2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Relative Pose Estimation and Planar Reconstruction via Superpixel-Driven Multiple Homographies

Xi Wang, Marc Christie, Eric Marchand

***Abstract*— This paper proposes a novel method to simultaneously perform relative camera pose estimation and planar reconstruction of a scene from two RGB images. We start by extracting and matching superpixel information from both images and rely on a novel multi-model RANSAC approach to estimate multiple homographies from superpixels and identify matching planes. Ambiguity issues when performing homography decomposition are handled by proposing a voting system to more reliably estimate relative camera pose and plane parameters. A non-linear optimization process is also proposed to perform bundle adjustment that exploits a joint representation of homographies and works both for image pairs and whole sequences of image (vSLAM). As a result, the approach provides a mean to perform a dense 3D plane reconstruction from two RGB images only without relying on RGB-D inputs or strong priors such as Manhattan assumptions, and can be extented to handle sequences of images. Our results compete with keypoint-based techniques such as ORB-SLAM while providing a dense representation and are more precise than direct and semi-direct pose estimation techniques used in LSD-SLAM or DPPTAM.**

I. Introduction

Nowadays, many visual tracking, pose estimation and SLAM (Simultaneous Localization And Mapping) algorithms are competing to achieve better performance – precision, accuracy, computation time – in both indoor and outdoor scenarios [1], [2], [3]. Some algorithms rely on the direct alignment of the intensity between images in order to generate a dense pixel-wised mapping [2], while others exploit keypoints or similar low-level image features (*e.g.,* lines, patterns) to achieve more precise and robust camera poses [1]. It seems a trade-off is inevitable between the sparse methods (*e.g.,* keypoints-based method) and the dense methods (which compute camera poses by aligning pixel intensities): the former is more robust under variant environment and more compatible with Bundle Adjustment techniques and the latter yields a more applicable map with denser information. Though some hybrid systems are proposed to balance the advantages of both systems [4], the topic keeps attracting researchers' attention and requires further explorations.

Intermediate features extracted from images or from low-level features can also be exploited. Typically planes are ubiquitous geometric features in human-crafted environments and objects, and exhibit good characteristics for tasks such as pose estimation and visual tracking: planes are widely studied, offer a light parameterisation, are robust against environmental variance w.r.t. spatially isolated keypoints, and most importantly, planes are easy to compute from image pairs via homography constraints. Many contributions also exploit planar assumptions in a variety of vision-based robotic applications [5], [6]. Homography estimation is indeed convenient and simple whilst the scene has a dominant plane such as ground or ceiling. However in the real world, the dominant plane assumption does not always hold as it can be occluded or the scene can be composed of multiple planar structures such as indoor environments or outdoor city landscapes.

In this paper, we propose a novel multi-homography based pose estimation method via superpixel-driven RANSAC which achieves simultaneously the camera pose and a dense planar mapping from a pair of color images. We also show that this method can be integrated within a vSLAM pipeline. Our contributions are: 1) a novel RANSAC technique for multiple homographies detection problem combining information from superpixels and keypoints 2) a voting-based ambiguity-free multiple homographies decomposition process for pose estimation, and 3) a non-linear optimization pose refiner for both image pair and a sequence of images (vSLAM).

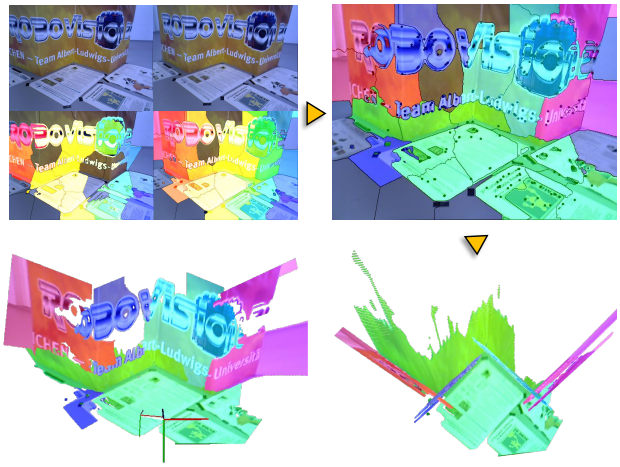

Fig. 1: From two RGB images of a monocular camera (left up), we propose a superpixel-driven technique to estimate simultaneously a relative camera pose and a 3D multi-planar map (down) without relying on a Manhattan assumption. In right up image, the different colors represent different 3D planes estimated from the images, using a novel approach we refer to as Winner-takes-all RANSAC.

Author are with Univ Rennes, Inria, CNRS, Irisa, France.
Email:{xi.wang}@inria.fr
{marc.christie, eric.marchand}@irisa.fr

II. Related Work

For the case of dominant planar scenes, [7], [8] developed visual tracking theory and applications. For example, the work of Pirchheim et al. [5] consists of a mobile AR application under the assumption of single planar homography. However, the decomposition ambiguities of the homography matrix seem difficult to resolve using merely a geometric approach [9]. Many works exploit additional information such as: a priori known geometric shapes or combining the information from IMU (Inertial Measurement Unit) not only for eliminating the ambiguities in homography but also improving the precision of pose estimation [10], [11].

Typically, the Manhattan assumption is widely exploited in planar vision tasks [11], [12], [13]. Principally the assumption is that all planes in the environment are perpendicular in 3D, such as typical buildings or standard rooms.

Many planar SLAMs and visual tracking applications exploit RGB-D cameras which are well suited for indoor-environments. By combining available depth information, Kaess [14] proposed a planar SLAM system with a quaternion formulation of 3D plane which improves convergence of optimization under RGB-D environments, and then [15] extended it to a keyframe-based dense planar SLAM with a factor graph map using incremental smoothing and mapping (iSAM). Le and Košecka [16] also combined RGB-D sensor with Manhattan Assumption.

Many contributions on plane segmentation in images are tightly associated with the *superpixel* technique. A superpixel is defined as a group of connected pixels with consistent color or intensity information. Superpixels are usually generated with segmentation methods; typical works include SLIC [17], SEEDS [18] and graph-segmentation superpixel [19].

Concha and Civera [20] are the first who proposed to exploit superpixel techniques in a SLAM system. Their approach uses a Monte Carlo ranking to achieve the correspondence and initial 3D pose of superpixels. Then an optimization is performed to refine the plane poses with an already known camera pose estimated separately from a PTAM system. In a more recent work (DPPTAM) [21] they integrate superpixel in a semi-dense tracking system. Plane estimation is achieved by RANSAC and SVD on 3D points from semi-dense tracking. A dense mapping is also designed with found superpixels information.

Inspired from [20], [21], we propose to exploit superpixels information for estimating relative camera pose and multi-planes structure simultaneously from two images (see Fig. 1). Such a system requires 1) the capacity of extracting multiple planes from two images; 2) the ability of eliminating ambiguities in homography decomposition; and 3) the possibility to combine the homography representations with the optimization framework of pose estimation for better performance;

III. Overview

The method we propose is composed of the following modules (see pipeline in Fig. 2 for the overview): (a) superpixelization and tracking process: extracting and matching corresponding superpixel information from a pair of images. (b) superpixel-driven RANSAC: detecting multi-planar structures in a robust way, (c) multiple homographies decomposition: computing camera pose and eliminating ambiguities in homographies, and (d) non-linear refiner: applying a Bundle Adjustment-like optimization camera and plane refiner for both image pairs and a sequence of images. All the modules are detailed in the following section respectively.

IV. Superpixel extracting and Tracking

Our work builds on the idea that superpixels are good initial guesses of planar regions in images for that they usually show strong chromatic consistency and spatial continuity at pixel level. We exploit superpixel spatial relations (adjacency) as well as local keypoint descriptors to perform a matching of superpixels in two different frames.

More specifically, we first superpixelize two frames I_i, I_{i+1} with SLIC [17] and obtain two sets of regions respectively, denoted by $V^i = \{V_k^i\}$ with $k = 1..K$, K being the total number of superpixels extracted from *ith* image. We then exploit a graph structure to conserve the information of adjacency between superpixels. An unidirectional unweighted graph is proposed: $\mathbf{G_i} = (V^i, E^i)$ where V^i the vertices are the set of superpixels in I_i and E presents their adjacency (equal to 1 when two superpixel regions are adjacent).

Once the segmentation is performed, a superpixel tracking system is required for matching superpixel regions between two frames. We undertake this step by matching keypoint descriptors (*e.g.,*ORB [22]) extracted from the each superpixel regions. A cross checked greedy matching policy is adopted during this procedure.

In contrast with common superpixel tracking tasks [23] which concentrate mostly on re-identification of moving objects from static background, SLAM and camera pose estimation works usually hold the assumption of static environment. Based on this assumption, we then propose a superpixel tracking method between two images: we search for the highest matched number of keypoints between not only two superpixel regions but also their neighbor superpixels in graph structure as in a static environment each superpixel should hold a relatively rigid local structure w.r.t others. The depth of the neighborhood d_G is represented by a on-graph distance (shortest path) used to manipulate the range of neighbor area. We denote these neighborhood regions around vertex V_k as $N^{d_G}(V_k)$, as also mentioned in Section V-C:

$$N^{d_G}(V_k) = \{V_j \in \{V\} : d(V_j, V_k) \leq d_G) \qquad (1)$$

As displayed in Fig. 2 and throughout the paper, matched superpixel between image pairs are highlighted with the same color.

V. Multi-Homography Estimation

A. Homography and RANSAC

In a planar environment, the homography matrix ${}^2\mathbf{H}_1 \in \mathbb{SL}(3)$ can be used to describe the transformation of one

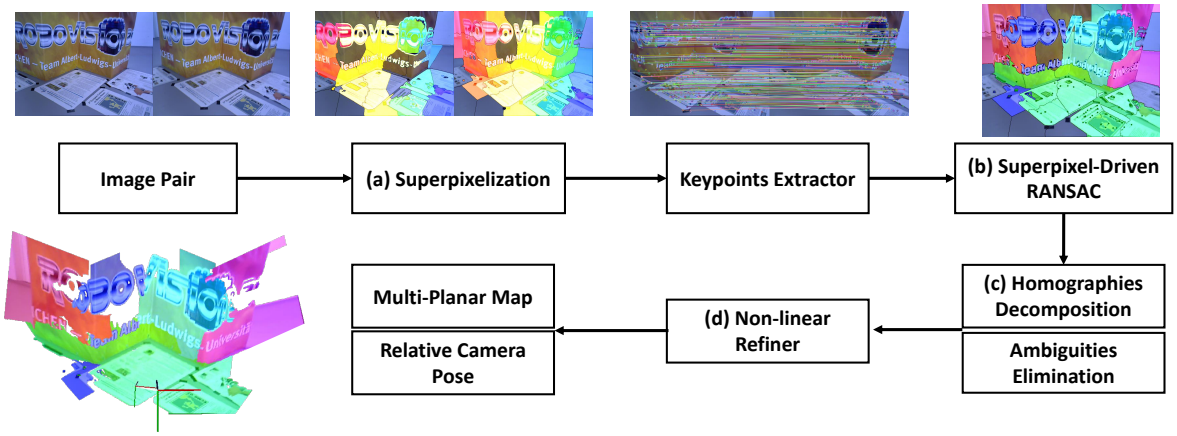

Fig. 2: Pipeline of our system which generates a relative camera pose and a 3D multi-planar map from a pair of color images.

plane between two images I_1 and I_2. When the intrinsic calibration matrix of the camera $\mathbf{K}$ is known, all pixels extracted from I_1 and I_2 can be inversely projected as normalized three dimensional coordinates denoted as: $\mathbf{p_1}$ and $\mathbf{p_2} \in \mathbb{R}^3$. Therefore the homography matrix constrains them with the following relation:

$$\mathbf{p_2} = {}^2\mathbf{H}_1\mathbf{p_1}$$

A homography matrix is composed of a rotation matrix ${}^2\mathbf{R}_1 \in \mathbb{SO}(3)$, a translation vector ${}^2\mathbf{t}_1 \in \mathbb{R}^3$ as well as a normal vector in I_1, defined as $\mathbf{n}_1 = (a, b, c)^\top \in \mathbb{R}^3$. A plane can be therefore described as $\mathbf{p}^\top\mathbf{n}_1 = d$, where $\mathbf{p} \in \mathbb{R}^3$ are three dimensional points on plane and d is the orthogonal distance from the plane to the origin:

$${}^2\mathbf{H}_1 = {}^2\mathbf{R}_1 + \frac{{}^2\mathbf{t}_1}{d}\mathbf{n}_1{}^\top \tag{2}$$

Multiple methods are available to compute the homography matrix ${}^2\mathbf{H}_1 \in \mathbb{SL}(3)$ from a pair of images. The *RANdom SAmple Consensus* (RANSAC) method [24] relies on two matched sets of keypoints $\{\mathbf{p_1}\}, \{\mathbf{p_2}\}$ in two frames and a Direct Linear Transform (DLT) technique [25]. Its goal is to divide the data in two sets: the set of inliers (*i.e.* Consensus-Set (CS)) and the outliers (spurious data).

We first introduce some notations used in RANSAC. We denote $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ as the set of all matched *pairs* of keypoints from I_1 and I_2: $\mathbf{x} = \{\mathbf{p_1}, \mathbf{p_2}\}$. In our case we consider the homography $\mathbf{H}$ as the model to estimate. We then define:

1) **Minimal Sample Set:** M: the minimum number of pairs of points to estimate a homography, which is 4 for one homography.
2) **Sampling Procedure**: $\mathcal{S}$: $\mathcal{D} \rightarrow \mathcal{D}^M$, it samples all subsets in $\mathcal{D}$ s.t. their cardinality equals M. The sampling is usually done by randomly selecting 4 points to compute a $\mathbf{H}$.
3) **Model Estimation Function**: $\mathcal{E}$: $\mathcal{D}^M \rightarrow \mathbf{H}$. In homography, DLT estimates $\mathbf{H}$ from 4 non-degenerated points.
4) **Inlier Threshold** ϵ: A threshold to determine inlier, here we take the distance between the point and the reprojection of it's matched pair: $(\mathbf{p_2} - {}^2\mathbf{H}_1\mathbf{p_1})^2$.

Using these definitions, one may reword the RANSAC process as an algorithm which searches for the largest Consensus-Set by randomly sampling M and evaluating their consensus via a measure function with a threshold ϵ.

B. Multi-Model RANSAC

Though RANSAC is proven to be efficient when extracting the principal plane in a scene, many applications display cases where dominant planes are occluded, and multiple planes with similar surfaces are visible. As multiple instances of same model occur in a dataset (*e.g.,*multiple planes), RANSAC suffers not only from *gross outliers* (pure noise, *e.g.,* wrong matches of keypoints) but also from *pseudo-outliers* [26]: outliers to the structure of interest but inliers to a different structure. To solve such multi-model estimation problems (*i.e.* searching for multiple planes), many RANSAC-like algorithms have been proposed such as Sequential RANSAC [27], [28] and [29].

Sequential RANSAC consists of applying RANSAC to a multi-model dataset in an iterative fashion. For each iteration of RANSAC, the found inliers (Consensus-Set) are removed from the dataset. While the sequential nature tends to be influenced by pseudo-outliers [29], one wrong estimation of previous iteration may lead into mistakes in the following ones. To alleviate this false estimation, Kanazawa's sampling technique [27] is widely applied and proven efficient by sampling in a local proximity w.r.t the previous chosen data point (*e.g.,*, by Gaussian distribution) instead of randomly choosing in all dataset: $\mathbf{p} \sim \mathcal{N}(\boldsymbol{p_0}, \boldsymbol{\Sigma})$, describes the probability to choose point $\mathbf{p}$ under the condition that the previous chosen one is $\mathbf{p_0}$ and the sampling range is manipulated by Σ.

Another issue with multi-model estimation is redundancy estimation. A same model may be estimated multiple times as the inlier-removing procedure fails to totally clear out the *pseudo-outliers* of previous detected model (usually because the threshold ϵ is ill-chosen or the data experiences a

heavy unbalance), so the rest *pseudo-outliers* of previous model can still form a similar model which outnumbers the CS over other models. Moreover, the rest *pseudo-outliers* implicitly increases the outlier ratio along the iterations of the sequential procedure and deteriorates the estimation.

C. Superpixel-Driven Winner-Takes-All RANSAC

To address these issues, we propose a Winner-Takes-All RANSAC which is inspired by [27] but benefits from the superpixel information to address the false detection and redundancy estimation problems simultaneously. We exploit superpixels for their relative coplanarity: we assume all information inside a superpixel should be relatively coplanar, as they share local proximity and color similarity. These coplanarity regions play the role of the sampling range Σ in the Kawazana sampling. Instead of an isotropic Σ decided empirically for all datasets, we use directly the regions of superpixel as an adaptive sampling range and even avoid the computation of the conditional probability: *e.g.,*by only selecting points in one superpixel or its neighbor in certain on-graph distance $\mathbf{N}^{d_G}(V_k)$ (see Eq (1)).

We present some notations for the sake of clarity.

1) **Superpixel Cluster Map**: $\mathbf{C}$: A map returns the superpixel label from a pixel in the image. $\mathbf{C} : \Omega \subset \mathbb{N}^2 \rightarrow \mathbb{N}$
2) **Superpixel Neighbor Sampling**: $\mathcal{S}_{\mathcal{N}}(\mathbf{D}, \mathbf{G}, d_G)$: A sampling method which chooses M (4 for homography) pairs of points in following way:
   a) sample first keypoint p_1 uniformly in all dataset.
   b) find the superpixel V_1 of p_1 via Cluster Map $\mathbf{C}$.
   c) sample other $M-1$ points only for data in the subgraph of certain distance d_G w.r.t the V_1: $\{p_2, .., p_M\} = \mathcal{S}(\mathbf{D}(\mathbf{N}^{d_G}(V_1)))$
3) **Ratio of Inliers** ρ: two ratios are defined in this paper, the ratio of all inliers $\bar{\rho}$ and ratio of inliers in each superpixel region ρ_k, defined as the number of inliers over the number of all the data (*e.g.,* extracted keypoint) and a superpixel region respectively.

The WTA-RANSAC algorithm is presented in Algo. 1. The main idea is similar to sequential RANSAC. However, after each iteration of estimation, instead of only removing CS from the dataset, we adopt a *winner-takes-all* policy: invalidate all the points in the superpixel regions where a significant higher inliers ratio (manipulated by q) shows that this superpixel is well dominated by a plane. This allows us to eliminate pseudo-outliers of the detected plane together with its Consensus-Set, as one superpixel is mainly composed by one plane, therefore improves the robustness against false and redundant estimation problem. The stop condition is designed as a ratio threshold of keypoints which have been assigned to a key plane.

Algorithm 1: Winner-Takes-All RANSAC

Data: $\mathbf{D}$, ϵ, M, $\mathbf{G}$, $\mathbf{C}$, q, d_G
// q a parameter controls the level of WTA
Result: S_H
$S_H = \{\}$ // the set of multiple H ;
$S_{ov} = \{\}$ // indicate the occupation of each vertex
while *!StopCondition* **do**
  // single iteration of RANSAC ;
  for *iterations* **do**
    $M = \{\mathcal{S}_{\mathcal{N}}(\mathbf{D}, \mathbf{G}, d_G) : C(p) \notin S_{ov}\}$;
    $H = DLT(M)$ // estimate H ;
    $CS = \{p \in D : E(H, p) < \epsilon, C(p) \notin S_{ov}\}$;
    if $(|CS| > MaxCS)$ **then**
      $BestH, MaxCS = H, |CS|$;
    end
  end
  // Winner-takes-all ;
  for $V_j \in V(G)$ **do**
    if $(\rho_j > q\bar{\rho})$ **then**
      $S_{ov} = S_{ov} \cup j$;
    end
  end
  $S_H = S_H \cup BestH$;
end

VI. Homography Decomposition and Ambiguities Elimination

Once a homography matrix is found, various ways exist to decompose the ${}^2\mathbf{H}_1$ matrix to ${}^2\mathbf{R}_1$, ${}^2\mathbf{t}_1/d$, and $\mathbf{n}_1$ (the translation is up to a scale). Analytically, linear decomposition methods are able to do the job yet generate some ambiguities. Two ambiguities exist even after applying the condition which all points are visible to the camera. Ambiguity can be solved if at least one element among $\mathbf{R}, \mathbf{t}, \mathbf{n}$ is known a priori, *e.g.,*: the normal direction of the floor is known as perpendicular to the up direction, or an IMU is able to indicate the direction of the motion or other measure methods to filter the ambiguity resu1ts.

The main reason of the impossibility in differentiating two ambiguities is that geometrically both of them hold the homography constraint. In the work of [9], the relation of the translation vector between these two ambiguities $\{\mathbf{R}_a, \mathbf{t}_a, \mathbf{n}_a\}$ and $\{\mathbf{R}_b, \mathbf{t}_b, \mathbf{n}_b\}$ is displayed as follows: (for simplicity and under the circumstance of no confusion, we abuse the notation of $\mathbf{R}_a$ to describe ambiguities ${}^2\mathbf{R}_{1a}$ in this section; this is similar for all other notations)

$$\mathbf{t}_b = \frac{\|\mathbf{t}_a\|}{\rho}\mathbf{R}_a\left(2\mathbf{n}_a + \mathbf{R}_a^\top \mathbf{t}_a\right) \tag{3}$$

$$\rho = \left\|2\mathbf{n}_e + \mathbf{R}_e^\top \mathbf{t}_e\right\| > 1; \quad e = \{a, b\} \tag{4}$$

Eq. (3) and (4) show that the difference between $\mathbf{t}_a$ and $\mathbf{t}_b$ is actually influenced by $\mathbf{R}_a$ and $\mathbf{n}_a$. For a case with a single homography, one cannot exploit this relation for selecting a *true* transformation between two images. However, under the condition of the multiple homographies, the Eq. (3) is applied with an extra constraint. All the homographies actually share a common translation and rotation across different planes, as the scene is static while the camera is moving. Our intuition

is then to rely on this shared information to eliminate the decomposition ambiguities.

For each $\mathbf{H}^i$ in the multiple homography scene $\{\mathbf{H}^i\}$, two possible ambiguities can be expressed as the ground truth set $\{{\mathbf{R}_t}^i, {\mathbf{t}_t}^i, {\mathbf{n}_t}^i\}$ and its ambiguity set $\{{\mathbf{R}_f}^i, {\mathbf{t}_f}^i, {\mathbf{n}_f}^i\}$. As all homographies share a unique $\mathbf{t}_t$ and $\mathbf{R}_t$:

$$\mathbf{t}_f^i = \frac{\|\mathbf{t}_t\|}{\rho}\mathbf{R}_t\left(2\mathbf{n}_t^i + \mathbf{R}_t^\top \mathbf{t}_t\right) \tag{5}$$

This means the relation between the real translation $\mathbf{t}_t$ and the ambiguous one ${\mathbf{t}_f}^i$ is only influenced by the normal vector of the plane ${\mathbf{n}_t}^i$. Under the assumption that at least two planes have different normal vectors (which is very common is the multiple planar scene), one could find the real transformation $\{\mathbf{R}_t, \mathbf{t}_t\}$ by simply choosing the common translation vector, and therefore eliminate the ambiguity solutions to the unique one. This procedure is performed by implementing a fairly straightforward voting system on the direction of all translation vectors. By accounting for an angle threshold δ ($15°$ in our implementation) to gather vectors, we select the most voted translation vector and therefore eliminate the ambiguities of each plane.

VII. Non-linear Multi-Plane Refiner

A. Non-linear Refiner of Image Pair

In traditional SLAM systems, Bundle Adjustment techniques are introduced to refine camera poses and landmarks by minimizing the re-projection error on image space of landmarks such as keypoints, lines or other features. Likewise, for the case of homography transformation between two images, previous work (*e.g.,* image-based visual servoing system [30]) have already shown that with a prior known plane, the estimation of the camera pose $\mathbf{q} \in \mathfrak{se}(3) \in \mathbb{R}^6$ (the minimal representation of transformation $\{\mathbf{R}, \mathbf{t}\}$) can be realized via a least square Gauss-Newton optimization process by similarly minimizing the re-projection error E between extracted $(\mathbf{p}_2^n - {}^2\mathbf{H}_1\mathbf{p}_1^n)^2$ being $n = 1..N_p$ as number of keypoints. By now adding the plane parameter $\mathbf{\Pi}_1 = \{\mathbf{n}_1, d\}$ into the system, for a single homography, the optimization framework has the following form:

$$\{\widehat{\mathbf{q}}, \widehat{\mathbf{\Pi}_1}\} = \underset{\mathbf{q},\mathbf{\Pi}_1}{\arg\min}\ E(\mathbf{q}) = \underset{\mathbf{q},\mathbf{\Pi}_1}{\arg\min}\ \sum_n^{N_p}\left(\mathbf{p}_2^n - {}^2\mathbf{H}_1\mathbf{p}_1^n\right)^2 \tag{6}$$

In a dense form the Jacobian of Eq. (6) can then be reformulated as:

$$J(\mathbf{q}, \mathbf{\Pi}) = \left[\begin{array}{cc}\frac{\partial E}{\partial \mathbf{q}} & \frac{\partial E}{\partial \mathbf{\Pi}}\end{array}\right] \in \mathbb{R}^{2\times 10} \tag{7}$$

With the Jacobian of camera pose $J(\mathbf{q})$ defined as the Jacobian of $E(\mathbf{q})$ in $\mathbf{q}$:

$$J(\mathbf{q}) = \left[\begin{array}{cccccc}-1/Z & 0 & x/Z & xy & -(1+x^2) & y \\ 0 & -1/Z & y/Z & 1+y^2 & -xy & -x\end{array}\right] \tag{8}$$

where (x, y) are 2D points coordinates corresponded to $\mathbf{p}$, $1/Z$ is the inverse depth and computed as follows with $\mathbf{p_2}$ keypoint in frame 2 (see [30]):

$$1/Z = \frac{d - {}^2\mathbf{t}_1\mathbf{n}_1}{{}^2\mathbf{R}_1\mathbf{n}_1\mathbf{p}_2} \tag{9}$$

Similarly for Jacobian of plane $\frac{\partial E}{\partial \mathbf{\Pi}}$, four columns representing $\frac{\partial E}{\partial n_x}$, $\frac{\partial E}{\partial n_y}$, $\frac{\partial E}{\partial n_z}$, $\frac{\partial E}{\partial d}$.

$$J(\mathbf{\Pi}) = \left[\begin{array}{cccc}\frac{x(t_z x - t_x)}{d} & \frac{y(t_z x - t_x)}{d} & \frac{(t_z x - t_x)}{d} & \frac{1/Z(t_z x - t_x)}{d} \\ \frac{x(t_z y - t_y)}{d} & \frac{y(t_z y - t_y)}{d} & \frac{(t_z y - t_y)}{d} & \frac{1/Z(t_z y - t_y)}{d}\end{array}\right] \tag{10}$$

t_x is the x axis value in $\mathbf{t} = (t_x, t_y, t_z)^\top$.

However, for the case of multiple homographies in a static environment, the relation of a set of homographies detected in the images $\{{}^2\mathbf{H}_1^i\}$ consists of a shared transformation ${}^2\mathbf{R}_1, {}^2\mathbf{t}_1$, where $i = 1..N_{\mathbf{\Pi}}$ as the number of plane:

$${}^2\mathbf{H}_1^i = {}^2\mathbf{R}_1 + \frac{{}^2\mathbf{t}_1}{d_i}{\mathbf{n}_1}^{i\top} \tag{11}$$

By exploiting this characteristic, we propose a camera pose and plane refiner for multiple homographies:

$$\{\widehat{\mathbf{q}}, \{\widehat{\mathbf{\Pi}_1^i}\}\} = \underset{\mathbf{q},\mathbf{\Pi}_1^i}{\arg\min}\ \sum_i^{N_{\mathbf{\Pi}}}\sum_n^{N_p}\left(\mathbf{p}_2^n - {}^2\mathbf{H}_1^i\mathbf{p}_1^n\right)^2 \tag{12}$$

The Jacobian actually holds a sparse form, for example the block of Jacobian for computing all keypoints in plane $i \in 1 \dots N_{\mathbf{\Pi}}$ can be then defined as:

$$J(\mathbf{q}, \mathbf{\Pi}^i) = \left[\begin{array}{ccccc}\frac{\partial E}{\partial \mathbf{q}} & \underbrace{0 \cdots 0}_{4(i-1)} & \frac{\partial e}{\partial \mathbf{\Pi}^i} & \underbrace{0 \cdots 0}_{4(N_{\mathbf{\Pi}}-i)}\end{array}\right] \in \mathbb{R}^{2\times(6+4N_{\mathbf{\Pi}})} \tag{13}$$

Therefore the Jacobian of single image refiner for all planes is:

$$J(\mathbf{q}, \mathbf{\Pi}) = \left[\begin{array}{cccc}J(\mathbf{q},\mathbf{\Pi}^0)^\top & J(\mathbf{q},\mathbf{\Pi}^1)^\top & \dots & J(\mathbf{q},\mathbf{\Pi}^{N_{\mathbf{\Pi}}})^\top\end{array}\right]^\top \tag{14}$$

Refer to Section VIII for the visualisation of estimation between image pairs.

B. Bundle Adjustment-like Refiner

1) Plane Association: Unlike keypoint-based Bundle Adjustment (BA) techniques widely used in [1][31], our 3D planar map is designed as a two-level structure: extracted keypoints belong to different planes respectively. Therefore a plane association process is mandatory for the following BA section. The problematic can be reformulated as follows: we search for a way of matching two sets of planes from two frames respectively $\{\mathbf{\Pi}_c\}$ and $\{\mathbf{\Pi}_{c+1}\}$.

In contrast with related work which directly compare these plane parameters $\{\mathbf{n}, d\}$ without considering image information [14], or others which only consider image overlapping information but do not account for geometric constraints, we propose a hybrid plane association policy considering both geometric and on-image information:

i) As the distance d is heavily influenced by scale ambiguity we first compare the angle between two normal vectors $d(\mathbf{n}_c, \mathbf{n}_{c+1})$. However this method does not differentiate two parallel planes in the environment.

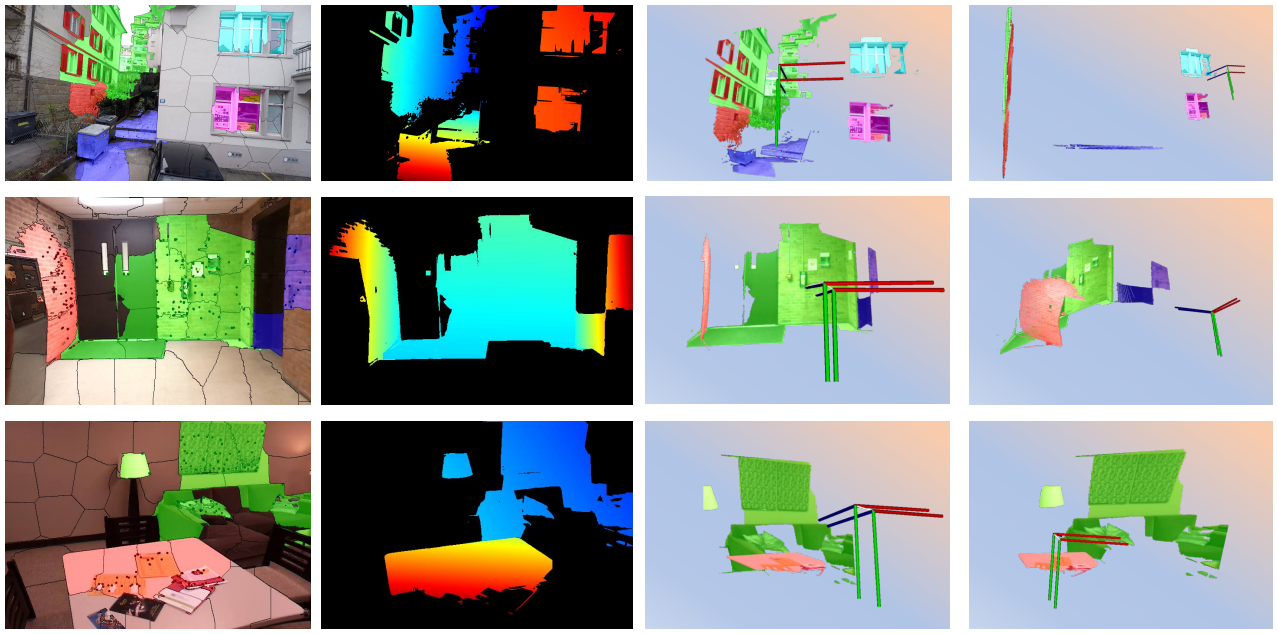

Fig. 3: Demonstration of results estimated from image pairs. Depth image and 3D planar maps are also illustrated showing that our method estimates well under the multi-planar environment. Result (c) shows that our method conserves well the orthogonality among planes without relying on the Manhattan assumption.

ii) Superpixel tracking results are also taken into consideration. It not only helps avoid the parallel planes from mismatching but can also reject the camera pose when the translation is too small between images and all planes become one homography.

iii) We finally check the number of matched descriptors among planes. A window search after re-projecting by homography can also be applied for a more robust matching result: *e.g.,* for comparing the keypoints between frame $\mathbf{p}_{c+n}$ and frame $\mathbf{p}_c$, as no direct ${}^{c+n}\mathbf{H}_c$ computed from image, we can simply propagate the keypoints in frame i by multiplying the homography matrices: ${}^{c+n}\mathbf{H}_{c+n-1} \dots {}^{c+1}\mathbf{H}_c\mathbf{p}_c$ and compare them with $\mathbf{p}_{c+n}$ in a window searching method.

2) Plane Map Refiner: The Plane map refiner consists in an optimization framework which refines all keyframes' poses and their common planes found by plane matching process. Each keyframe contains multiple planes and keypoints in each plane. Once the *joint plane* information is gained over different keyframes, like global BA for point-based SLAMs, this procedure eliminates the drifting problem, solves scale ambiguity and refines camera trajectory w.r.t whole sequence. The BA-like optimization approach we propose accounts for all homographies from all different keyframes:

$$\arg\min_{\mathbf{q}_c,\mathbf{\Pi}_c^i} \sum_{c}^{N_c} \sum_{i}^{N_\Pi} \sum_{n}^{N_p} \left(\mathbf{p}_{c+1}^n - {}^{c+1}\mathbf{H}_c^i \mathbf{p}_c^n\right)^2 \qquad (15)$$

where c and i are the index of frame and plane number, N_c and N_Π represent the total frame and plane number respectively.

3) Keyframe Selection: Our proposed keyframe selection is a straightforward heuristic comparable to systems like [1], [2]. We rely on the parallax metric (defined as an average translation of all matched keypoints between images) and matching quality for choosing keyframes. Two conditions are checked i) to have a parallax on at least a given number of pixels; this is a hyper-parameter from one dataset to another, empirically found between 20 to 40 pixels, and ii) at least a certain number of planes is well matched. This parameter is also adjustable as some environments include many small planes and some comprise less.

VIII. Experiments

Our experiments include three parts: image pairs, indoor experiment and outdoor experiment.

We test various image pairs under different environment and camera types across a wide range of datasets including RGB image of Kinect camera [32], hand-held mobile phone [33] and Micro Air Vehicle images [34]. Results are presented in Fig 3 with the plane estimation, correspondent depth image as well as a 3D planar map with camera pose. Another example of comparison is given in Fig 4, the estimated depth image corresponds well to the ground truth estimated by Kinect camera and is able to keep a very dense form which seems difficult for sparse and even semi-dense RGB monocular mapping systems.

To test indoor environment on whole image sequences, we relied on the TUM RGB-D dataset[32] also used in [21], [35]. The scene is constructed as a pure planar environment, however the homogeneous color distribution on the pop-up shape wall is relatively challenging for superpixel extraction: many superpixels are spawned at the frontier of two planes as their color seems very similar. See Table I for the

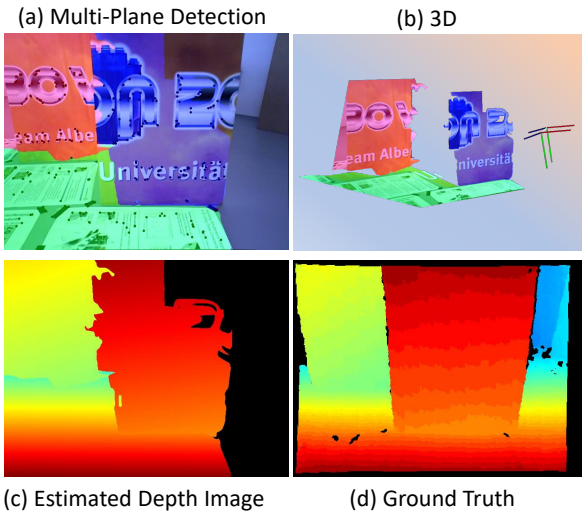

Fig. 4: Comparison of estimated results from *image pair* against the depth map from ground truth on the dataset TUM [32]. With a small number of parameters (3 planes), our proposed method is able to generate a very dense map.

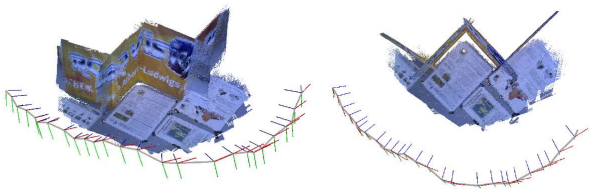

Fig. 5: 3D multiple plane map and camera trajectory of the dataset TUM [32] generated by our method.

generated results by comparing with ORB-SLAM [1], LSD-SLAM [2], Multi-Level Mapping [35] and DPPTAM [21]. Our method outperforms all dense and semi-dense methods in terms of absolute pose error (ATE) and reaches a good level of precision against a state-of-the-art monocular sparse keypoint-based SLAM [1] which only provides sparse point cloud mapping. A possible reason of lower performance against [1] could be our primitive keyframe selection policy (VII-B.3), as a significantly improved result is demonstrated while the keyframes are well-selected manually.

	ATE (m)		
Methods	***Mean***	***Median***	***RMSE***
ORB-SLAM	**0.010**	**0.009**	**0.012**
LSD-SLAM	0.157	0.124	0.170
Multi-Level Mapping	-	-	0.17
DPPTAM	0.063	0.063	0.065
Well Selected KF (ours)	**0.023**	**0.017**	**0.027**
Mean of 5 consecutive runs (ours)	0.037	0.031	0.045
Median of 5 consecutive runs (ours)	0.040	0.029	0.047

TABLE I: Evaluation of ATE of dataset TUM RGB-D [32]. The proposed method outperforms DPPTAM, LSD-SLAM and Multi-Level Mapping. Despite behind ORB-SLAM performance (a keypoint-based sparse SLAM technique without planar assumption), our approach provides a dense map representation.

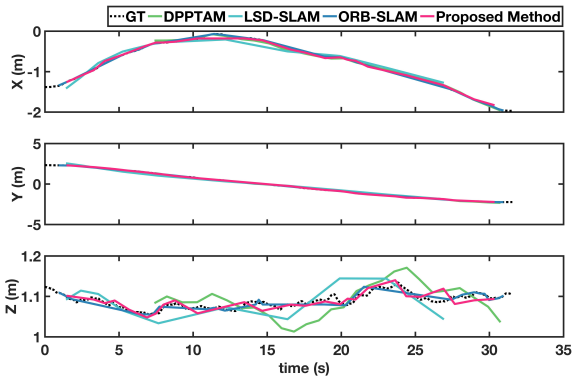

Fig. 6: Comparison of trajectories generated from different methods: Our proposed method shows a more stable and similar trajectory results w.r.t LSD-SLAM and DPPTAM, reaches the save level of state-of-the-art sparse SLAM method ORB-SLAM, thanks to the global planar representation and non-linear BA.

Finally we test our system on image sequence from hand-held monocular gray-level camera dataset [36], under an outdoor and corridor-like environment. Fig 7 displays that our system successfully recovers the multiple planes structure as well as a camera trajectory from the sequence.

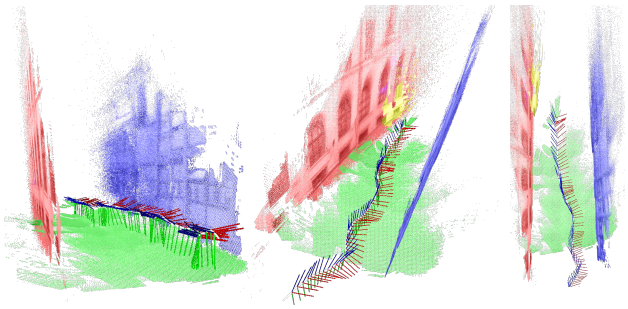

Fig. 7: Experiment on an outdoor dataset [36], coordinates represent the camera pose of keyframes. The multi-planar structure is well conserved without applying any assumptions under a corridor-like environment.

IX. Conclusion

We proposed a novel method to estimate a camera pose from sparse keypoints and simultaneously reconstruct a dense planar map representation via multiple homographies. A superpixel-driven RANSAC method was introduced to perform multiple homography extractions from planes, and homography ambiguities were resolved using a voting system. We also introduced an optimization camera and plane map refiner to perform more precise mapping and tracking results. Results demonstrate the benefits of the approach in comparison with existing contributions.

Future work will focus on improving plane matching techniques and life-long performance, to match the precision of sparse SLAM techniques and yielding more lightweight map representations than dense SLAM techniques.

REFERENCES

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "Orb-slam: A versatile and accurate monocular slam system," *IEEE Trans. on Robotics*, vol. 31, no. 5, pp. 1147–1163, Oct 2015.

[2] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European conference on computer vision*. Springer, 2014, pp. 834–849.

[3] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Mar. 2018.

[4] N. Krombach, D. Droeschel, and S. Behnke, "Combining feature-based and direct methods for semi-dense real-time stereo visual odometry," in *International conference on intelligent autonomous systems*. Springer, 2016, pp. 855–868.

[5] C. Pirchheim and G. Reitmayr, "Homography-based planar mapping and tracking for mobile phones," in *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, 2011, pp. 27–36.

[6] A. Dame and E. Marchand, "Second-order optimization of mutual information for real-time image registration," *IEEE Transactions on Image Processing*, vol. 21, no. 9, pp. 4190–4203, 2012.

[7] S. Benhimane and E. Malis, "Homography-based 2d visual tracking and servoing," *The International Journal of Robotics Research*, vol. 26, no. 7, pp. 661–676, 2007.

[8] G. Silveira, E. Malis, and P. Rives, "An efficient direct approach to visual slam," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 969–979, Oct 2008.

[9] E. Malis and M. Vargas, "Deeper understanding of the homography decomposition for vision-based control," 2007.

[10] B. Guan, P. Vasseur, C. Demonceaux, and F. Fraundorfer, "Visual odometry using a homography formulation with decoupled rotation and translation estimation using minimal solutions," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 2320–2327.

[11] O. Saurer, F. Fraundorfer, and M. Pollefeys, "Homography based visual odometry with known vertical direction and weak manhattan world assumption," in *ViCoMoR 2012: 2nd Workshop on Visual Control of Mobile Robots (ViCoMoR): Half Day Workshop: October 11th, 2012, Vilamoura, Algarve, Portugal, in conjunction with the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012)*, 2012, pp. 25–30.

[12] A. Flint, D. Murray, and I. Reid, "Manhattan scene understanding using monocular, stereo, and 3d features," in *2011 International Conference on Computer Vision*, 2011, pp. 2228–2235.

[13] S. Yang, Y. Song, M. Kaess, and S. Scherer, "Pop-up slam: Semantic monocular plane slam for low-texture environments," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 1222–1229.

[14] M. Kaess, "Simultaneous localization and mapping with infinite planes," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 4605–4611.

[15] M. Hsiao, E. Westman, G. Zhang, and M. Kaess, "Keyframe-based dense planar slam," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 5110–5117.

[16] P.-H. Le and J. Košecka, "Dense piecewise planar rgb-d slam for indoor environments," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 4944–4949.

[17] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[18] M. Van den Bergh, X. Boix, G. Roig, B. de Capitani, and L. Van Gool, "Seeds: Superpixels extracted via energy-driven sampling," in *European conference on computer vision*. Springer, 2012, pp. 13–26.

[19] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *International journal of computer vision*, vol. 59, no. 2, pp. 167–181, 2004.

[20] A. Concha and J. Civera, "Using superpixels in monocular slam," in *2014 IEEE international conference on robotics and automation (ICRA)*, 2014, pp. 365–372.

[21] ——, "Dense Piecewise Planar Tracking and Mapping from a Monocular Sequence," in *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[22] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*, 2011, pp. 2564–2571.

[23] S. Wang, H. Lu, F. Yang, and M.-H. Yang, "Superpixel tracking," in *2011 International Conference on Computer Vision*, 2011, pp. 1323–1330.

[24] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, pp. 381–395, 1981.

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

[26] C. V. Stewart, "Bias in robust estimation caused by discontinuities and multiple structures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 8, pp. 818–833, Aug 1997.

[27] Y. Kanazawa and H. Kawakami, "Detection of planar regions with uncalibrated stereo using distribution of feature points," in *In British Machine Vision Conference*, 2004, pp. 247–256.

[28] E. Vincent and R. Laganiere, in *ISPA 2001. Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis.*, June 2001, pp. 182–187.

[29] M. Zuliani, C. S. Kenney, and B. Manjunath, "The multiransac algorithm and its application to detect planar homographies," in *IEEE International Conference on Image Processing 2005*, vol. 3, 2005, pp. III–153.

[30] E. Marchand, H. Uchiyama, and F. Spindler, "Pose estimation for augmented reality: a hands-on survey," *IEEE transactions on visualization and computer graphics*, vol. 22, no. 12, pp. 2633–2651, 2015.

[31] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *2007 6th IEEE and ACM international symposium on mixed and augmented reality*, 2007, pp. 225–234.

[32] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[33] A. Furlan, D. Miller, D. G. Sorrenti, L. Fei-Fei, and S. Savarese, "Free your camera: 3d indoor scene understanding from arbitrary camera motion," in *BMVC*, 2013.

[34] A. L. Majdik, C. Till, and D. Scaramuzza, "The zurich urban micro aerial vehicle dataset," *The International Journal of Robotics Research*, vol. 36, no. 3, pp. 269–273, 2017.

[35] W. N. Greene, K. Ok, P. Lommel, and N. Roy, "Multi-level mapping: Real-time dense monocular slam," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 833–840.

[36] J. Engel, V. Usenko, and D. Cremers, "A photometrically calibrated benchmark for monocular visual odometry," in *arXiv:1607.02555*, July 2016.