The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLVIII-1/W4-2025
EuroCOW 2025 – European Workshop on Calibration and Orientation Remote Sensing, 16–18 June 2025, Warsaw, Poland

# Perspective-n-Point in Practice: Performance, Robustness, and Accuracy for Mesh-Based Localisation

F. Vultaggio[1, 2], P. Fanta-Jende[1], M. Gerke[2]

[1] Austrian Institute of Technology - Center for Vision, Automation and Control, Unit Assistive and Autonomous Systems
[2] Technische Universität Braunschweig - Institute of Geodesy and Photogrammetry

Emails: (francesco.vultaggio, phillipp.fanta-jende)@ait.ac.at
(m.gerke)@tu-braunschweig.de

**Keywords:** Perspective-n-Point, PnP, Benchmark, Image Orientation, Mesh, Aerial Image, Smartphone Image, Visual Localisation

**Abstract**

Visual localisation, the task of determining camera poses from images, has matured significantly, offering various solutions for handheld device localisation. This paper investigates the Perspective-n-Point (PnP) problem, a crucial step in visual localisation that is often underexplored in practical applications. We evaluate the performance of state-of-the-art PnP algorithms with real-world data, analysing their impact on localisation accuracy and robustness. Using a dataset comprising a large-scale aerial mesh and smartphone images, we conduct experiments to assess PnP algorithm performance. Specifically, we examine the effects of PnP algorithms in isolation, followed by the incorporation of RANSAC for outlier rejection, and finally, the addition of non linear pose refinement. By maintaining a fixed set of 2D-3D correspondences, this approach allows us to: assess the true outlier rejection capabilities of PnP algorithms, quantify the accuracy improvement achievable with non linear pose refinement, and identify superior PnP algorithms for robust visual localisation.

## 1. Introduction

Visual localisation is a well studied problem in the field of computer vision and photogrammetry. The task of determining the camera pose from images has matured significantly, offering various solutions for handheld device localisation (Miao et al., 2024).The exploration of newer, more robust image matching techniques based on Deep Learning (Xu et al., 2024) has been a primary driver of increased performance. Simultaneously, the literature has explored novel map representations as a means to scale up the localisation process. In this context, mesh-based visual localisation has emerged as a promising approach, leveraging the advantages that 3D meshes offer in terms of reduced storage requirements (Panek et al., 2022), availability (Vultaggio et al., 2024), and flexibility (Shahat et al., 2021).

However, the Perspective-n-Point (PnP) problem (Pan and Wang, 2021), a crucial step in visual localisation, is often underexplored in practical applications. PnP algorithms solve the camera resection problem, estimating the camera pose from a set of 2D-3D correspondences, and their performance can significantly impact the overall localisation accuracy and robustness. In this paper, we investigate the performance of state-of-the-art PnP algorithms in the context of mesh-based visual localisation, focusing on their effectiveness in real-world scenarios.

Mesh-based localisation, albeit a promising avenue for scalable visual localisation, offers a particularly challenging environment for PnP algorithms, as the 3D points often suffer from high levels of noise. This is because the underlying 3D models are derived from a dense reconstruction process, followed by meshing, and texturing steps. At each phase, errors are introduced and compounded, making the final 3D model especially noisy. Additionally, outliers originating from the initial image matching stage can propagate, leading to erroneous 2D-3D inputs for PnP algorithms and further complicating the pose estimation process. Therefore, it is essential to evaluate the performance of PnP algorithms in this context and evaluate the trade-offs between accuracy, robustness, and speed.


Figure 1. Example of the rendering process. The central image shows the aerial view of the map with overimposed dots where the camera poses are, and at its side examples of the rendered and real images side by side.

Our key contributions include: (1) a benchmark for PnP algorithms in mesh-based localisation, (2) quantitative analysis of outlier rejection and pose refinement, and (3) practical recommendations for real-world deployment.

## 2. Related Works

Perspective-n-Point (PnP) algorithms are fundamental components in various computer vision tasks, notably visual localisation (Miao et al., 2024). Despite their importance, the literature currently lacks a systematic performance analysis of state-of-the-art PnP methods and an agreed-upon benchmark for their evaluation. While (Pan and Wang, 2021) provides a comprehensive theoretical review of recent PnP algorithms, their work omits a comparative performance benchmark.

Within the visual localisation field, benchmarks typically focus on the influence of different feature matching strategies on localisation accuracy [1] rather than comparing the performance of the underlying PnP solvers themselves. Consequently, performance analyses of specific PnP algorithms are often confined to the papers introducing them, see Table 2. However, these evaluations frequently rely on synthetic data generated under disparate conditions regarding the quantity and quality of 2D-3D correspondences, making it challenging to generalise findings or compare results across studies reliably.

A valuable contribution towards empirical evaluation was made by (Henry and Christian, 2024), who utilised a real-world Structure-from-Motion (SfM) dataset (Schops et al., 2017) to assess PnP algorithm performance. Our work distinguishes itself by proposing a benchmark built upon a dataset comprising significantly more challenging 2D-3D correspondences. As detailed in Sec. 3.1, the increased difficulty stems from noisier source data (meshes and real imagery versus SfM point clouds) and the presence of substantial outliers introduced during image matching, thus providing a more demanding testbed for evaluating modern PnP algorithms.

## 3. Methodology

This section details the experimental methodology designed to evaluate the performance of various PnP algorithms within a realistic and scalable visual localisation pipeline. The core idea is to utilise real-world imagery, corresponding ground truth (GT) poses, and a 3D mesh model of the scene to generate challenging, yet representative, 2D-3D correspondence sets, which then serve as input for the PnP solvers.

### 3.1 Dataset

The dataset used in this study comprises a large-scale aerial mesh with average ground sampling distance (GSD) of 7.5 cm, which serves as the 3D reference model for the scene, and a collection of query images collected from a smartphone. The mesh is generated using a dense reconstruction pipeline, which processes a set of aerial images captured over the area of interest. For each real query image, a high-accuracy 6 Degree-of-Freedom (DoF) ground truth pose, $T_{\text{gt}}$, is required. This pose defines the transformation from the camera's coordinate system to the world coordinate system shared with the 3D mesh. The GT poses are obtained through a combination of high-accuracy GNSS measurements and inertial data, which are then refined using a Structure from Motion (SfM) pipeline. The real query images' GT poses and 3D mesh are co-referenced using a set of Ground Control Points (GCPs) that are accurately surveyed in the field. The GCPs are used to establish a common coordinate system between the aerial images and the 3D mesh, ensuring that the generated dataset is spatially consistent. Furthermore, the GCPs allow us to evaluate the uncertainty of the GT pose estimate which results in a pose uncertainty of 1.1 cm. More details on the dataset generation process can be found in our previous work (Vultaggio et al., 2024).

[1] https://www.visuallocalisation.net/

### 3.2 Correspondences Generation

Figure 1 shows qualitative examples of the rendering process. As can be seen, certain images appear to be more difficult to match than others, often due to the presence of occlusions in the map caused by vegetation, which is not always modelled accurately in the mesh-generation process.

To simulate a mesh-based localisation scenario where a query image is matched against a model-based representation, we generate 2D-3D correspondences by matching real images to rendered views. This process involves the following steps for each real image $I_{\text{real}}$ and its GT pose $T_{\text{gt}}$:

1. Render a view of the 3D mesh using the GT pose $T_{\text{gt}}$, and the real camera's intrinsic parameters, resulting in a synthetic image pair: the RGB buffer, $I_{\text{syn}}$, and its associated depth buffer, $D_{\text{syn}}$.
2. Extract and match 2D features from $I_{\text{real}}$ and $I_{\text{syn}}$.
3. For each matched feature, compute the corresponding 3D point in the mesh using the depth map associated with the rendered image $D_{\text{syn}}$.
4. Compute the per-correspondence reprojection error as the pixel distance between the keypoints matched across real and synthetic images.

In our tests we match real images to images rendered from their GT position to isolate the error contributions caused by the resectioning pipeline. A realistic mesh-based visual localization pipeline (Vultaggio et al., 2024, Panek et al., 2022) would introduce other potential sources of noise in the pose estimate complicating the pose error analysis. The matching process between real and synthetic images is performed using two different techniques: (1) a traditional feature matching approach based on UprightRootSIFT (Lowe, 2004, Arandjelović and Zisserman, 2012) features, matched using FLANN (Muja and Lowe, 2009) with ratio test set to 0.75, and (2) a deep learning-based method using XFeat (Potje et al., 2024) for feature extraction and LightGlue (Lindenberger et al., 2023) for matching them.

We evaluate two distinct matching techniques to assess their impact on the downstream performance of PnP algorithms. These approaches are characterised by analysing the distribution of their per-correspondence reprojection errors.

Aggregating the reprojection errors from all the matched features across all images for each matching technique reveals that the error distributions exhibit a bi-modal

| | SIFT | XFeat |
|---|---|---|
| Inlier % | 51.0 | 76.4 |
| Mean Inlier error [px] | 44.9 | 16.3 |
| Mean Outlier error [px] | 489.2 | 172.7 |
| Threshold [px] | 119.1 | 51.3 |
| Total Correspondences | 42K | 1.3M |

Table 1. Results of the GMM analysis on matches across real and synthetic views

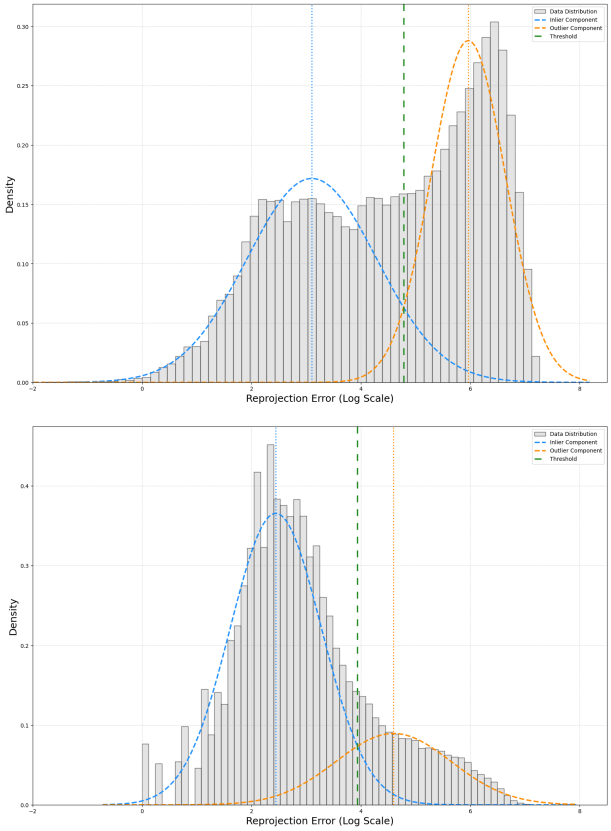


Figure 2. Distribution of the reprojection errors in log space for SIFT, top, and XFeat, bottom.

log-normal distribution, see Fig. 2. The bimodal nature of the data is expected, given the nature of the correspondence generation process. Image matching is a process prone to misassociations: features may be close in descriptor space even if they do not represent the same real-world point, resulting in incorrect image level correspondences. Typically, inliers and outliers are distinguished by applying a threshold to an error metric to ensure the final model estimate derived from these correspondences meets an acceptable tolerance. However, lacking a predefined maximum pose error tolerance our models have to respect, we statistically define the threshold as the value that best separates the modes of the reprojection errors. By performing a Gaussian Mixture Model (GMM) analysis on these distributions, we can estimate the relative proportion of inliers and outliers in both populations, together with their average reprojection error, see Tab 1.

We observe that the learned feature matching approach XFeat (Potje et al., 2024) yields a higher percentage of inliers - and with greater accuracy - compared to the traditional SIFT-based method (Lowe, 2004). In fact, using SIFT in its default configuration yields no matches for the vast majority of images in our dataset. In our experiments, we had to increase the number of octaves and decrease the edge-suppression thresholds, as this was necessary to detect keypoints given the limited resolution of the textures used in the mesh map.

| Method | Reference |
|---|---|
| EPnP | (Lepetit et al., 2009) |
| RPnP | (Li et al., 2012) |
| OPnP | (Zheng et al., 2013) |
| CPnP | (Zeng et al., 2023) |
| oDLT | (Henry and Christian, 2024) |
| REPPnP | (Ferraz et al., 2014) |
| R1PPnP | (Zhou et al., 2019) |

Table 2. PnP algorithms used in this study.

### 3.3 Pose Estimation

Once the set of 2D–3D correspondences has been established (as described in Section 3.2), we proceed to estimate the camera pose using a selection of Perspective-n-Point (PnP) algorithms (see Table 2). The PnP problem is defined as follows: given a set of $n$ 2D image points $\boldsymbol{x}_i \in \mathbb{R}^2$ and their corresponding 3D world points $\boldsymbol{X}_i \in \mathbb{R}^3$, the objective is to determine the camera pose $T = [R \mid t]$. This pose, consisting of a rotation $R \in SO(3)$ and a translation $t \in \mathbb{R}^3$, best aligns the 3D points with their 2D projections according to the camera's projection model.

The PnP algorithms benchmarked in this study represent a range of approaches developed over the years, from efficiency focused solvers to more recent methods focusing on statistical optimality or integrated outlier handling. For clarity in our analysis, we group these methods into two primary families based on their inherent design towards outlier robustness. The first and largest group comprises algorithms that lack a specific mechanism to reject outliers from the input 2D–3D correspondences.

EPnP (Lepetit et al., 2009) was one of the first non iterative solutions to the PnP problem able to run in $O(n)$ time with respect to the input number of points. It still remains the baseline against which other methods measure themselves in terms of both accuracy and speed. EPnP's central idea is to express the 3D world points as a weighted sum of four virtual control points, reducing the problem to efficiently estimating the camera coordinates of these control points.

RPnP (Li et al., 2012) focuses on improving the PnP robustness to degenerate point configurations, which is especially common when the number of input points is low. It selects a primary rotation axis from an edge of the 3D points and forms a cost function from the sum of squares of P3P-derived polynomials for remaining point triplets. The optimum is found by solving the roots of a 7th-order polynomial derived from this cost function.

OPnP (Zheng et al., 2013) is widely regarded as one of the most accurate solutions in the literature. Although it has an analytical complexity of $O(n)$, it exhibits near-constant execution time with respect to the number of

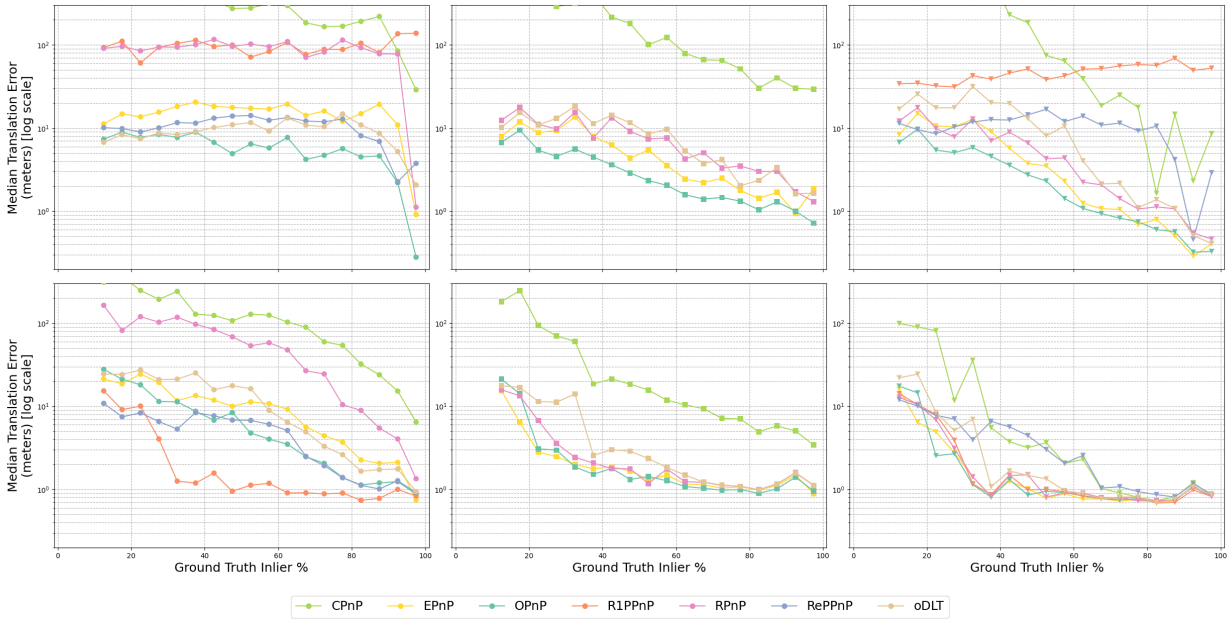


Figure 3. Pose error compared to the inlier percentage in each image. Top row: results with UprightRootSIFT matches. Bottom row: Results with XFeat matches. From left to right: PnP alone, RANSAC + PnP, RANSAC + PnP + Refine

points. However, this constant is relatively high, and the method only outperforms others when processing thousands of points. It formulates the PnP problem as an unconstrained optimisation of an algebraic error using a non-unit quaternion parametrisation for the rotation. It retrieves all stationary points of its cost function via a Gröbner basis technique, aiming for a globally optimal solution with respect to this algebraic cost function.

CPnP (Zeng et al., 2023) is a modern PnP solver whose focus is on tackling the large number of correspondences that modern image matching techniques often produce. It achieves this by constructing linear equations from the projection model, performing variable elimination, and then explicitly estimating and subtracting the asymptotic bias from the least-squares solution before refinement with Gauss-Newton iterations.

oDLT (Henry and Christian, 2024) significantly refines the classic Direct Linear Transform (DLT) (Hartley and Zisserman, 2003) for calibrated PnP. It achieves near-optimal accuracy by a "two-shot" process: an initial DLT solution on a small, random subset of points is used to compute fixed weights for all correspondences. These weights are then applied to a single solve of the full DLT system. The resulting projection matrix is then decomposed into an optimal rotation (via weighted Procrustes) and translation, with an optional non-iterative refinement for translation. oDLT aims for the accuracy of iterative methods at the speed of DLT.

On the other hand, several methods have been proposed to address the problem of outliers in the set of input correspondences. This is an inevitable aspect of real data and is often tackled with a Random Sample Consensus (Bolles and Fischler, 1981) (RANSAC) based prefiltering step, which can at times be slow and sub-optimal (Jin et al., 2021).

REPPnP (Ferraz et al., 2014) builds on EPnP by maintaining the formulation of the PnP problem as a sum of four virtual control points but introduces an algebraic outlier rejection scheme. It robustly estimates the 1D null space of the PnP linear system by iteratively reweighting correspondences based on their algebraic error. This allows REPPnP to quickly converge while maintaining tolerance to outliers in the input data.

R1PPnP (Zhou et al., 2019) combines an iterative core PnP algorithm (which uses a single 3D-2D correspondence as a "control point" and alternates between estimating relative depths and camera pose) with two outlier handling strategies: a soft re-weighting mechanism based on reprojection errors integrated into the core algorithm, and a 1-Point RANSAC scheme to try different correspondences as the control point. This design aims for robustness to very high outlier percentages, potentially exceeding the limits of algebraic rejection methods.

## 4. Experiments

### 4.1 Experimental configuration

To comprehensively benchmark their performance and robustness, each PnP algorithm (see Table 2) will be evaluated under three distinct configurations:

1. **Standalone Execution:** The PnP algorithm is run directly on the *entire set* of 2D–3D correspondences, including any potential outliers. This configuration assesses the inherent accuracy and outlier handling capabilities of the PnP algorithm itself.
2. **RANSAC Pre-filtering:** A RANSAC loop, employing a minimal P3P solver (Ding et al., 2023) as its model generator, is first used to identify a consensus set of inlier correspondences. The PnP algorithm then estimates the camera pose using *only these identified inliers*. For this RANSAC step, we will use as the inlier reprojection thresholds those

identified in subsection 3.2 and a maximum of 2000 iterations. The methods which incorporate already an outlier rejection mechanism, i.e. the works of (Ferraz et al., 2014, Zhou et al., 2019), are not tested in this configuration.

3. **RANSAC Pre-filtering and Non-linear Refinement:** Following the RANSAC pre-filtering and initial pose estimation by the PnP algorithm on the inliers (as in configuration 2), the resulting pose $[R_{\text{pnp}} \mid t_{\text{pnp}}]$ is further optimised. This non-linear refinement step minimises the sum of squared reprojection errors for the *inlier correspondences* identified by RANSAC (Schönberger and Frahm, 2016).

We will present the accuracy metrics of each of the algorithms in each of the configurations detailed above in terms of their pose error compared to both inlier percentage and execution time.

### 4.2 Experimental results

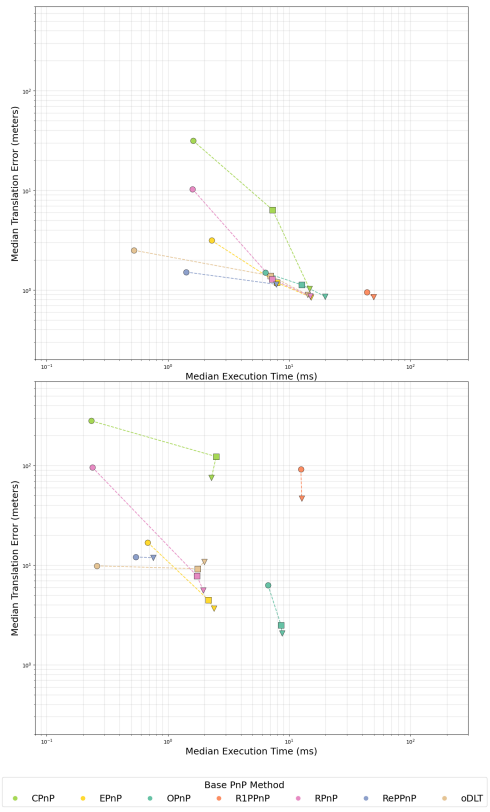


Figure 4. Median pose error [m] compared to median execution time [ms]. Top: results with UprightRootSIFT matches. Bottom: results with XFeat matches. Circle denotes PnP alone, square RANSAC + PnP, triangle RANSAC + PnP + Refine

Our experimental findings are presented in Figure 3. This figure displays the median pose error (y-axis) as a function of the inlier percentage (x-axis). To generate these data points, images were first grouped into 20 bins based on their inlier percentage, spanning from 10% to 100%. Then, for each bin, the median pose error was calculated using all images falling within that specific inlier percentage range. This allows us to evaluate the performance of the different PnP algorithms we analysed (see Table 2) across the input 2D-3D correspondences derived from SIFT and XFeat matches. Regarding rotational accuracy, the observed trends largely mirror those for translation errors across the different PnP setups and correspondence types. For conciseness, the results for rotation errors are provided in Figure 5 in the Appendix.

Comparing the accuracy across all PnP setups between the SIFT and XFeat sets of input 2D-3D correspondences, it is evident that the XFeat correspondences result in a positional error approximately an order of magnitude lower for most of the inlier level thresholds. This can be explained by the higher mean reprojection error of the SIFT inliers, see Table 1, especially in the PnP alone setup. However, it is evident that the pose errors for the XFeat based matches do not appear to decrease further than 0.9m, regardless of inlier threshold or PnP setup. In contrast with SIFT-based correspondences, the best PnP methods can achieve sub-meter median positional error, 0.1m, for very high inlier thresholds. This result can be explained by the fact that most deep learning based feature extractors predict keypoints by learning how to score every image pixel for its potential as a keypoint, meaning that these methods cannot operate at the subpixel level. The lack of subpixel precision can also be observed when looking at the sparse left tail of XFeat reprojection error distribution, see Figure 2.

When analysing the accuracy of the different PnP setups, we find that the relative performance among the algorithms generally remains consistent across both SIFT and XFeat correspondences. R1PPnP (Zhou et al., 2019) deviates from this pattern. It achieves accurate results with XFeat's 2D-3D correspondences, yet its performance is considerably poorer with SIFT-derived ones, suggesting its outlier rejection may depend heavily on low-noise inliers. In contrast, the other outlier-resistant method, REPPnP (Ferraz et al., 2014), maintains similar relative accuracy when compared to the other methods, in practice being one of the most accurate algorithms, when tested alone, but lags behind the others once the pose refinement step is included. This is likely due to the fact that, although the method is resistant to outliers, it does not explicitly provide a per-correspondence inlier classification. As a result, the pose refinement step must operate on all correspondences, which greatly reduces its effectiveness.

Turning to the other non-outlier-resistant methods, we observe that they generally perform similarly to one another, with the exception of CPnP (Zeng et al., 2023), which performs noticeably worse than the others, even when RANSAC pre-filtering or pose refinement is applied. The authors claim this method is specifically designed to work on large correspondence sets, possibly indicating it is better tailored to process correspondences generated by dense approaches, such as RoMa (Edstedt et al., 2024). Amongst the others, OPnP (Zheng et al., 2013) remains the most accurate method, as other works have found (Pan and Wang, 2021), followed by EPnP (Lepetit et al., 2009), then RPnP (Li et al., 2012), and finally oDLT (Henry and Christian, 2024). However, especially when looking at the results coming from the XFeat correspondences pre-filtered using RANSAC and

whose pose has been refined, the final pose errors are so close as to be practically equivalent for most applications.

When examining the median pose error versus the median execution time across all images in our dataset, we observe that although some algorithms achieve similar accuracy, their execution times differ significantly (see Figure 4). In fact, a clear Pareto frontier can be observed, where the two setups able to offer a better accuracy to execution-time trade-offs are oDLT (Henry and Christian, 2024) and REPPnP (Ferraz et al., 2014), both running alone. However, if we consider only the methods that achieve sub-meter median positional error, we observe that all of them require both pre- and post-processing - except for R1PPnP (Zhou et al., 2019), which is the only method to reach this level of accuracy without RANSAC pre-filtering, at the cost of a greater execution time than the methods which employ a standard RANSAC pre-filtering.

Synthesising these findings, we can offer practical recommendations for selecting a PnP strategy based on specific application needs. If the primary requirement is a strong balance between *execution speed and positional accuracy*, our experiments indicate that REPPnP or oDLT, when used alone, present the most compelling options by pushing beyond the Pareto frontier of the other setups.

However, in applications where *accurate* camera pose estimation is critical, we find that the combination of RANSAC pre-filtering followed by a robust classical solver (EPnP, RPnP, OPnP, or oDLT) and concluded with non-linear pose refinement, yields the lowest median errors.

In contrast, while R1PPnP demonstrates commendable accuracy without external RANSAC, its significantly higher computational cost makes it less suitable for time-sensitive applications compared to the RANSAC-based pipelines achieving similar or better accuracy. CPnP, within our experimental setup, did not reach the accuracy levels of the other leading methods, even with the aid of pre- and post-processing steps.

A critical factor limiting further accuracy improvements, especially with modern deep-learning-based features like XFeat, appears to be their inherent lack of subpixel precision. As observed in Figure 3 and Figure 2, this can introduce a performance plateau (around 0.9 m in our XFeat tests) that even the best PnP algorithms struggle to break through. This suggests that the quantisation of keypoint coordinates from such feature extractors currently forms a practical bottleneck for achieving finer pose estimates.

## 5. Conclusions

This paper presented a performance analysis of state-of-the-art Perspective-n-Point (PnP) algorithms based on real-world data. Our findings highlight the impact of input correspondence quality, robust outlier filtering, and pose refinement on final accuracy, leading to clear recommendations for practitioners. Once again, we have identified the importance of good input correspondences, as no technique, or combination thereof, can yield accurate pose estimates from high-noise, low-inlier-percentage 2D-3D correspondences. For applications prioritising fast execution, standalone REPPnP or oDLT are advised. When maximum accuracy is paramount, established solvers such as EPnP, OPnP, RPnP, or oDLT, augmented with RANSAC pre-filtering and non-linear refinement, consistently yield the best results. Crucially, our analysis identified a performance plateau when using modern learned features, attributable to their lack of subpixel keypoint precision, which currently forms a significant bottleneck. Addressing this challenge by either developing subpixel-aware feature extractors or quantisation-aware PnP methods is critical to advancing the achievable precision of visual localisation systems.

## Acknowledgements


Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the EUSPA. Neither the European Union nor the EUSPA can be held responsible for them.

We thank Alexander Kern (TU Braunschweig, Institute of Flight Guidance) for the data he contributed to this work.

## References

Arandjelović, R., Zisserman, A., 2012. Three things everyone should know to improve object retrieval. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 16–21, doi: 10.1109/CVPR.2012.6248018.

Bolles, R. C., Fischler, M. A., 1981. A ransac-based approach to model fitting and its application to finding cylinders in range data. *IJCAI*, 1981, 637–643.

Ding, Y., Yang, J., Larsson, V., Olsson, C., Åström, K., 2023. Revisiting the p3p problem. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4872–4880.

Edstedt, J., Sun, Q., Bökman, G., Wadenbäck, M., Felsberg, M., 2024. RoMa: Robust Dense Feature Matching. *IEEE Conference on Computer Vision and Pattern Recognition*.

Ferraz, L., Binefa, X., Moreno-Noguer, F., 2014. Very fast solution to the pnp problem with algebraic outlier rejection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 501–508.

Hartley, R., Zisserman, A., 2003. *Multiple view geometry in computer vision*. Cambridge university press.

Henry, S., Christian, J. A., 2024. Optimal DLT-based Solutions for the Perspective-n-Point. *arXiv preprint arXiv:2410.14164*.

Jin, Y., Mishkin, D., Mishchuk, A., Matas, J., Fua, P., Yi, K. M., Trulls, E., 2021. Image Matching Across Wide Baselines: From Paper to Practice. *Int. J. Comput. Vision*, 129(2), 517–547. doi: 10.1007/s11263-020-01385-0.

Lepetit, V., Moreno-Noguer, F., Fua, P., 2009. EPnP: An Accurate O(n) Solution to the PnP Problem. *Springer Verlag*, 81, 155–166. doi: 10.1007/s11263-008-0152-6.

Li, S., Xu, C., Xie, M., 2012. A robust O (n) solution to the perspective-n-point problem. *IEEE transactions on pattern analysis and machine intelligence*, 34(7), 1444–1450.

Lindenberger, P., Sarlin, P.-E., Pollefeys, M., 2023. Lightglue: Local feature matching at light speed. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 17627–17638.

Lowe, D. G., 2004. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision*, 60(2), 91–110. doi: 10.1023/B:VISI.0000029664.99615.94.

Miao, J., Jiang, K., Wen, T., Wang, Y., Jia, P., Wijaya, B., 2024. A Survey on Monocular Re-Localization: From the Perspective of Scene Map Representation. *IEEE Trans. Intell. Veh.*, 1–33. doi: 10.1109/TIV.2024.3378716.

Muja, M., Lowe, D. G., 2009. Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*, 2(331-340), 2.

Pan, S., Wang, X., 2021. A Survey on Perspective-n-Point Problem. *2021 40th Chinese Control Conference (CCC)*, IEEE, 26–28, doi: 10.23919/CCC52363.2021.9549863.

Panek, V., Kukelova, Z., Sattler, T., 2022. MeshLoc: Mesh-Based Visual Localization. *Computer Vision – ECCV 2022*, Springer, Cham, Switzerland, 589–609, doi: 10.1007/978-3-031-20047-2_34.

Potje, G., Cadar, F., Araujo, A., Martins, R., Nascimento, E. R., 2024. Xfeat: Accelerated features for lightweight image matching. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2682–2691.

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

Schops, T., Schonberger, J. L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A., 2017. A multi-view stereo benchmark with high-resolution images and multi-camera videos. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3260–3269.

Shahat, E., Hyun, C. T., Yeom, C., 2021. City Digital Twin Potentials: A Review and Research Agenda. *Sustainability*, 13(6), 3386. doi: 10.3390/su13063386.

Vultaggio, F., Fanta-Jende, P., Schörghuber, M., Kern, A., Gerke, M., 2024. Investigating Visual Localization Using Geospatial Meshes. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48, 447–454.

Xu, S., Chen, S., Xu, R., Wang, C., Lu, P., Guo, L., 2024. Local feature matching using deep learning: A survey. *Information Fusion*, 107, 102344. doi: 10.1016/j.inffus.2024.102344.

Zeng, G., Chen, S., Mu, B., Shi, G., Wu, J., 2023. Cpnp: Consistent pose estimator for perspective-n-point problem with bias elimination. *2023 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 1940–1946.

Zheng, Y., Kuang, Y., Sugimoto, S., Åström, K., Okutomi, M., 2013. Revisiting the pnp problem: A fast, general and optimal solution. *2013 IEEE International Conference on Computer Vision*, 2344–2351. doi: 10.1109/ICCV.2013.291.

Zhou, H., Zhang, T., Jagadeesan, J., 2019. Re-weighting and 1-Point RANSAC-Based PnP Solution to Handle Outliers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(12), 3022-3033. doi: 10.1109/TPAMI.2018.2871832.

## Appendix

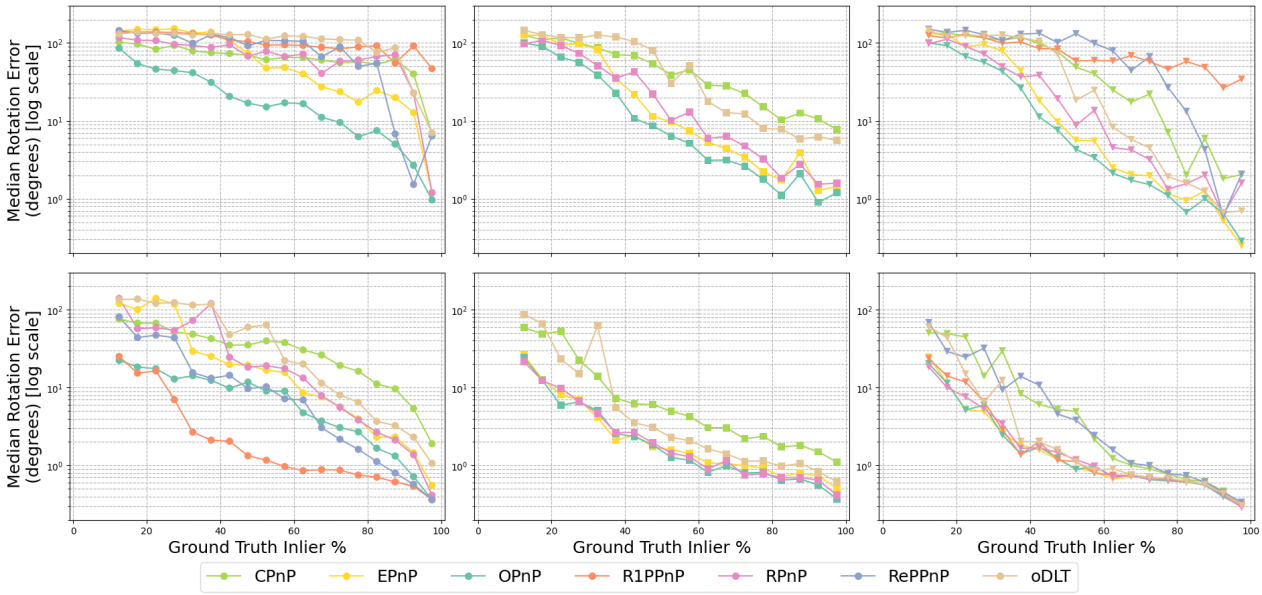


Figure 5. Rotation error compared to the inlier percentage in each image. Top row: results with UprightRootSIFT matches. Bottom row: Results with XFeat matches. From left to right: PnP alone, RANSAC + PnP, RANSAC + PnP + Refine