U–ARE–ME: Uncertainty-Aware Rotation Estimation in Manhattan Environments

Aalok Patwardhan*, Callum Rhodes*, Gwangbin Bae, and Andrew J. Davison
Dyson Robotics Lab, Imperial College London
{a.patwardhan21, c.rhodes, g.bae, a.davison}@imperial.ac.uk
* denotes equal contribution.

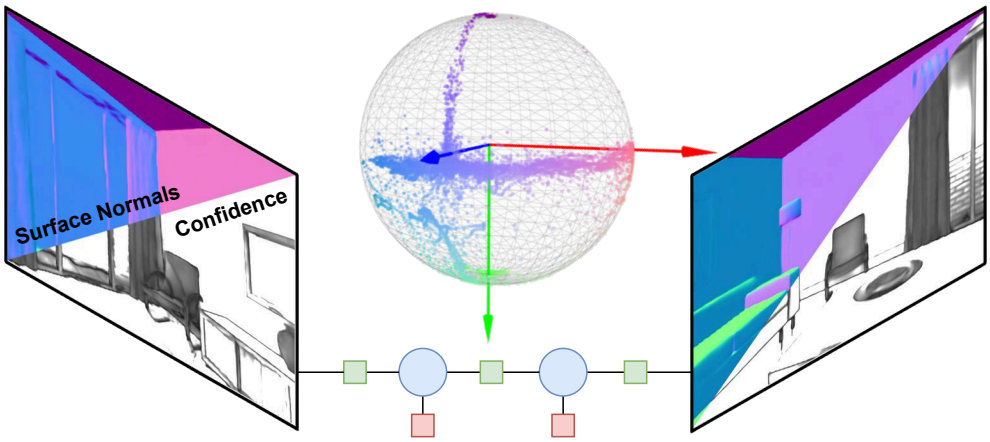

Figure 1. U–ARE–ME provides globally consistent rotation estimates in Manhattan environments across sequences of uncalibrated RGB images – no camera intrinsics needed. Rotations are estimated using per-pixel predicted surface normals and confidence.

Abstract

Camera rotation estimation from a single image is a challenging task, often requiring depth data and/or camera intrinsics, which are generally not available for in-the-wild videos. Although external sensors such as inertial measurement units (IMUs) can help, they often suffer from drift and are not applicable in non-inertial reference frames. We present U-ARE-ME, an algorithm that estimates camera rotation along with uncertainty from uncalibrated RGB images. Using a Manhattan World assumption, our method leverages the per-pixel geometric priors encoded in single-image surface normal predictions and performs optimisation over the SO(3) manifold. Given a sequence of images, we can use the per-frame rotation estimates and their uncertainty to perform multi-frame optimisation, achieving robustness and temporal consistency. Our experiments demonstrate that U-ARE-ME performs comparably to RGB-D methods and is more robust than feature-based vanishing point and SLAM methods.

1. Introduction

Accurate estimation of camera rotation from a sequence of monocular images is crucial for many computer vision applications, including visual odometry [20], image stabilisation [37], and augmented reality [46]. Many solutions have been proposed for a variety of sensor setups. For instance, the recently released Apple Vision Pro operates using visual-inertial odometry, relying on both the cameras and the inertial measurement units (IMUs). However, IMUs are prone to drift, are by design not suitable for non-inertial frames of reference and simply may not be available alongside images.

If depth measurements (paired with the input images) are available, the RGB-D frames can be aligned — based on photometric and geometric consistency — to recover their relative camera poses [9]. If the surface normal vectors in the scene are aligned with a set of *principal directions*, the camera rotation can be found by aligning the input normals (extracted from the depth maps) to those directions [16, 42, 44]. While such approaches provide drift-free rotation estimates with high accuracy, they cannot be applied to

in-the-wild videos or devices without a depth sensor.

This paper focuses on the most challenging setup in which only RGB input is available. Previous attempts have focused on detecting and matching *2D image features*. For instance, ORB-SLAM [38] tracks sparse ORB features, while methods like [4, 31, 40] group line segments to identify the vanishing points (VPs) and hence the camera rotation with respect to the principal directions. However, such methods are sensitive to image degradation (e.g. noise and motion blur) and perform poorly in textureless environments. More importantly, many of these methods assume known camera intrinsics — which are often not available for in-the-wild videos. While a neural network can be trained to regress the rotation between consecutive frames [6], such an approach is prone to overfitting and drift. It is also computationally costly to train such a specialised model.

In this work, we propose to make use of the dense pixel-wise geometric priors learned by *single-image surface normal estimation models*. Surface normal estimation models are efficient (e.g. [3] runs at ~70+ fps on an NVIDIA 4090 GPU) and have strong generalisation ability [2, 3]. In recent years, their usefulness has been demonstrated for various computer vision tasks, including object grasping [51], vision-language reasoning [35], simultaneous localisation and mapping [34], and CAD model alignment [28]. We explore whether such powerful front-end perception can also be used for rotation estimation.

Similar to previous optimisation-based approaches [16, 42, 44], we assume a certain distribution of surface normal vectors in world coordinates and optimise for the camera rotation that would align the predicted normals to the principal directions of the scene. While previous methods (1) used depth sensors to extract the normal vectors and (2) were only applicable to a single image, we attempt to remove both constraints.

Two types of uncertainty arise in the process of removing these two commonly adopted constraints. First is the heteroscedastic aleatoric uncertainty [25] in surface normal predictions. As shown in [3], surface normals predicted by a neural network — unlike those extracted from a depth map — are unreliable, especially for the pixels near object boundaries and on small objects. As these pixels should be down-weighted in the optimisation objective, we introduce a new uncertainty-weighted cost function and show how the uncertainty can be learned in a data-driven manner.

The second type of uncertainty arises when the image contains a limited number of principal directions. For instance, when a Manhattan World (MW) [7] is assumed, two (or more) of the six directions ($\pm X, \pm Y, \pm Z$) should be observed to determine the camera rotation. If only one axis is visible, any rotation around that axis would result in an equally valid prediction. To this end, we quantify the uncertainty around each principal axis and use it to enhance the temporal consistency in the predictions.

To summarise, our framework alternates between two optimisation steps:

- **Single-frame optimisation:** We optimise the world-to-camera rotation matrix such that the rotated principal directions are best aligned with the predicted surface normals. We improve the accuracy and robustness by introducing an uncertainty-weighted cost function.
- **Multi-frame optimisation:** We take the covariance matrix of rotation around each axis — which is readily available from the Hessian approximation in the second-order optimisation of the first step — and use it to jointly optimise a sliding window of previous frame rotations. We improve the global consistency of our solution, reject outlier rotations and intuitively handle frames that may contain limited information on certain principal axes.

The proposed method runs at ~60+ fps on an NVIDIA 4090 GPU. Note that, unlike the learning-based models that can only be used for rotation estimation, the surface normal predictions — from which we infer the rotation — can be used for other tasks, reducing the overall computational overhead.

The main strength of our approach lies in its *robustness*. Compared to the methods that rely on sparse feature tracking or line segment detection, our approach is more robust to the presence of image degradation. Unlike SLAM-based methods, our approach does not require camera intrinsics and can be applied to a single image or in-the-wild videos, making it useful in a wider range of scenarios[1].

2. Related work

Rotation estimation of a camera from single images has been extensively studied and is generally based upon the assumption that indoor scenes exhibit inherent structure, conforming to the MW assumption. These approaches for Manhattan Frame (MF) estimation broadly fall into two domains; using RGB images to extract perspective cues, and using 3D information such as surface normals from RGB-D images.

Earlier work estimated the MF by considering vanishing points and lines in an image as perspective cues [27]. The work in [29] generated several MF hypotheses from an image, using line segments to find the best fitting model. In [14], line segments were extracted onto a hemisphere, and clustered to identify three orthogonal directions although this method was sensitive to the chosen resolution of the hemisphere discretisation. The algorithm proposed by [13] used line clustering to find three vanishing points, and achieved real-time camera rotation estimation over a sequence of images in a video. The work in [31] does not

[1] We encourage the reader to view the supplementary material for further details and experiments on the robustness of our approach.

rely on the MW assumption but uses sequential Bayesian filtering to jointly estimate rotation and vanishing points. Recently a Hybrid Vanishing Point algorithm (H-VP) [40] was presented for uncalibrated images, which can extract the MF by making use of a gravity-direction prior to robustly extract vanishing points. These RGB methods rely on the existence of multiple parallel lines and vanishing points, and are not robust in the presence of noise and outliers, or in texture-less scenes.

Rotation estimation methods that use RGB-D images are more accurate and stable as they utilise 3D information in the scene, whether this is directly through depth camera data, or by using this data to computing the surface normals in the scene. The approach in [42] uses point normals and perspective cues to perform an Exhaustive Search (ES) over a set of candidate directions and using a scoring heuristic to estimate the MF, although this incurs a high computational cost. Depth data from a Kinect camera was used in [47] to determine the MF of a scene by identifying the ground, and selecting a perpendicular direction from one of the walls. This method relies on the presence of a visible floor and multiple walls in the image, and so is not generalisable. In [16] the MF is estimated through non-convex optimisation by considering the sparsity constraints of MF-aligned surface normals.

In [43], the authors argue that real-world scenes contain a Mixture of Manhattan Frames (MMF), which they simultaneously estimate from surface normals calculated from depth data.

These methods are often sensitive to initial conditions and cannot guarantee global optimality, unlike the family of Branch and Bound (BnB) methods which operate in the rotation search space [5, 19, 39]. These methods guarantee global optimality, but cannot be considered real-time algorithms. Real-time rotation estimation is enabled in [24] using the BnB method by maximising the consensus set of inliers over the search space of rotation. Surface normals are discretised on an equi-rectangular plane to generate the Extended Gaussian Image (EGI) [21], from which the BnB approach is used to estimate the MF.

Recently [52] proposed a novel and efficient cost function of Multiple Normal vectors and Multiple MF Axes (MNMA) for MF estimation. The cost function makes use of the vector dot and cross products between the scene surface normals and the axes of the MF leading to an efficient, accurate and real-time algorithm.

All of the methods considered here have used surface normals calculated from depth data from RGB-D images, rather than directly estimating them from an RGB input. Estimating surface normals has traditionally been computationally intensive, producing unreliable results.

3. Method

Given a monocular video, our goal is to estimate the per-frame camera rotation relative to the world coordinates. We begin by assuming that the scene satisfies the MW assumption [7], which is valid for a wide range of indoor/outdoor scenes. Note that it is straightforward to extend our approach to other world assumptions (e.g. Mixture of Manhattan Worlds [43], Atlanta World [41], and Hong Kong World [32]), given that the principal directions in the world coordinates are known *a priori*.

Our method is named **U-ARE-ME** (Uncertainty-Aware Rotation Estimation in Manhattan Environments), as it can be used to complement or replace I-M-U sensors. We leverage recent advances in single-image surface normal estimation and propose to infer the camera rotation by aligning the predicted normals to the world assumption. In Sec. 3.1 we introduce a new uncertainty-aware optimisation objective and show how the uncertainty can be learned from data. For real-time video applications, it is important to ensure temporal consistency in the predictions. We explain in Sec. 3.2 the factor graph formulation required for this.

3.1. Uncertainty-aware rotation estimation from a single image

Suppose that a surface normal vector $\mathbf{n}_i \in S^2$ corresponding to the i-th pixel is aligned with one of the principal directions. Then, for any Manhattan axis $\mathbf{r} \in \{\pm X, \pm Y, \pm Z\}$, the angle $\theta = \cos^{-1}(\mathbf{n}_i \cdot \mathbf{r})$ should be $\{0°, 90°, 180°\}$. To this end, Zhang et al. [52] introduced a cost function $E(\mathbf{r}|\mathbf{n}_i) = \sin^2\theta\cos^2\theta$, which is visualised in Fig. 2. We modify this cost by multiplying it by some confidence measure κ.

$$E(\mathbf{r}|\mathbf{n}_i, \kappa_i) = \kappa_i \sin^2\theta\cos^2\theta \tag{1}$$

To learn κ in a data-driven manner, we pre-train a neural network using the following training loss:

$$\mathcal{L}(\mathbf{n}_i^{gt}|\mathbf{n}_i, \kappa_i) = C(\kappa_i) + \kappa_i \sin^2\theta\cos^2\theta \tag{2}$$

where $\mathbf{n}_i^{gt}$ is the ground truth and θ is the angular error of the predicted normal $\mathbf{n}_i$. $C(\kappa)$ should be a monotonically decreasing function of κ to prevent the model from estimating $\kappa_i = 0$ for every pixel. Another thing to note is that the second term should be defined only for $0° \leq \theta < 45°$. Otherwise, the loss could be minimised by *increasing the error*. To satisfy such constraints, we assume that the surface normal probability distribution can be parameterised as follows:

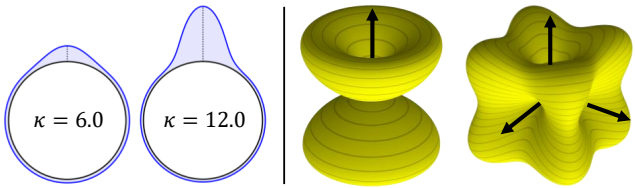

Figure 2. **(left)** Visualisation of the proposed surface normal probability distribution for different values of κ. As κ increases, the distribution becomes more concentrated towards the mean direction. **(right)** Visualisation of the cost function defined by a single axis and another one defined by three mutually orthogonal Manhattan axes. The optimisation cost is minimised when the predicted normals are parallel or vertical to the axes.

$$p(\mathbf{n}_i^{gt}|\mathbf{n}_i,\kappa_i) = \begin{cases} \theta < \frac{\pi}{4} \Rightarrow D(\kappa_i)\exp(-\kappa_i \sin^2\theta\cos^2\theta) \\ \theta \geq \frac{\pi}{4} \Rightarrow D(\kappa_i)\exp(-\frac{\kappa_i}{4}) \end{cases}$$
$$\text{where } C(\kappa_i) = -\log D(\kappa_i). \tag{3}$$

Then, $\mathcal{L}(\mathbf{n}_i^{gt}|\mathbf{n}_i,\kappa_i)$ can be interpreted as the negative log-likelihood of the above distribution. $D(\kappa)$ is a monotonically increasing function of κ as the distribution should be normalised. During training, the network is encouraged to increase the value of κ for the pixels with lower error, thereby encoding the confidence in the prediction. In Fig. 2, we visualise the proposed distribution for different values of κ. As the analytic form for $D(\kappa)$ could not be found, we obtain the values numerically for $\kappa \in [0, 10^5]$ and fit them using natural cubic splines. We use a lightweight convolutional encoder-decoder architecture [1] and use the training data of [2]. See the supplementary material for additional details regarding network training.

Given an image with a set of estimated normals and confidence $\mathcal{I} = \{\mathbf{N} = \{\mathbf{n}\}_i, \mathbf{K} = \{\kappa\}_i\}$, the next task is to determine the optimal rotation $\mathbf{R} \in SO(3)$ that gives the relative rotation of the camera to the MW frame. Since normals are predicted densely, evaluating the cost of a rotation against all normals in the image is prohibitively expensive. To avoid this, a single weighted average cost function can be cheaply calculated for the set of all normals and stays fixed throughout optimisation.

Therefore, using Eq. (1), the modified cost function for the optimisation becomes:

$$E(\mathbf{r}|\mathcal{I}) = \frac{1}{|\mathbf{N}| \times \sum \mathbf{K}} \sum_i^{|\mathbf{N}|} E(\mathbf{r}). \tag{4}$$

At each stage of optimisation, the cost per X, Y, Z axes of the rotation matrix $\mathbf{R} = [\mathbf{r}_x, \mathbf{r}_y, \mathbf{r}_z]$ is evaluated to give the total cost:

$$E(\mathbf{R}|\mathcal{I}) = E(\mathbf{r}_\text{x}|\mathcal{I}) + E(\mathbf{r}_\text{y}|\mathcal{I}) + E(\mathbf{r}_\text{z}|\mathcal{I}). \tag{5}$$

We use Levenberg-Marquardt (LM) optimisation to minimise Eq. (5) (initialised with the identity matrix $\mathbf{I}_3$), rewriting it in terms of the corresponding residual function $f(\mathbf{R})$ (using the parameterisation in [52]) to obtain the optimal rotation $\mathbf{R}^*$.

$$\mathbf{R}^* = \arg\min_{\mathbf{R}} E(\mathbf{R}|\mathcal{I}) \tag{6}$$
$$= \arg\min_{\mathbf{R}} f(\mathbf{R})^\intercal f(\mathbf{R}). \tag{7}$$

During optimisation, the analytical Jacobian of the residual function, $J_f(\mathbf{R})$, is calculated with respect to a perturbation $\Delta\phi$ on the tangent plane (Lie algebra) at the rotation matrix $\mathbf{R}$ i.e. $\mathbf{R} \circ \text{Exp}(\Delta\phi)$, where $\text{Exp}(\cdot)$ denotes the exponential map of the SO(3) group. Using the Hessian approximation from the Jacobian in LM optimisation, we can acquire an approximate measure of the covariance of the converged rotation matrix $\Sigma^{\mathbf{R}^*}$ as:

$$J_f(\mathbf{R}^*) = \frac{\partial f(\mathbf{R}^*)}{\partial \Delta\phi} \tag{8}$$
$$\Sigma^{\mathbf{R}^*} = H^{-1} \approx (J_f(\mathbf{R}^*)^\intercal J_f(\mathbf{R}^*))^{-1} \tag{9}$$

where H is the Hessian matrix approximation.

Having obtained the optimal MW rotation $\mathbf{R}_\text{mw} = \mathbf{R}^{*\intercal}$ and the uncertainty about this frame, the estimate can then be used for further downstream tasks. This could be for bootstrapping visual odometry systems, correcting drift in inertial pipelines, rectifying images for CNNs, which are sensitive to image rotation, as well as camera calibration to name a few. In the following section we address one such extension, that being to estimate camera rotation across a sequence of images to achieve a coherent trajectory.

3.2. Multi-frame rotation estimation for temporal consistency

When estimating single-frame rotation, our method finds the optimal rotation relative to Identity. The problem with applying this method consecutively to a sequence of images is that there will be no temporal consistency between rotations, and therefore when rotating around any single axis, several MW configurations could satisfy the normal distribution. It is simple to naively initialise the current rotation $\mathbf{R}_t$, with the optimised rotation from the previous frame $\mathbf{R}_{t-1}$ which removes this ambiguity and speeds up convergence of the optimisation. However, two main problems arise from this initialisation. Firstly, for frames that are less Manhattan, we have no way of loosening the MW assumption and therefore a single frame's rotation is only dependent on its own (potentially poorly defined) cost function.

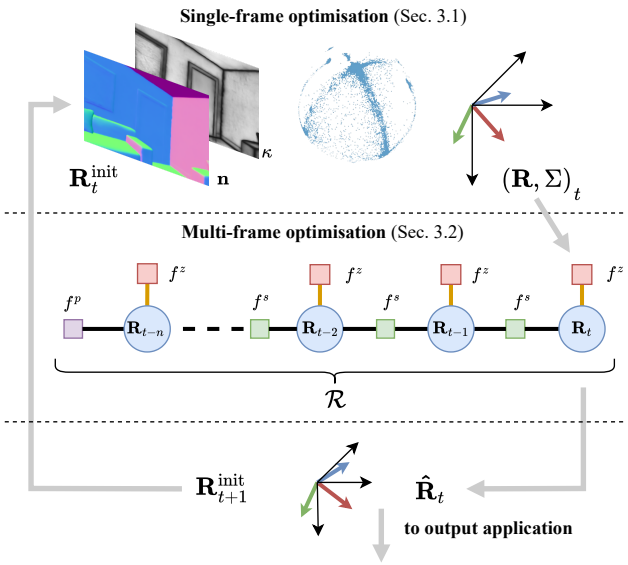

Figure 3. The multi-frame optimisation process. Single-frame rotation and covariance estimates are used to initialise a sliding window factor graph in order to provide temporal consistency between frames and reject outlier measurements. Robust factors are shown along orange edges on measurements. The latest frame is then used to initialise the rotation estimate for the next frame.

Secondly, not all frames in a sequence may optimise to a consistent minimum and may give erroneous rotations that initialise subsequent frames with a bad orientation, potentially poisoning the rest of the sequence. In the following section, we address these issues in order to extend rotation estimation to sequences of images.

3.2.1 Sliding window optimisation

To tackle both of these issues, we implement a sliding window optimisation that jointly considers previously estimated frame rotations and acts as a support for the current rotation estimate. To model this non-linear joint optimisation, we employ a simple factor graph [11] that consists of factor nodes $f \in \mathcal{F}$ and variables $\mathcal{X} = \{x_t, \ldots, x_{t-n}\}$, where t refers to the latest variable and n defines the length of the sliding window. The joint distribution $p(\mathcal{X})$ is represented as a product of all factors in the graph which, when considering a Gaussian factor graph, can be solved by the following minimisation:

$$\hat{\mathcal{X}} = \arg\min_{\mathcal{X}} \sum_i ||h_i(\mathcal{X}_i) - z_i||^2_{\Sigma_i} \tag{10}$$

where $\mathcal{X}_i$ represents the clique of variables corresponding to the factor f_i, $h_i(\cdot)$ represents a function that predicts a measurement based on the state of input variables, and z_i represents some observed measurement.

For our problem, each variable $x_i \in \mathcal{X}$ represents an $SO(3)$ rotation $\mathbf{R}_i \in \mathcal{R}$, and 3 types of factors exist to constrain the problem. Firstly, $f^z(\mathbf{R}_i)$, is a prior factor on each variable that is set to the rotation estimated from the single-frame estimation problem, $\mathbf{Z}_i \in SO(3)$. The second factor, $f^s(\mathbf{R}_i, \mathbf{R}_j)$, is a smoothness factor which enforces that adjacent frames should have a similar rotation. Finally, another prior factor $f^p(\mathbf{R}_i)$ is added to the oldest frame in the sliding window which represents the marginalised state of the oldest variable in the previous sliding window optimisation, $\mathbf{R}_p$. This multi-frame factor graph is shown pictorially in Fig. 3. The minimisation for our problem is therefore given by:

$$\hat{\mathcal{R}} = \arg\min_{\mathcal{R}} \sum_{(i,j)\in\mathcal{F}^s} ||\mathbf{R}_i \ominus \mathbf{R}_j||^2_{\Sigma_i^s} + \tag{11}$$

$$\sum_{i\in\mathcal{F}^z} ||\mathbf{R}_i \ominus \mathbf{Z}_i||^2_{\Sigma_i^z} + \sum_{i\in\mathcal{F}^p} ||\mathbf{R}_i \ominus \mathbf{R}_p||^2_{\Sigma_i^p} \tag{12}$$

where $\ominus$ denotes the vector increment (defined on the tangent space of the right hand variable) between two rotations via the logarithmic map. For ease of implementation, we use the popular sensor fusion library GTSAM [10] to optimise the multi-frame factor graph based on the definitions above.

Once $\hat{\mathcal{R}}$ has been computed, the latest frame's rotation can then be fed back into the single frame optimisation as initialisation for the subsequent frame i.e. $\mathbf{R}_{t+1}^{\text{init}} = \hat{\mathbf{R}}_t \in \hat{\mathcal{R}}$

3.2.2 Uncertainty and robust estimation

To address the issue of non-Manhattan frames, the covariance estimate from Eq. (9) can be directly applied to each measurement factor (Σ^z in Eq. (11)), since Eq. (8) is defined around the global MF upon which we are optimising. Subsequently, frames which exhibit strong Manhattan normals will have a greater influence on the result, stabilising rotation estimates for non-Manhattan frames.

For the smoothness factors, the covariance Σ^s is set to an isotropic covariance $\lambda\mathbf{I}_3$, where λ is a tuning parameter that defines how strongly the smoothness constraint is enforced. The covariance for the prior measurement Σ^p, is automatically defined by the marginalisation of the last variable $\mathbf{R}_{t-n}$ in the previous iterations window.

In order to reduce the influence of outlier measurements due to dropped frames, poor normal predictions, and incorrect local minima from the single frame optimisation process, we also apply robust factors to all prior measurement factors f^z. We leverage the Huber cost function which represents a Gaussian energy for small residuals, but transitions to a linear function for large residuals. This effectively dampens the influence of measurements that grossly disagree with the smoothness model between variables.

Table 1. Quantitative evaluation on ICL-NUIM and TUM RGB-D [deg]. The best single image RGB method is **bold** and the second-best is underlined. *Italicised* averages don't include failed runs and are not considered for best accuracy metric.

Sequences	Ours	RGB Single Image Methods					RGB VO	RGB-D Methods		
		RMFE*[15]	RTMF*[44]	ES*[42]	H-VP[40]	H-VP†[40]	ORB[38]	GOME[24]	Compass[26]	E-Graph[33]
Office 0	4.99	4.99	4.97	5.21	1.24	**1.00**	0.60	5.12	0.37	0.11
Office 1	3.87	89.18	44.59	3.90	**3.45**	4.79	×	×	0.37	0.22
Office 2	2.38	3.35	2.36	41.99	**0.91**	0.97	0.69	6.67	0.38	0.39
Office 3	**2.72**	2.84	41.98	2.87	3.69	3.20	2.53	5.57	0.38	0.24
Living 0	8.43	8.36	**8.25**	11.53	×	×	0.35	×	0.31	0.44
Living 1	**3.58**	91.95	3.81	14.04	7.10	6.37	×	8.56	0.38	0.24
Living 2	**2.39**	2.45	2.50	2.44	4.17	3.84	0.57	8.15	0.34	0.36
Living 3	**5.38**	5.64	5.58	57.62	7.23	6.56	0.84	×	0.35	0.36
ICL-NUIM Avg.	**4.22**	26.10	14.25	17.45	*3.97*§	*3.82*§	*0.85*§	*6.81*§	0.36	0.30
Struc notex	4.61	**4.55**	4.94	7.96	19.10	20.82	×	4.07	1.96	4.46
Struc tex	**3.03**	3.10	3.18	3.14	11.13	8.25	0.37	4.71	2.92	0.60
Large cabinet	4.54	**4.30**	4.60	5.20	7.57	6.82	1.13	3.74	2.04	1.45
Cabinet	**5.41**	40.15	6.27	70.39	33.38	20.90	×	2.59	2.48	2.47
Long office	**5.62**	5.78	5.98	46.74	14.49	12.73	7.86	×	1.75	-
Nostruc notex	**6.93**	54.46	27.52	30.77	×	×	×	×	×	-
Nostruc tex	28.94	24.16	63.62	**11.58**	31.21	27.81	17.42	×	×	-
TUM RGBD Avg.	**8.44**	19.50	16.59	25.11	*19.48*§	*16.22*§	*6.70*§	*3.78*§	*2.12*§	2.25

*Reimplemented using our predicted normals †Hybrid VP without the gravity prior §Averages excluding failure sequences

However, by maintaining these measurements in the factor graph, a genuine large change in rotation will be properly estimated after a few frames of consistent measurements.

4. Experiments

Following previous methods [17, 26, 33], we evaluate the accuracy and robustness of our method on ICL-NUIM [18] and TUM RGB-D [45] which cover synthetic and real indoor scenes, respectively. To assess the wider performance in challenging real-world scenarios, we also evaluate the performance on ScanNet [8]. ScanNet images have a significant amount of noise and blur as they were captured using hand-held cameras in poorly lit environments. The scenes are also cluttered with many objects that violate the Manhattan World assumption. We do not discard such scenes and evaluate on all 100 test sequences.

We then also show how our method can be used for up-vector estimation and compare against methods specifically designed for this purpose. Further discussion surrounding using U–ARE–ME for ground segmentation as well as horizon estimation can be found in the supplementary material.

To measure accuracy, each frame is fed into the algorithm sequentially and the estimated rotation is recorded after each frame. The rotations are then aligned with the relevant ground truth so that methods that do not estimate a specific world alignment can also be compared with the MW-based methods. The metric used for accuracy is the average rotation error (ARE) and is given by

$$\text{ARE} = \cos^{-1}\left(\frac{\text{tr}(\mathbf{R}_{\text{gt}}^{-1}\hat{\mathbf{R}}) - 1}{2}\right) \tag{13}$$

where $\hat{\mathbf{R}}$ is the rotation estimate and $\mathbf{R}_{\text{gt}}$ is the ground truth.

We compare our approach to various monocular rotation estimation methods. Since we are the first method to directly use learnt normals from monocular images, we compare against other normal optimisation methods designed for RGB-D sensors replacing depth with our predicted normals: RMFE [15], RTMF [44], ES [42]. This is to show the value of our novel cost function.

We also compare against a recent hybrid vanishing point method (H-VP) for uncalibrated images [40], which uses a gravity prior to extract more potential VPs (tested with and without the prior). As in the original paper, since H-VP only extracts the Manhattan VPs with no specific X,Y,Z assignment, rotation axes are assigned based on the nearest axis to the ground truth rotation (which we don't rely on). This would not normally be possible on in-the-wild data, giving H-VP perfect temporal consistency across the sequence – thus is comparable to our multi-frame approach.

Furthermore, direct comparisons are drawn with the popular monocular SLAM system ORB-SLAM [38]. Whilst this is not a single image rotation estimation method and requires accurate intrinsics, it is still one of the most widely used methods for obtaining accurate real-time odometry and provides a challenging benchmark from which to draw conclusions of our work.

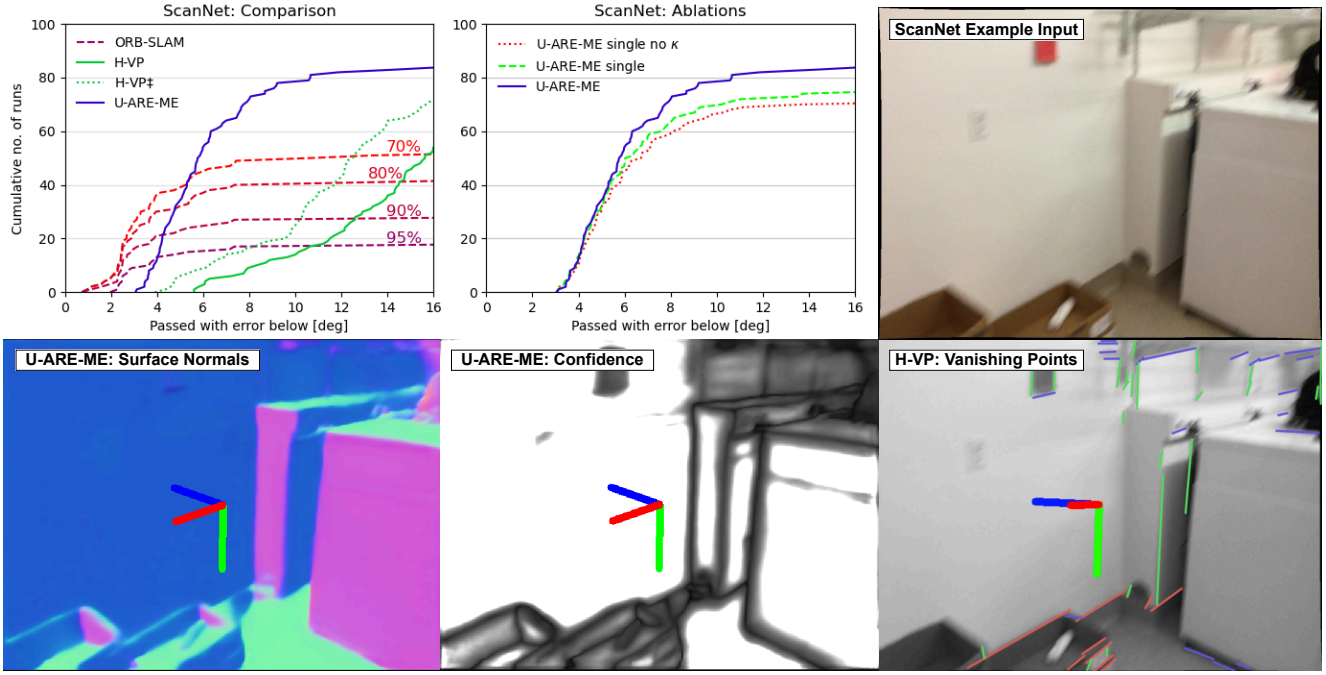

Figure 4. **(Top left)** U–ARE–ME accuracy comparison on 100 sequences from the ScanNet dataset. Lines show cumulative number of runs below a certain accuracy threshold. Percentage pass rate is shown for ORB-SLAM, whereby at least X% of frames per sequence must contain a valid rotation estimate (this includes any initialisation and loss of tracking). **(Top middle)** ablation study experiments. The blue line shows the results of the full pipeline. 'single' means that multi-frame optimisation is disabled and 'no κ' means that the uncertainty weighting in the cost function is removed. **(Top right)** example blurred low-texture image from ScanNet, indicative of the challenging scenes contained within. **(Bottom row)** example output from U–ARE–ME and H-VP. Estimated MW rotation from each algorithm are shown centrally, and in the bottom-middle image white indicates a high confidence.

Lastly, several RGB-D methods (GOME [24], Compass [26] and E-Graph [33]) are also used in our comparisons so that we can give context on how accurate our system is compared to methods that require a depth map.

4.1. Results from ICL-NUIM and TUM RGB-D

The overall results for both the ICL-NUIM and TUM RGB-D datasets are shown in Tab. 1. U–ARE–ME can accurately estimate a valid trajectory of rotations in 14/15 sequences and has on average the best accuracy of the non-SLAM based monocular methods. Whilst RTMF, RMFE and ES sometimes show the same accuracy as the proposed method, the lack of temporal consistency means that they often shift into a globally inconsistent MF and in many of the sequences show large errors.

In some sequences the presence of strong line features mean that VP methods retrieve very accurate results (e.g. office scenes due to ceiling tiles), however in less textured scenes and with lower quality images, the line segment detectors of VP methods struggle to acquire enough features to perform RANSAC reliably. As we use a dense normal predictor we don't suffer in these texture-less scenes.

Comparing to the RGB-D methods, the proposed method is often better than GOME despite only using RGB images, and whilst we are worse than Compass and E-Graph for the synthetic ICL-NUIM sequences (which have perfect depth), we achieve comparable accuracy in some real-world TUM sequences.

Comparing to ORB-SLAM, for the ICL-NUIM sequences in which ORB-SLAM does not fail, we find that ORB-SLAM is significantly better than all the RGB methods and is even on par with the RGB-D methods. However, ORB-SLAM may be unable to initialise or lose tracking in texture-less scenes. Such sparse feature-based approaches also suffer from image degradation. To provide a further analysis on how different approaches would perform in challenging, real-world scenarios, we provide the results on ScanNet in the following section.

4.2. Results from ScanNet

The ScanNet suite provides a large set of real-world sequences from which we can draw better conclusions about the generalisability of our system. We therefore further test U–ARE–ME on all 100 test sequences and compare again to H-VP and ORB-SLAM. We then also perform an ablation study on the proposed method.

Fig. 4 (top left) shows the cumulative number of runs where the mean angular accuracy for a particular sequence is below a threshold. Since ORB-SLAM is a multi-view system and therefore will never produce rotation estimates for every single frame, we show the results of ORB-SLAM at different success rates e.g. 70% defines that at least 70% of frames per sequence need valid rotation estimates (the missing frames being from either initialisation or loss of tracking). In this regard, we allow ORB-SLAM to lose tracking and do not consider this an outright failure.

The results show that U–ARE–ME has a much higher robustness to the ScanNet sequences and will estimate rotations with an accuracy $< 10°$ in 80% of the sequences. Comparing this to ORB-SLAM with a 95% frame threshold which only successfully achieves $< 10°$ in 17% of the sequences. For higher accuracy $< 3°$, we find that ORB-SLAM is more capable in some sequences (10%) whereas the proposed method achieves at best $3°$. We argue that this is primarily caused by the accuracy of the surface normal predictions, e.g. the state of the art [2] reports a mean normal error of $16.2°$ on ScanNet, and therefore as normal predictions improve so should our results.

H-VP similarly struggles on ScanNet and is actually hindered by the gravity prior – most likely due to the wide range of rotations that violate the gravity assumption. Most of the ScanNet sequences are indoor scenes containing many blank walls which do not provide strong line features or point features. VP methods struggle in these scenes, as shown in Fig. 4 where shadow lines are mistakingly classed as MW vanishing points. Our normal network however predicts these regions as planar and also predicts a low confidence (shown in black) so robustly ignores these regions during optimisation. More generally, in low-texture scenes normal prediction networks can rely on subtle lighting cues at the intersection of walls to determine the orientation of these large planar surfaces.

The ablation study Fig. 4 (top middle), shows that the overall reliability of the system improves as firstly, the uncertainty-weighted normals reject non-Manhattan pixels within the image, and more so secondly, the multi-frame optimisation provides a consistent global MF which anchors the solution across the sequence.

4.3. Up-vector Estimation

Estimating the 'upward' direction of a given image is a task that has been tackled by many recent works and can naturally also be extracted from our method by simply dropping the yaw component of the rotation matrix. Recent neural network approaches have shown success in estimating camera parameters such as the up-vector. CTRL-C [30] proposes an end-to-end transformer approach to combine detected vanishing points with learned features. Perspective Fields (PF) [23] predicts the per-pixel information about the camera parameters, and demonstrated the use of the up-vector for AR effects such as compositing rainfall and 3D objects into the scene.

Table 2. Accuracy of the estimated up-vector [°]. Note that PF [23] and CTRL-C [30] were trained specifically to estimate the up-vector.

Sequence	U–ARE–ME	PF	CTRL-C
Living 0	**7.53**	9.90	12.14
Living 1	**2.67**	3.17	13.74
Living 2	**1.77**	4.67	9.71
Living 3	**4.01**	10.84	7.67
Office 0	**3.95**	5.52	18.87
Office 1	**3.50**	4.65	22.32
Office 2	**1.61**	3.80	15.95
Office 3	**2.13**	3.95	18.47

We compare our method against these baselines on the ICL-NUIM dataset and report the angular difference of the up-vector in Table 2. As these baseline methods only operate on single RGB images, we perform single frame rotation estimation in our method for a fair comparison. Our method outperforms the baselines which were trained on feature-rich image datasets, and is able to more accurately estimate camera rotation in the relatively texture-less scenes from the ICL-NUIM dataset.

5. Conclusion

Motivated by the need to provide extrinsic rotation estimates from *in-the-wild* images and sequences, we have presented U-ARE-ME, an accurate and robust camera rotation estimator that operates on uncalibrated RGB images. The system is capable of outputting estimates for single images and has also been extended to reliably handle multi-frame scenarios. An extensive evaluation has been performed and it has been shown that our method is capable of providing globally consistent multi-frame rotation estimates which rivals the performance of similar methods that leverage accurate depth maps – all whilst remaining real-time. By accounting for the uncertainty and inherent ambiguity of the common MW assumption, we are also capable of providing accurate results on scenes that superficially appear to not contain structural regularities, and where other 2D feature-based methods often fail.

6. Acknowledgements

This research has been supported by the EPSRC Prosperity Partnership Award with Dyson Technology Ltd. We are grateful to the members of the Dyson Robotics Lab for their insightful views and advice.

References

[1] Ibraheem Alhashim and Peter Wonka. High quality monocular depth estimation via transfer learning. *arXiv preprint arXiv:1812.11941*, 2018. 4

[2] Gwangbin Bae and Andrew J. Davison. Rethinking inductive biases for surface normal estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2, 4, 8, 1

[3] Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. In *ICCV*, 2021. 2

[4] Jean-Charles Bazin and Marc Pollefeys. 3-line ransac for orthogonal vanishing point detection. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4282–4287. IEEE, 2012. 2

[5] Jean-Charles Bazin, Yongduek Seo, Cédric Demonceaux, Pascal Vasseur, Katsushi Ikeuchi, Inso Kweon, and Marc Pollefeys. Globally optimal line clustering and vanishing point estimation in manhattan world. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 638–645, 2012. 3

[6] Ruojin Cai, Bharath Hariharan, Noah Snavely, and Hadar Averbuch-Elor. Extreme rotation estimation using dense correlation volumes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14566–14575, 2021. 2

[7] James Coughlan and Alan L Yuille. The manhattan world assumption: Regularities in scene statistics which enable bayesian inference. In *NeurIPS*, 2000. 2, 3

[8] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 6

[9] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM Transactions on Graphics (ToG)*, 36(4): 1, 2017. 1

[10] Frank Dellaert and GTSAM Contributors. GTSAM. https://github.com/borglab/gtsam, 2022. 5

[11] Frank Dellaert, Michael Kaess, et al. Factor graphs for robot perception. *Foundations and Trends® in Robotics*, 6(1-2): 1–139, 2017. 5

[12] Weiye Deng, Xiaoping Chen, and Jingwei Jiang. A staged real-time ground segmentation algorithm of 3d lidar point cloud. *Electronics*, 13:841, 2024. 2

[13] Wael Elloumi, Sylvie Treuillet, and Remy Leconge. Real-time camera orientation estimation based on vanishing point tracking under manhattan world assumption. *Journal of Real-Time Image Processing*, 2014. 2

[14] Yasutaka Furukawa, Brian Curless, Steven M. Seitz, and Richard Szeliski. Manhattan-world stereo. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1422–1429, 2009. 2

[15] Bernard Ghanem, Ali Thabet, Juan Carlos Niebles, and Fabian Caba Heilbron. Robust manhattan frame estimation from a single rgb-d image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3772–3780, 2015. 6

[16] Bernard Ghanem, Ali Thabet, Juan Carlos Niebles, and Fabian Caba Heilbron. Robust manhattan frame estimation from a single rgb-d image. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3772–3780, 2015. 1, 2, 3

[17] Ruibin Guo, Keju Peng, Dongxiang Zhou, and Yunhui Liu. Robust visual compass using hybrid features for indoor environments. *Electronics*, 8(2):220, 2019. 6

[18] A. Handa, T. Whelan, J.B. McDonald, and A.J. Davison. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In *IEEE Intl. Conf. on Robotics and Automation, ICRA*, Hong Kong, China, 2014. 6

[19] Richard I. Hartley and Fredrik Kahl. Global optimization through searching rotation space and optimal estimation of the essential matrix. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007. 3

[20] Ming He, Chaozheng Zhu, Qian Huang, Baosen Ren, and Jintao Liu. A review of monocular visual odometry. *The Visual Computer*, 36(5):1053–1065, 2020. 1

[21] B.K.P. Horn. Extended gaussian images. *Proceedings of the IEEE*, 72(12):1671–1686, 1984. 3

[22] Intel. Intel/Realsense. https://www.intelrealsense.com/sdk-2/, 2024. 2

[23] Linyi Jin, Jianming Zhang, Yannick Hold-Geoffroy, Oliver Wang, Kevin Matzen, Matthew Sticha, and David F. Fouhey. Perspective fields for single image camera calibration. *CVPR*, 2023. 8

[24] Kyungdon Joo, Tae-Hyun Oh, Junsik Kim, and In So Kweon. Robust and globally optimal manhattan frame estimation in near real time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(3):682–696, 2019. 3, 6, 7

[25] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NeurIPS*, 2017. 2

[26] Pyojin Kim, Brian Coltin, and H Jin Kim. Indoor rgb-d compass from a single line and plane. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4673–4680, 2018. 6, 7

[27] Jana Košecká and Wei Zhang. Video compass. In *Computer Vision — ECCV 2002*, pages 476–490, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg. 2

[28] Florian Langer, Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. Sparc: Sparse render-and-compare for cad model alignment in a single rgb image. *arXiv preprint arXiv:2210.01044*, 2022. 2

[29] David C. Lee, Martial Hebert, and Takeo Kanade. Geometric reasoning for single image structure recovery. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2136–2143, 2009. 2

[30] Jinwoo Lee, Hyunsung Go, Hyunjoon Lee, Sunghyun Cho, Minhyuk Sung, and Junho Kim. CTRL-C: Camera calibration TRansformer with Line-Classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 8

[31] Jeong-Kyun Lee and Kuk-Jin Yoon. Real-time joint estimation of camera orientation and vanishing points. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1866–1874, 2015. 2

[32] Haoang Li, Ji Zhao, Jean-Charles Bazin, Pyojin Kim, Kyungdon Joo, Zhenjun Zhao, and Yun-Hui Liu. Hong kong world: Leveraging structural regularity for line-based slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 3

[33] Yanyan Li and Federico Tombari. E-graph: Minimal solution for rigid rotation with extensibility graphs. In *European Conference on Computer Vision*, pages 306–322. Springer, 2022. 6, 7

[34] Yanyan Li, Nikolas Brasch, Yida Wang, Nassir Navab, and Federico Tombari. Structure-slam: Low-drift monocular slam in indoor environments. *IEEE Robotics and Automation Letters*, 5(4):6583–6590, 2020. 2

[35] Shikun Liu, Linxi Fan, Edward Johns, Zhiding Yu, Chaowei Xiao, and Anima Anandkumar. Prismer: A vision-language model with an ensemble of experts. *arXiv*, 2023. 2

[36] Yunze Man, Xinshuo Weng, Xi Li, and Kris Kitani. Groundnet: Monocular ground plane normal estimation with geometric consistency. *Proceedings of the 27th ACM International Conference on Multimedia*, 2018. 2

[37] Carlos Morimoto and Rama Chellappa. Evaluation of image stabilization algorithms. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, pages 2789–2792. IEEE, 1998. 1

[38] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015. 2, 6

[39] Alvaro Parra Bustos, Tat-Jun Chin, Anders Eriksson, Hongdong Li, and David Suter. Fast rotation search with stereographic projections for 3d registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2227–2240, 2016. 3

[40] Rémi Pautrat, Shaohui Liu, Petr Hruby, Marc Pollefeys, and Daniel Barath. Vanishing point estimation in uncalibrated images with prior gravity direction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14118–14127, 2023. 2, 3, 6

[41] Grant Schindler and Frank Dellaert. Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments. In *CVPR*, 2004. 3

[42] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, 2012. 1, 2, 3, 6

[43] Julian Straub, Guy Rosman, Oren Freifeld, John J. Leonard, and John W. Fisher. A mixture of manhattan frames: Beyond the manhattan world. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3770–3777, 2014. 3

[44] Julian Straub, Nishchal Bhandari, John J. Leonard, and John W. Fisher. Real-time manhattan world rotation estimation in 3d. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1913–1920, 2015. 1, 2, 6

[45] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, 2012. 6

[46] Markus Tatzgern, Raphael Grasset, Denis Kalkofen, and Dieter Schmalstieg. Transitional augmented reality navigation for live captured scenes. In *2014 IEEE Virtual Reality (VR)*, pages 21–26, 2014. 1, 2

[47] Camillo Taylor and Anthony Cowley. Parsing indoor scenes using rgb-d imagery. In *Proceedings of Robotics: Science and Systems*, 2012. 3

[48] YouTube. 'Race The Tube - London Parkour POV'. https://www.youtube.com/watch?v=tXMPRK2LQAE, 2018. 4

[49] YouTube. 'Walking in the Rain Tokyo, Japan (Relaxing Binaural Thunderstorm Sounds for Sleep) 4k ASMR'. https://www.youtube.com/watch?v=Et7O5-CzJZg, 2019. 3

[50] YouTube. 'A Walk Around Orison City - Star Citizen Alpha 3.14 gameplay (no commentary)'. https://www.youtube.com/watch?v=G3gqBaqDSaE, 2021. 3

[51] Guangyao Zhai, Dianye Huang, Shun-Cheng Wu, HyunJun Jung, Yan Di, Fabian Manhardt, Federico Tombari, Nassir Navab, and Benjamin Busam. Monograspnet: 6-dof grasping with a single rgb image. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1708–1714. IEEE, 2023. 2

[52] Yutong Zhang, Yan Ding, Jianmei Song, Jiaxin Li, and Hua-Liang Wei. A fast manhattan frame estimation method based on normal vectors. *Journal of Field Robotics*, 39(5):557–579, 2022. 3, 4