSFM method is twice as effective as feature point + detector based methods although the direct LoFTR method is much worse.

7. Conclusion

We explored the use of deep learning models in a modern SfM pipeline. We proposed the use of ”denser” models for a better reconstruction. With this inspiration, detector less methods that utilize semi-dense matches were explored. Further more DBARF, a bundle adjusting variant of IBRNet, which is nerual rendering model was explored as a final pose refinement optimizer with no success.

(a) LoFTR : 48246 pts (b) DFSFM Coarse : 175195 pts (c) DFSFM Refined : 127183 pts

Figure 5: Comparison of the reconstruction of the two detector free methods for the Multi-temporal-temple-baalshamin dataset. Note that the reconstruction accuracy is comparable between LoFTR and DFSFM in every other datasets, with LoFTR even slightly outperforming in Church and Lizard Dataset. With the temple dataset however, LoFTR performs poorly. It is useful to note that the temple dataset contains multi-temporal data, which possibly introduces a lot of outliers. The pooling and downsampling of keypoints in DFSFM during the coarse step already seems to yield a better reconstruction than the simple quantization of keypoints in LoFTR.



(a) Lizard : 238099 pts (b) Church : 267767 pts (c) Pond : 135937 pts

Figure 6: Sparse Reconstructions using the DFSFM pipeline

References

[1] H. Chen, Z. Luo, L. Zhou, Y. Tian, M. Zhen, T. Fang, D. Mckinnon, Y. Tsin, and L. Quan. Aspanformer: Detector-free image matching with adaptive span transformer, 2022.

[2] Y. Chen and G. H. Lee. Dbarf: Deep bundle-adjusting generalizable neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24–34, June 2023.

[3] D. DeTone, T. Malisiewicz, and A. Rabinovich. Superpoint: Self-supervised interest point detection and description, 2018.

[4] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and T. Sattler. D2-net: A trainable cnn for joint detection and description of local features, 2019.

[5] D. M. E. T. Fabio Bellavia, Jiri Matas. Image matching challenge 2024 - hexathlon, 2024.

[6] X. He, J. Sun, Y. Wang, S. Peng, Q. Huang, H. Bao, and X. Zhou. Detector-free structure from motion. *CVPR*, 2024.

[7] C.-H. Lin, W.-C. Ma, A. Torralba, and S. Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5741–5751, 2021.

[8] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection, 2017.

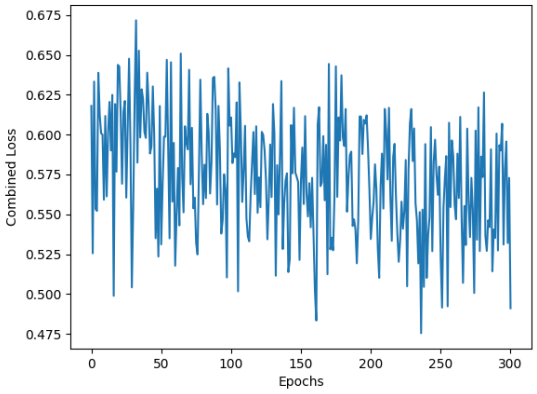
[9] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys. Lightglue: Local feature matching at light speed, 2023.

[10] R. Martin-Brualla, N. Radwan, M. S. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7210–7219, 2021.

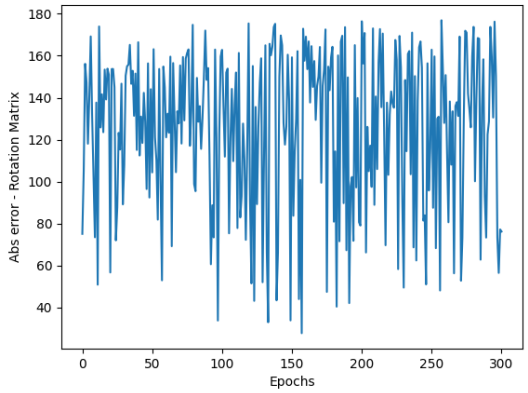
[11] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[12] O. Ozyesil, V. Voroninski, R. Basri, and A. Singer. A survey of structure from motion, 2017.

[13] R. Pautrat*, I. Suárez*, Y. Yu, M. Pollefeys, and V. Larsson. GlueStick: Robust Image Matching by Sticking Points and



(a) Combined Loss during finetuning DBARF on the Pond Dataset, using a pretrained checkpoint. It failed to converge with several variations of learning rate and batch size



(b) Absolute error during finetuning DBARF on the Pond Dataset, using a pretrained checkpoint.

Figure 7: Finetuning of the DBARF model on the pond dataset failed despite a hyperparameter sweep

Lines Together. In *International Conference on Computer Vision (ICCV)*, 2023.

[14] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *CVPR*, 2019.

[15] J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[16] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, volume 1, pages 519–528. IEEE, 2006.

[17] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006.

[18] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou. LoFTR: Detector-free local feature matching with transformers. *CVPR*, 2021.

[19] M. J. Tyszkiewicz, P. Fua, and E. Trulls. Disk: Learning local features with policy gradient, 2020.

[20] B. Ummenhofer, H. Zhou, J. Uhrig, N. Mayer, E. Ilg, A. Dosovitskiy, and T. Brox. Demon: Depth and motion network for learning monocular stereo. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, July 2017.

[21] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki. Sfm-net: Learning of structure and motion from video, 2017.

[22] Q. Wang, Z. Wang, K. Genova, P. Srinivasan, H. Zhou, J. T. Barron, R. Martin-Brualla, N. Snavely, and T. Funkhouser. Ibrnet: Learning multi-view image-based rendering, 2021.

[23] Q. Wang, J. Zhang, K. Yang, K. Peng, and R. Stiefelhagen. Matchformer: Interleaving attention in transformers for feature matching, 2022.

[24] S. Wang, R. Clark, H. Wen, and N. Trigoni. Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2017.

[25] X. Wei, Y. Zhang, Z. Li, Y. Fu, and X. Xue. Deepsfm: Structure from motion via deep bundle adjustment, 2020.

[26] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu. Light field image processing: An overview. *IEEE Journal of Selected Topics in Signal Processing*, 11(7):926–954, 2017.

[27] X. Zhao, X. Wu, W. Chen, P. C. Y. Chen, Q. Xu, and Z. Li. Aliked: A lighter keypoint and descriptor extraction network via deformable transformation, 2023.

[28] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video, 2017.