# Final Project: Fundamental Matrix Estimation For Out Door Tourism Scenes

Asya Olshansky

February 2025

## 1 Introduction

Feature matching is crucial in computer vision for tasks like structure-from-motion, visual localization, and image stitching. Traditionally, methods fall into two categories: sparse and dense feature matching. Sparse and dense matching techniques each have distinct advantages depending on the application requirements. In this document, we compare both approaches with a focus on three modern methods: D2Net [3], DKM [4] and RoMa feature matching [9].

Sparse feature matching relies on detecting and describing keypoints in an image. These keypoints are then matched across images. Popular traditional methods include SIFT, SURF, and ORB. D2Net [3] is a modern approach that integrates deep learning for feature detection and description. Other notable CNN-based sparse feature extractors include SuperPoint [2] and R2D2 [8], which offer improved robustness and repeatability.

Dense matching, in contrast, attempts to find correspondences for every pixel in an image. This can be beneficial for textureless regions and occlusions. RoMa and DKM are examples of deep-learning-based dense matching techniques that improve robustness in such scenarios. Other CNN-based dense matching methods include RAFT (Recurrent All-Pairs Field Transforms) and PDC-Net, which propose deep neural networks for pixel-wise matching.

| **Method** | **Type** | **Strengths** | **Limitations** |
|---|---|---|---|
| D2Net | Sparse | Robust deep features | Sensitive to scale changes |
| SuperPoint | Sparse | Efficient, self-supervised | Requires good keypoint coverage |
| R2D2 | Sparse | Robust repeatability | High computational cost |
| ALIKED | Sparse | Lightweight, efficient | Limited training data |
| RoMa | Dense | Robust to challenging textures | Computationally expensive |
| DKM | Dense | High accuracy, generalizable | High memory usage |
| RAFT | Dense | Efficient optical flow estimation | Requires large datasets |
| PDC-Net | Dense | Strong local matching | Sensitive to lighting changes |

Table 1: Comparison of CNN-based sparse and dense feature matching methods.

Sparse and dense matching techniques serve different purposes. D2Net, SuperPoint, and R2D2 enhance traditional sparse matching with deep learning, while RoMa, DKM, RAFT, and PDC-Net offer improved robustness in dense matching scenarios. The choice of method depends on the application, balancing accuracy, efficiency, and robustness.

## 2 The Project Experiments

In this project, I explored two networks, ALIKED and RoMa. However, the ALIKED GitHub repository lacks training code, and its test errors were significantly lower compared to RoMa. However, the ALIKED GitHub repository lacks training code, and its test errors were significantly lower than those of RoMa. Therefore, I focus solely on RoMa in the following discussion.

## 3 RoMa Method

RoMa is a dense feature matching method designed for robust correspondence estimation in challenging outdoor scenes. Unlike traditional sparse keypoint-based methods, RoMa computes dense feature maps, making it highly effective for scenes with large variations in illumination, viewpoint, scale, and orientation.

RoMa uses a frozen DINOv2 backbone to extract high-level image features. Additionally, it integrates VGG and ResNet50 (potentially frozen) with trainable layers on top to enhance feature representation. The trainable layers allow RoMa to adapt to the specific requirements of dense feature matching while maintaining the generalization capabilities of the frozen backbones.

Contrastive loss is used to ensure that corresponding features across images are similar while non-corresponding ones are distinct. Cycle consistency loss enforces robust matching by ensuring that feature correspondences are bidirectionally consistent. Geometric consistency loss improves the alignment of matched features with real-world scene structures, making the method reliable for large-scale visual localization and mapping.

RoMa is trained on a diverse set of outdoor scenes, including urban and natural environments, which ensures generalizability. Its design makes it highly robust to illumination changes, as it learns structural rather than color-dependent features. The multi-scale representation helps in dealing with varying object and scene sizes. Rotation and orientation robustness is achieved by incorporating spatially invariant features, making it effective for images captured from different angles.

Due to its dense matching approach, RoMa excels in 3D reconstruction, visual localization, and Structure-from-Motion (SfM). Compared to traditional sparse feature matchers, RoMa achieves higher accuracy, robustness, and reliability, especially in challenging tourism scenes with dynamic lighting and occlusions.

Overall, RoMa significantly improves feature matching in outdoor environments, making it a valuable tool for computer vision tasks requiring high-quality correspondence estimation.

I chose to focus on this model because our matching project involved images captured under varying illumination, weather conditions, and scales.

## 4 RoMa Network Refinement

RoMa is trained using dense matches. The supervised warps are derived from dense depth maps from multi-view-stereo (MVS) of SfM reconstructions in the case of MegaDepth, and from RGB-D for ScanNet.

To incorporate our training data into the process, it needed to follow the same procedure. However, due to resource constraints, using COLMAP was not feasible, so I explored an alternative approach. First, I manually marked four prominent points on each image pair and utilized DPT [6] or MiDaS [7] networks to obtain dense depth estimates.

Both networks output inverse depth, whose inverse provides relative depth. Using the exact matches along with intrinsic and extrinsic parameters, I estimated highly accurate depth using triangulation method. I then aligned the obtained relative depth to the exact one using a linear mapping:

$$\hat{d} = \frac{a}{d_{\text{inv}} + \epsilon} + b, \tag{1}$$

The estimation is based on matching estimated CNN depth with the real depth at the marked points by minimizing the mean squared error (MSE). According to literature this is the way the DPT and MiDaS ground truth is generated.

Subsequently, the manually selected points were replaced with the best-matching pairs identified using the SIFT algorithm, as we studied in class. However, my findings indicate that this cost-effective method for obtaining ground truth depth does not yield sufficiently accurate results. I provide code showing the entire process of creation ground truth for RoMa and obtained results. Examples of this process using DPT depth are presented in Figures 1-3. Similar results using MiDaS are presented in Figures 4-6. Given the suboptimal depth estimation results and limited computational resources, I decided to abandon this approach and instead focus on improving RoMa inference.

The code to perform ground truth dense matching based on this approach is in my github [5].

# 5 Improving Fundamental Matrix Estimation

In the paper [9], MegaDepth and ScanNet datasets sets are used to train the RoMa network and evaluation is performed on the Wide Baseline Stereo (WxBS) dataset. This dataset is known for its challenging image pairs that exhibit significant variations in scale, illumination, viewpoint, and texture. The WxBS dataset is designed to test the robustness of feature matching algorithms under extreme conditions. It is also well-suited for handling different scales of images. Usage of pretrained DINOv2 is trained on large-scale datasets and inherently capture multi-scale information. Extreme scale variations (e.g., a small object in one image appearing much larger in another) can still present challenges. For multiple extreme scales further fine-tuning or explicit scale-aware adaptations (e.g., multi-resolution training) may be needed.

# 6 Multiple Estimations of Fundamental Matrices

The idea is to run the network multiple times by additionally cropping patches from one of a pair of images. The coordinates of matches corresponding to image patches are returned back to the original image coordinates. Matching patches along with full-image features can improve fundamental matrix estimation by enhancing local correspondence robustness, especially in textureless or repetitive regions. Integrating RoMa with patch-based matching could refine epipolar geometry, reducing outliers and improving accuracy. The estimated matches and fundamental matrices $F_i,\ i = 1, \dots N$, where $N$ is the number of combinations tried (number of patches and original image-to-image match), can be used differently:

1. Merging all matches obtained and estimating $F$ once with a large number of matches.
2. Selecting the best $F_i$ estimation, the winner takes all approach.
3. Averaging fundamental matrices.

## 6.1 Merging All Matches

Combining matches from all combinations (global + patch-level) before running RANSAC once can improve robustness by increasing inlier density. However, the possible drawbacks are listed below:

1. More Outliers – Patch-level matches might introduce noise.
2. Computational Overhead – More matches = higher RANSAC cost.
3. Scale Variations – Global and local matches may have different error distributions.

One of the ideas explored was to combine matches (filtering or weighing) before RANSAC to prioritize high-confidence correspondences.

## 6.2 Potential Benefits of Averaging from Patches

Below we discuss why patching can help in better $F$ estimation.

**Robustness to Local Variations** Different image patches may contain different structures, reducing bias from dominant patterns (e.g., repeated textures, dominant planes). By averaging, multiple perspectives are incorporated, reducing noise.

**Improved Outlier Resistance** Some patches may contain outlier matches (e.g., due to occlusions or distortions), and averaging can help smooth their impact.

**Increased Correspondence Density** Some patches may have better-conditioned feature matches, while others may not. Averaging integrates multiple reliable local estimates.

### 6.3 Potential Drawbacks and Challenges

Below we discuss why patching can harm in better $F$ estimation.

**Patch Misalignment Issues** If patches are too small or contain features on different depth planes, estimated fundamental matrices may not be globally consistent. If fundamental matrices vary too much between patches, their average may not be meaningful.

**Nonlinear Constraints on** $F$ The epipolar constraint $x'^T Fx = 0$ is nonlinear. Simply averaging $F$ matrices does not guarantee that the resulting matrix satisfies fundamental matrix constraints.

**Noise Accumulation** If some patches produce bad estimates (e.g., due to poor feature distribution), averaging may still carry their error.

### 6.4 Weighed Average with rank-2 enforcement

In order to overcome these difficulties, the proposed approach is using weighed average. Instead of simple averaging, weight each $F$ based on how well it fits its own matches (e.g., using the number of inliers from RANSAC):

$$\bar{F} = \sum_i w_i F_i, \quad \text{where } w_i = \frac{1}{\text{reprojection error of } F_i} \tag{2}$$

Simple or weighed averaging is also non-valid due to final $F$ requirement to be rank-2. This is not guaranteed due to null space of $F_i$ being different. The trick is applying SVD correction after averaging. The SVD of $\bar{F}$ is computed, and the smallest singular value is set to zero, and $F$ is reconstructed back.

### 6.5 Bidirectional Approach

One of the tried ideas was considering matching in both directions. Assuming that matches are ideally perfect in both directions: $x'Fx^t = 0$ can be transformed and written as $xF^t x'^t = 0$. This means that in the opposite direction the ideal estimated $F$ matrix should be transpose of the one in estimated in the first direction. In other words, we can estimate the average of the $F_1$ and $F_2^t$. Similar to Section 6.4 the averaging should be weighed and follow the same rank-2 enforcment.

Another approach is usage of matching points twice similar to "merging all matches" approacch and estimating $F$ once with a large number of matches. Keypoints in the opposite direction are switched directions before merging.

### 6.6 Histogram Equalization Approach

The histograms equalization approach, while seems obvious does not work well as introduces significant artifacts. It did not work well in my experiments.

## 7 Experiments

The Table 2 summarizes all experiments that were conducted to improve RoMa inference. The files to generate ground truth dense matxhing between images based on depth appear in github []. The driving code is in texttt demo_depth.py

I used the conventional models available from public sites such as

```
vgg19_bn-c79401a0.pth from pytorch hub
dinov2_vit14_pretrain.pth from pytorch hub
roma_outdoor.pth from RoMA github
```

I also provide herein RoMa github location used by me [1].

| **File Name** 1 | **Method** |
|---|---|
| my-inference.py 1 | basic RoMa |
| my-allpoints-inference.py | all points merged (Section 6.1) |
| my-ave-inference.py | averaged estimated $F_i$ Section 6.2) |
| my-winner-inference.py | winner takes all approach (Section 6), 2. |

Table 2: All considered experiments. The first column is a driving file name generating submission.csv file. Second column refers to the method used with references to the method discussed

# 8 Conclusion

RoMa is a highly robust algorithm capable of handling challenging scenes effectively. It works seamlessly out of the box. Resource constraints prevented me from testing all proposed inference improvements I believe that additional averaging strategies could be beneficial, but I haven't been able to thoroughly verify this.

# References

[1] Romain Brégier. Roma2024, 2024. Version 1.0, accessed: Feb 8, 2025.

[2] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. *arXiv preprint arXiv:1712.07629*, 2017.

[3] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint detection and description of local features. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8092–8101, 2019.

[4] Johan Edstedt and Michael Felsberg. Dkm: Dense kernelized matching for robust image correspondence. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[5] Asya Olshasky. computervisionproject2025, 2025. Version 1.0, accessed: Feb 8, 2025.

[6] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. *arXiv preprint arXiv:2103.13413*, 2021.

[7] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1623–1637, 2022.

[8] Jerome Revaud, Philippe Weinzaepfel, and Hadrien De Souza, Cordelia Schmid. R2d2: Repeatable and reliable detector and descriptor. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[9] Shunzhou Sun, Shichao Zhao, Zhaopeng Wang, Marc Pollefeys, Andreas Geiger, and Qianqian Dai. Roma: Robust dense feature matching for visual localization. *arXiv preprint arXiv:2210.03034*, 2022.

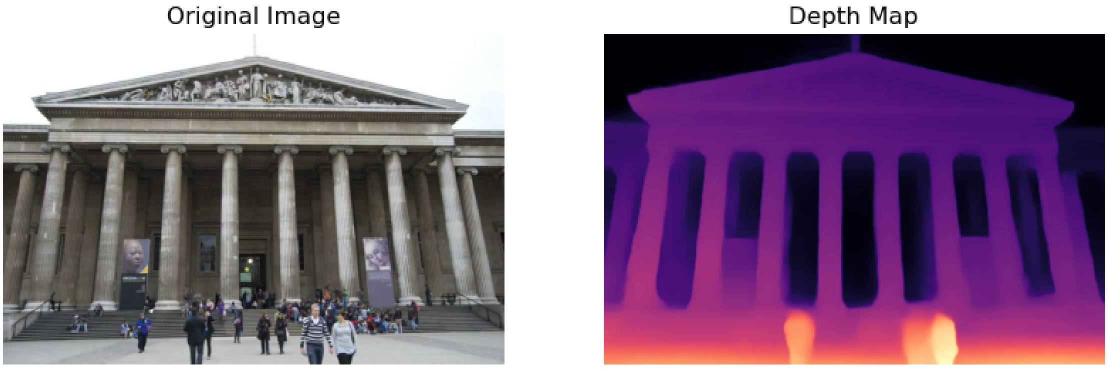

Figure 1: DPT: Depth for the first image

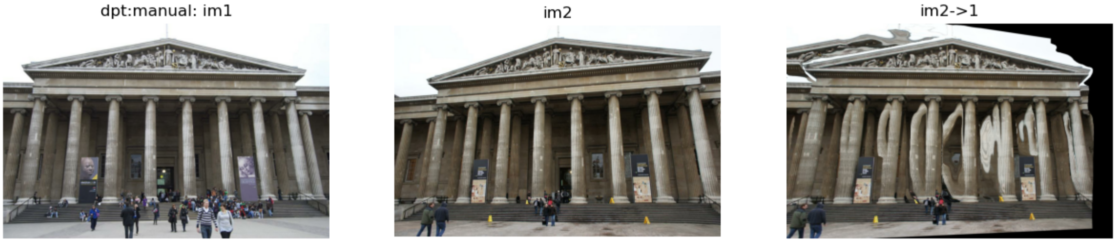

Figure 2: DPT: Second image warped to the first one using depth of the first image and matched points nanually marked

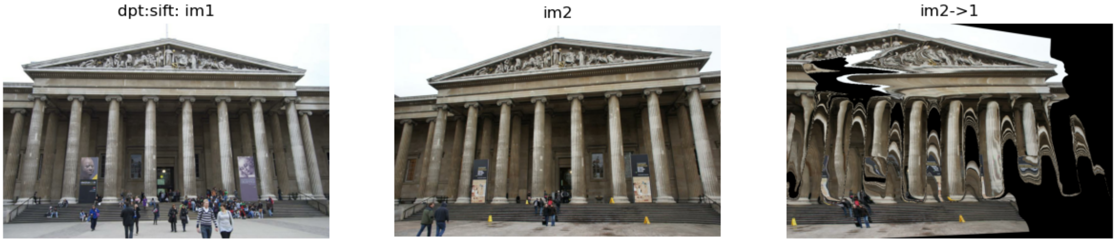

Figure 3: DPT: Second image warped to the first one using depth of the first image and best matched points selected using sift features

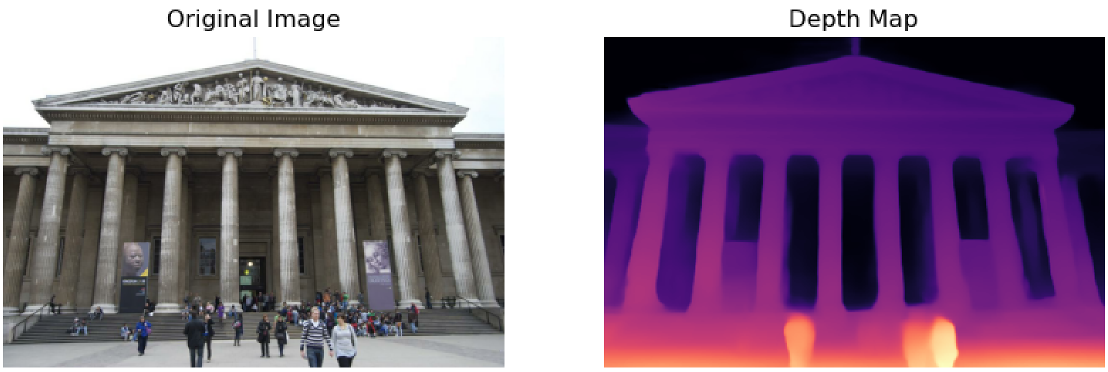

Figure 4: MiDaS: Depth for the first image

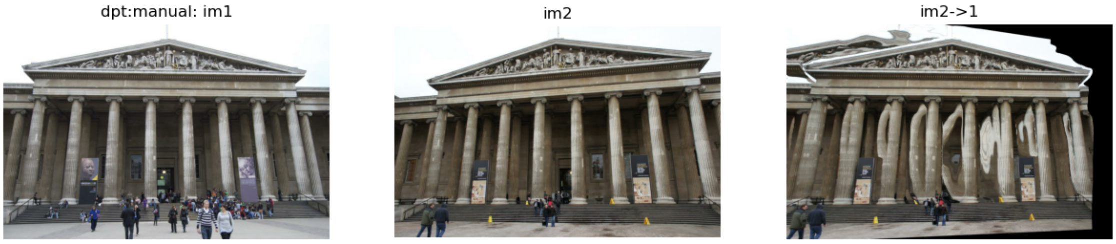

Figure 5: MiDas: Second image warped to the first one using depth of the first image and matched points nanually marked

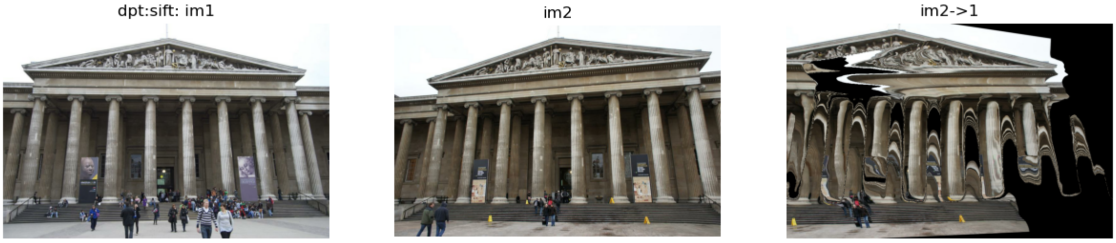

Figure 6: MiDaS: Second image warped to the first one using depth of the first image and best matched points selected using sift features