

# Topology-Guided Path Planning for Reliable Visual Navigation of MAVs

Dabin Kim\*, Gyeong Chan Kim\*, Youngseok Jang, and H. Jin Kim

**Abstract**—Visual navigation has been widely used for state estimation of micro aerial vehicles (MAVs). For stable visual navigation, MAVs should generate perception-aware paths which guarantee enough visible landmarks. Many previous works on perception-aware path planning focused on sampling-based planners. However, they may suffer from sample inefficiency, which leads to computational burden for finding a global optimal path. To address this issue, we suggest a perception-aware path planner which utilizes topological information of environments. Since the topological class of a path and visible landmarks during traveling the path are closely related, the proposed algorithm checks distinctive topological classes to choose the class with abundant visual information. Topological graph is extracted from the generalized Voronoi diagram of the environment and initial paths with different topological classes are found. To evaluate the perception quality of the classes, we divide the initial path into discrete segments where the points in each segment share similar visual information. The optimal class with high perception quality is selected, and a graph-based planner is utilized to generate path within the class. With simulations and real-world experiments, we confirmed that the proposed method could guarantee accurate visual navigation compared with the perception-agnostic method while showing improved computational efficiency than the sampling-based perception-aware planner.

## I. INTRODUCTION

Fully automated robotic operation requires a perception module that recognizes surrounding environment and estimates the robot state. In particular, visual odometry (VO) and simultaneous localization and mapping (SLAM) using vision sensors have been conducted for self-localization of micro aerial vehicles (MAVs) due to low weight, cost, and small size of the sensors while capturing abundant visual information of surrounding environments. The integration of motion planning and perception modules poses significant issues for consideration, one of which is that the robot's localization capabilities are dependent on the path chosen by the robot. Therefore, it is necessary to take the perception module into account at the motion planning level, and this approach is called perception-aware motion planning.

In general, the performance of visual navigation is affected by the number and distribution of salient keypoints in the

This work was supported by Institute of Information Communications Technology Planning Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2019-0-00399, Development of A.I. based recognition, judgement and control solution for autonomous vehicle corresponding to atypical driving environment)

Dabin Kim, and Youngseok Jang are with the Mechanical and Aerospace Engineering Department, Seoul National University, Seoul 08826, South Korea (e-mail: dabin404@snu.ac.kr, duscjs59@gmail.com).

Gyeong Chan Kim and H. Jin Kim are with the Aerospace Engineering Department, Seoul National University, Seoul 08826, South Korea (e-mail: skykim0609@snu.ac.kr, hjinkim@snu.ac.kr, corresponding author: H. Jin Kim).

\* These authors contributed equally to this manuscript.



Fig. 1. Snapshot from the experiment. For robust visual navigation, a micro aerial vehicle should decide a path which can guarantee enough visible features from multiple path candidates.

observed images. For example, if the generated trajectory passes through a texture-less area, it may become difficult to execute given missions due to accumulated error of state estimation. Therefore, instead of conventional planners that are agnostic to the performance of the perception module, perception-aware motion planning is needed to lead stable visual navigation of MAVs.

To this end, the previous perception-aware planners have focused on approximation of navigation performance by inserting a perception-related cost on existing motion planning algorithms. Especially, sampling-based methods [1], [2], [3] have been mainly used for perception-aware planners. However, since the evaluation of perception quality itself is usually computationally heavy [4], difficulty arises when the planner suffers from sample inefficiency in large environments.

Instead of covering the entire environment with randomly drawn samples, exploiting higher-level information about the environment improves efficiency of global planning. The topology of the environment is determined from the geometrical distribution of obstacles. This fact leads to an important heuristic in the perspective of visual navigation. The visibility of the landmarks is limited by the relative pose between MAV and obstacles, thus the topological class of path restricts the maximum obtainable visual information [5], [6]. Therefore, taking topological information of paths into account helps to find whether the path is advantageous for visual navigation. In addition, once a reference path is generated, it cannot be updated to belong to a different topology through gradient-based optimization [7]. As a result, finding topological classes of paths with enough visual features gives important prior information for perception-aware path planning.

In this paper, a topological perception-aware planner is suggested to prevent situations difficult to obtain accurate state estimation due to limited visual information. To create a path which attains both short path length and good perception quality, we propose the process of generating paths belonging to distinct topology classes and evaluate each path's quality with respect to path length and visual information. According to the authors' knowledge, this work is the first research on perception-aware motion planning that incorporates visual information with topological planning. The proposed planner can be exploited as an reference path to generate a feasible trajectory via trajectory optimization, or to provide prior information for a lower-level global planner.

## II. RELATED WORK

### A. Perception-aware Planning

Perception-aware motion planning refers to the algorithms that generate motion by considering the localization quality of the onboard navigation system. This study focuses specifically on motion planning algorithms that are applicable to visual navigation systems. In order to find a path from the start to the goal point, most perception-aware global planning algorithms are based on sampling-based methods. [8] suggested the Rapidly-exploring random belief tree (RRBT) for planning in belief space with a linear estimator based on a sampling-based method. As an extension of RRBT for vision systems, [1] applied local bundle adjustment (BA) in offline to estimate the covariance of future pose, and [2] designed perception score based on the feature numbers in images as an approximation of localization quality. [3] utilized the photometric information of images for dense VO and biased the path toward texture-rich region. Unlike aforementioned methods, our work suggests a global planning method which can utilize the environment's topological information as a heuristic for perception quality.

Another focus of perception-aware motion planning is improving perception quality of the trajectory via considering tracking and triangulation of local landmarks. [9] formulated an optimal control problem which simultaneously minimizes the energy and velocity of the point of interest in image plane. [10] suggested differentiable cost for keeping visible features inside the Field of View (FoV) of the camera. [11] applied feature triangulation and covisibility-related costs to gradient-based trajectory optimization, encouraging the tracking of local landmarks. [12] and [13] used receding-horizon method with designed planning cost regarding perception quality, which is evaluated from local landmarks.

### B. Topological Planning

Among other global planning methods such as sampling-based methods and search-based methods, topological planning methods obtain paths from the topological graph of the environment. Topological planning is widely used in motion planning in the sense of reducing planning dimension with topological constraints [6], storing and searching for pre-visited regions [14], and finding the global optimal path from distinctive topologies [15]. Our work also evaluates paths

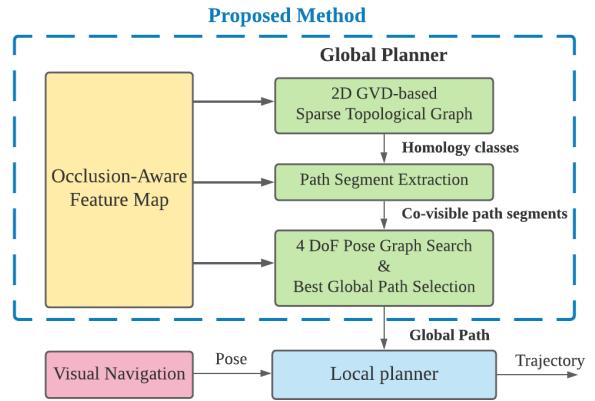


Fig. 2. Overview of the proposed method.

from distinctive topologies, though we focus on path planning with consideration about the path's perception quality.

To create a traversable topological graph structure from a given environment, PRM-based methods and generalized Voronoi diagram (GVD) can be used. However, as pointed out in [15], GVD is more beneficial than PRM-based methods in global planning since GVD guarantees coverage of the environment. GVD is a form of a roadmap structure, which can be computed online via the Euclidean signed distance field (ESDF) with low computational cost [16]. Although topology in 3D spaces can be computed to better describe MAV flight as in [17], in this study, we use a 2D topological graph for the sake of efficient computation under the assumption that the flight altitude would not drastically change.

## III. SYSTEM DESCRIPTION

In this paper, we suggest a planner that allows MAV to move from the start to the goal point while maintaining good self-localization. The MAV can observe surrounding environments and estimate ego-motion using a vision sensor. Although proposed method can also be adapted to multi-camera systems, but single camera is used for demonstration.

The overall structure for the perception-aware motion planning is described in Fig. 2. To represent the environment, we used an occlusion-aware feature map, which is an integrated map of volumetric and landmark maps. This allows to obtain information about visibility of map points and occupancy of the environment, which is required for evaluating the perception quality and generating a collision-free path. The global planner first constructs a sparse topological graph via GVD. Then initial paths belonging to distinct topologies are extracted from the graph. For each initial path, it is divided into multiple segments based on visibility information. Pose samples are generated near segments, and perception quality of each sample is evaluated. Then the path which can obtain the maximum perception information is generated via graph search. Among the generated paths from distinctive topologies, the best path with respect to the path length and perception quality is selected as an output of the global planner.

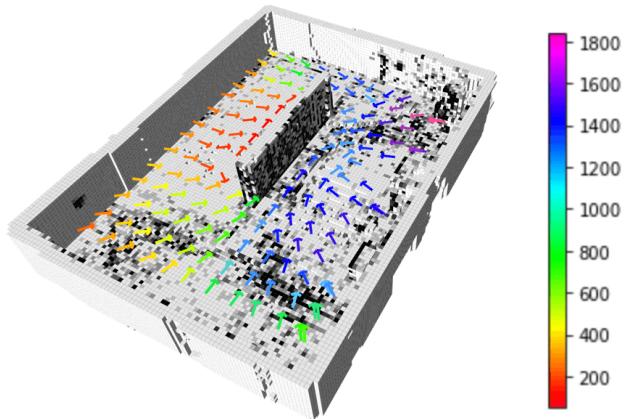


Fig. 3. Occlusion-aware feature map representation of the environment drawn in Fig. 4. The darker the voxel color is, the more landmark information it contains. The arrows in the figure indicate the orientation with maximum visible landmarks from sampled position, and the color of each arrow represents the number of visible landmarks from the respective pose.

#### IV. MAP REPRESENTATION

In perception-aware planning, it is required to compute visibility information of landmarks from arbitrary poses to evaluate perception quality at candidate waypoints. In this paper, we consider visual SLAM algorithms which represent the map using point features observed in keyframes [18], [19]. While sparse pointcloud map allows efficient query of visible landmark candidates using adjacent keyframes, it does not provide the capability to consider geometry of the scene and exclude occluded landmarks. This may limit the accuracy of the queried visibility information, especially in an obstacle-filled environment where occlusion of landmarks occurs frequently. To tackle this issue, we have considered 3D volumetric map representation which allows more accurate reasoning on 3D geometry of the scene.

3D volumetric map representations have been developed to enable robots to differentiate between the traversable and occupied spaces. By modeling 3D space with probabilistic occupancy grid [20] or ESDF [21], these map representations inherently provide the ability to reason about scene geometry. We devise a method to integrate SLAM map with volumetric map representation to create occlusion-aware feature map representation. Our method is similar to the method proposed in [3], where the authors stored texture information for each occupied voxel's surface. In our case, instead of storing photometric information, we embed each landmark's information in the occupied voxel corresponding to its location. Example of this map representation is illustrated in Fig. 3.

To construct the integrated map of an environment, we first construct a volumetric map and a SLAM map separately from series of measurements by RGB-D camera. Then the respective global coordinate frames of the maps are aligned and each landmark's information is embedded into the voxel at its location. Finally, to allow raycasting-based visibility query, landmark information is shifted to the nearest surface voxel which is visible from reference keyframe's pose. In

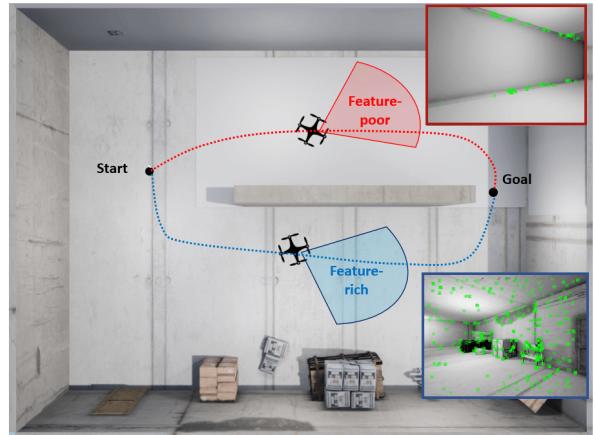


Fig. 4. Illustration of dependence of perception quality on homology classes. Tracking a trajectory belonging to homology class marked in red leads to bad localization because feature-rich surfaces are occluded by the wall in the middle. Therefore, homology class of blue path is preferred in terms of perception quality.

this work, we used ORB-SLAM2 [18] and Voxblox [21] for SLAM map and volumetric map representation, respectively.

#### V. TOPOLOGICAL GLOBAL PLANNING

Based on the integrated map, we generate a global path which serves as an initial reference for a low-level planner. From the observation that visual information and relative position between MAV and obstacles are strongly related, we devise a method to generate multiple global paths with distinctive topologies and to select the path with respect to the perception quality and path length. This section is configured as follows. In Sec. V-A, we will give a detailed explanation on why it would be beneficial for a perception-aware planner to consider topological properties of the path. The process of perception-aware planning will be covered in Sec. V-B ~ V-E. The overall process of global planner is illustrated in Fig. 5 and Alg. 1.

##### A. Homology Classes and Perception Quality

In this subsection, the close relationship between topological planning and perception quality is explained. To express topological equivalence of trajectories, the concept of homology class is widely used. Two trajectories with fixed start and goal points in 2D belong to the same homology class if the cycle formed by them does not include or intersect any obstacle. For a more formal definition of homology class, refer to [22].

During trajectory generation for MAV flight, the local planner refines the reference path from global planner to smooth and feasible trajectory. In order to maintain visual navigation stable, the global planner should generate a path such that navigation does not fail, not only in a reference path but also in a path refined by the low-level planner. However, the local planner module updates the path with collision avoidance constraint. For example, some algorithms restrict the search space into free, convex region as a hard constraint [23], [24] and other algorithms which use gradient-based optimization update paths to the opposite direction

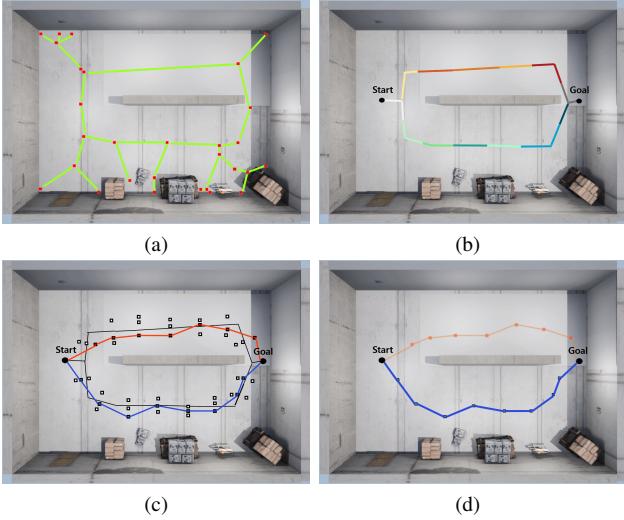


Fig. 5. Process of the proposed planner (a) Topological graph generation from GVD (b) For each homology class, path segments are extracted based on feature co-visibility. (c) Pose samples are generated and optimal paths are found. (d) The best path is selected with respect to the perception quality and path length.

of obstacles [25], [26]. Therefore, it is difficult for local planners to update a path to jump over obstacles and change its homology class, thus reference and refined paths are topologically equivalent.

On the other hand, a path's homology class affects visual information which MAV can obtain by following the path. Selecting the homology class fixes relative topology of the path to the obstacles, which determines visible surfaces of obstacles. In the vision-based navigation system, features are detected on the surface of obstacles. If feature-rich surfaces are hindered by occlusion, we can conclude that corresponding homology class is disadvantageous for visual navigation. Thus, the homology class which can guarantee visibility of feature-rich surfaces is preferred. The relationship between homology class and perception quality of the path can be observed in Fig. 4.

Therefore, searching distinctive homology classes can be a helpful heuristic for finding a reference path for reliable visual navigation. In addition, it also enables to boost computation by searching each homology class in parallel, exploiting multi-process CPU.

### B. Topological Graph Generation

The methodology that we present aims to find a global path through the topological structure of the environment. To this end, it is necessary to generate topological graphs in a given environment and extract homology classes. To generate a topological graph, we create a 2D GVD with the given reference height. Construction of GVD follows the method suggested by [16], in which the ESDF map is used to obtain a 2D voxel map of GVD. Since the graph structure of GVD is needed, vertices and edges are extracted from the GVD. A voxel is a vertex if 1 or more than 2 neighboring voxels are elements of GVD. And a connected set of voxels is an edge if it connects two vertices. By classifying each voxel

---

### Algorithm 1 Topological perception-aware path planner

---

```

1: Input: Map  $M$ , Start  $s$ , Goal  $g$ 
2: Output: Global Path  $P^*$ 
3: Topological Graph Generation  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ 
4: Generate Initial Paths from Distinct Homology Classes
    $H = \{h_1, \dots, h_T\}$ 
5: for  $h_t \in \{h_1 \dots h_T\}$  do
6:   Extract Path Segments
    $h_t \rightarrow s(s, p_1), s(p_1, p_2), \dots, s(p_m, g)$ 
7:   Generate 4DoF pose samples  $(n_k^j)$  & evaluate perception quality  $I(n_k^j)$ 
8:   Pose graph search  $P_t^* = (n_{k_1}^1, n_{k_2}^2, \dots, n_{k_N}^N)$ 
9:   Evaluate quality of the path  $q(P_t^*)$  (7)
10:  end for
11:  $P^* = \operatorname{argmax}_{q(P_t^*)} \{P_1^*, \dots, P_T^*\}$ 

```

---

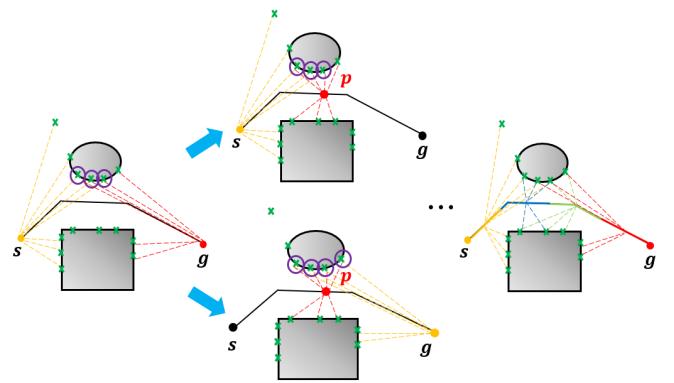


Fig. 6. Illustration of path segment extraction process. Green cross represents landmarks, and the co-visible candidate set is marked in purple circles. A path is iteratively divided until there are significant portion of co-visible landmarks at both ends of every segment.

as a vertex or an edge, a bi-directional topological graph is acquired. By adding the start and goal points to the graph, connected paths from the start to the goal can be obtained from this graph. By adopting the methodology from [15], we can find ‘initial paths’, which are sets of consecutive vertices, from distinct homology classes using breadth first search on the topological graph.

### C. Extracting Path Segments

After initial paths are generated in different homology classes, each path and corresponding homology class are evaluated with respect to perception quality. It is important to maintain co-visibility to the landmarks along the path for vision-based localization. Thus, we evaluate the perception quality of the path based on co-visible landmarks while traveling along the path. However, especially in environments with multiple obstacles, visible landmarks change as the robot travels through the path, which makes it difficult to find a set of globally co-visible landmarks along the whole path.

Thus, we seek to split the initial path into smaller ‘path segments’ where points in the same segment share a significant amount of visible landmarks. This is inspired by the

previous study on co-visibility based regional segmentation [27], where different positions in the map are grouped based on the similarity of their observations. While the method in [27] was devised to obtain a topological map of the environment, our objective is to divide a path based on the similarity of predicted observation along the path. The algorithm is depicted in Fig. 6.

We defined the ‘visibility candidate set’ at point  $\mathbf{p} \in \mathbb{R}^3$ ,  $V(\mathbf{p})$ , as the set of the landmarks which are within the vertical FoV and sensing range. Given two points  $\mathbf{p}, \mathbf{q} \in \mathbb{R}^3$ , the ‘co-visible candidate set’  $CV(\mathbf{p}, \mathbf{q})$  and ‘co-visibility ratio’  $CR(\mathbf{p}, \mathbf{q})$  are respectively defined as

$$CV(\mathbf{p}, \mathbf{q}) = V(\mathbf{p}) \cap V(\mathbf{q}) \quad (1)$$

$$CR(\mathbf{p}, \mathbf{q}) = \frac{|CV(\mathbf{p}, \mathbf{q})|}{|V(\mathbf{p}) \cup V(\mathbf{q})|} \quad (2)$$

As in Fig. 6, we begin with evaluating co-visibility ratio of the start and goal point:  $CR(\mathbf{s}, \mathbf{g})$ . If it is larger than the threshold  $\eta \in [0, 1]$ , we consider the path as a path segment  $s(\mathbf{s}, \mathbf{g})$  and assign  $CV(\mathbf{s}, \mathbf{g})$  as the co-visible candidate set for the path segment. Otherwise, we query the visibility candidate set at the midpoint  $\mathbf{p}$  of the path and evaluate  $CR(\mathbf{s}, \mathbf{p})$  and  $CR(\mathbf{p}, \mathbf{g})$ . We iteratively divide the path until the co-visibility ratio of the end points for each segment becomes larger than  $\eta$ , or length of the segment reaches the minimum length  $\ell_{min}$ . As a result, an initial path is transformed into multiple path segments  $s(\mathbf{s}, \mathbf{p}_1), s(\mathbf{p}_1, \mathbf{p}_2), \dots, s(\mathbf{p}_m, \mathbf{g})$ , where the points in the same segment share significant co-visible landmarks. Furthermore, the previously stored co-visible candidate set of the path segment is used during the evaluation of perception quality to avoid redundant querying of visible landmarks, which include time-consuming raycasting.

#### D. Pose Graph Construction & Graph Search

While a path over sparse graph can represent the homology class, directly using it as global path precludes the existence of a path with better perception quality within the homology class. Also, since GVD is extracted from the 2D plane at the reference height, 3D obstacles cannot be considered in the sparse graph. From this need, we construct a dense graph from given sequence of path segments as in Fig. 5 (c). Each node of the dense graph represents a 4 Degree of Freedom (DoF) pose including the position and yaw angle of the MAV ( $x, y, z$ , and  $\theta$ ), and graph search is performed to find the optimal 4 DoF path.

The 4 DoF pose graph is constructed as follows. First we divide the initial path into intervals of length  $\ell$  which is determined by nominal speed  $v_{nom}$  and time step  $T_s$ . Each of these intervals represents a *layer* of the graph; only the nodes within consecutive layers are allowed to form edges. We denote layers as  $L^1, L^2, \dots, L^N$  where  $N$  is the total number of the intervals obtained by splitting the entire path. For each layer  $L^j$ , waypoint candidates are randomly sampled from the plane passing through the starting point of the interval and perpendicular to the edge of the interval. Samples are generated within a distance  $R_{sample}$  centered

around the path. On top of this, yaw angles are sampled at equal intervals. The sampled waypoint candidates and yaw angles are combined to form nodes of the layer  $\{n_1^j, \dots, n_m^j\}$ , where  $n_k^j = (x_k^j, y_k^j, z_k^j, \psi_k^j)$ . Nodes in the consecutive layers are connected only if the difference of yaw angles between the nodes is smaller than a certain limit  $\dot{\psi}_{lim}$  to prevent an abrupt yaw change.

To find the path with maximum visual information, each node’s perception quality needs to be evaluated. We use the Fisher information matrix (FIM) as the metric for perception quality, which quantifies the information that can be obtained about the desired state through measurement. By modeling the camera as a bearing sensor as in [28], FIM of measurement on a landmark located at  $\mathbf{l}$  observed from pose  $\mathbf{x}$  can be formulated as

$$FIM(\mathbf{l}; \mathbf{x}) = \frac{1}{\sigma^2} (\mathbf{J}(\mathbf{l}; \mathbf{x}))^T \mathbf{J}(\mathbf{l}; \mathbf{x}) \quad (3)$$

$$\mathbf{J}(\mathbf{l}; \mathbf{x}) = \left( \frac{1}{\|\mathbf{l}^c\|} \mathcal{I}_3 - \frac{1}{\|\mathbf{l}^c\|^3} \mathbf{l}^c (\mathbf{l}^c)^T \right) [\mathcal{I}_3 \quad [\mathbf{l}^w]_x] \quad (4)$$

where  $\mathbf{l}^c, \mathbf{l}^w$  are the position of point seen from camera frame and global frame respectively, and  $\sigma$  is standard deviation of measurement noise. For each node, we compute the visible landmarks from previously stored co-visible candidate set along the path segment containing the layer node is in. For the  $j$ -th node, we evaluate the FIM of the visible landmarks at pose  $n_{k_j}^j$  and denote it as  $I(n_{k_j}^j)$ .

We perform a graph search over the pose graph to find the 4 DoF path with the smallest value of combined distance cost and perception cost. Given a path  $P$  as a sequence of connected nodes  $P = (n_{k_1}^1, n_{k_2}^2, \dots, n_{k_N}^N)$ , we can formulate the graph search problem as

$$P^* = \arg \min_{k_1, k_2, \dots, k_N} \lambda_d c_d(P) - \lambda_p c_p(P), \quad (5)$$

where  $\lambda_d, \lambda_p \in \mathbb{R}$  are weights for distance cost and perception cost, the distance cost  $c_d(P)$  can be defined straightforward as the length of total path. The perception cost of the path  $c_p(P)$  is formulated as

$$c_p(P) = \frac{1}{N} \sum_{i=1}^N \log(\det(I(n_{k_i}^i))). \quad (6)$$

To perform graph search, a layered structure of the graph can be exploited. Since each layer is arranged in time order, the graph is a directed acyclic graph. Also, topological sorting is used for graph search by dynamic programming. It can find an optimal path with less computation than Dijkstra search [29].

#### E. Selection of the best path

We design the cost function to quantify the quality of the generated path  $P^*$  for each homology class as

$$q(P^*) = \eta_d \left( \frac{d}{d_{min}} - 1 \right) - \eta_p f(c_{p,min} - c_{p,thr}), \quad (7)$$

$$c_{p,min} = \min(c_p(s(\mathbf{s}, \mathbf{p}_1)), \dots, c_p(s(\mathbf{p}_m, \mathbf{g}))) \quad (8)$$

where  $f$  is defined as  $f(x) = 1/(1 + \exp(x))$ ,  $d$  is the distance of the path,  $d_{min}$  is the length of the shortest

TABLE I: Parameter List for Simulation

| Types              | Parameter Name               | Value                    |
|--------------------|------------------------------|--------------------------|
| Segment Extraction | Min Segment Length [m]       | $l_{min} = 0.5$          |
|                    | Covisibility Ratio           | $\eta = 0.5$             |
| Pose Graph         | Nominal speed [m/s]          | $v_{nom} = 0.4$          |
|                    | Time Step [s]                | $Ts = 1.0$               |
|                    | The number of Samples        | 10                       |
|                    | Max Radius [m]               | $R_{sample} = 0.4$       |
|                    | Yaw Rate Limit [rad/s]       | $\dot{\psi}_{lim} = 0.3$ |
|                    | Edge Distance Weight         | $\lambda_d = 0.1$        |
| Selection          | Edge Perception Weight       | $\lambda_p = 1.0$        |
|                    | Perception Quality Threshold | $c_{p,thr} = 6.0$        |
|                    | Path Distance Weight         | $\eta_d = 0.2$           |
|                    | Path Perception Weight       | $\eta_p = 1.5$           |

path among the generated path candidates, and  $c_{p,thr}$  is the parameter to indicate the required information for robust visual navigation. As in (8), the perception quality of the path is determined by the minimum  $c_p$  value among the segments of the path. It enables to avoid selecting the path passes through feature-poor region, which would result in high estimation error. The reason for using the sigmoid function is because effect of FIM on the localization performance degrades if there is enough information. Among the selected best path within each homology class, the path with the smallest cost is selected as the global path.

## VI. VALIDATION

To validate the proposed topological global planner, simulations and experiments were performed. The global planner was connected to local trajectory optimization to obtain a smooth and kinodynamically feasible trajectory. We used the gradient-based optimization method proposed in [26] as an example of local planner.

### A. Simulation

We used the Unreal engine with AirSim plugin [30] to perform high-fidelity simulations in photo-realistic environments. For visual navigation, a front-looking RGB-D camera was used and we conducted experiments in two realistic environments, *storage* and *gallery*, as in Fig. 7. Simulations were performed on a desktop with 8 core Intel i7 3.2GHz CPU and 32GB RAM.

Four metrics were used for evaluation of the path quality. The first is the total length of the path and the second is the translational estimation error of VO algorithm, which is for quantifying the perception quality of the path. Third metric is the ratio of successful runs of VO without loss in feature tracking, which is related to the reliability of the path for stable navigation. Lastly, computation time was calculated to check computational efficiency of the algorithms. We used ORB-SLAM2 [18] as an example for VO algorithm.

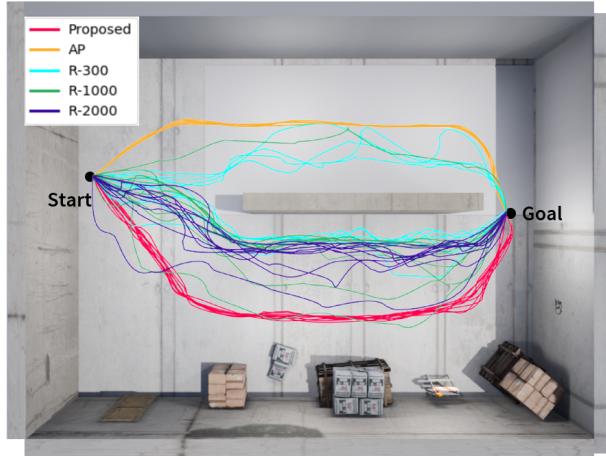
We compared the performance of the proposed method with two other methods. 1) Perception-Agnostic Planner (AP), which is a topological planner but without consideration about perception quality, 2) perception-aware sampling-

based planning based on RRT\* similar to [2], with some modification to fit in our settings. In perception-aware RRT\*, at each sample, visible landmarks are queried, and the perception cost is evaluated using landmarks co-visible from the sample pose and its parent's pose. The criterion for choosing the best path is comparing weighted sum of the distance cost and the perception cost, similar to the proposed planner. Three different numbers of samples were used for the sampling-based planner. Perception-aware RRT\* with  $N$  samples are shortened as ‘R-N’. Parameters used in the simulations are noted in Table I.

We ran each algorithm 10 times for each scenario, and the resulting trajectories are presented on Fig. 7. Also, absolute trajectory error (ATE) for each path and success rates of VO are presented in Fig. 8. For clear visualization, ATE values for only three trials are visualized for each planner.

In the first environment, *storage*, there are two selectable homology classes, distinguished by the wall in the middle. The class at upper side passes through a texture-less region and the lower side has many objects with abundant visual information. As in Fig. 7(a), the proposed algorithm selected the lower side in every trials although path length became longer than choosing the upper side. On the contrary, AP selected the upper side’s homology class and estimation error became higher than the proposed method’s trajectory. For sampling-based planner, we chose 300, 1000, and 2000 samples respectively. Even though the sampling-based planner is designed to prefer feature-rich paths, planner with low sample numbers (300, 1000) often failed to find homology class beneficial for visual navigation and failed to maintain VO during flight. With large samples (2000), it can generate path toward feature-rich region and result in small estimation error, but it takes more computation time compared to the proposed method.

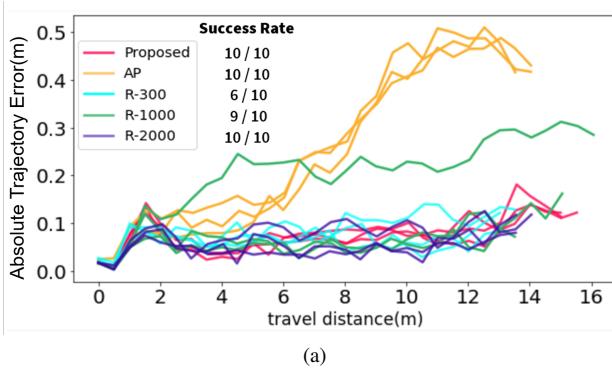
The second environment, *gallery*, contains two major texture-poor regions: a horizontal corridor to the right of the center and a longitudinal corridor to the upper middle. In contrast, many texture-rich objects are distributed along the walls surrounding the environment, especially the walls of left and upper side. Similar to the previous environment, AP chose the homology class with the shortest length among 10 distinct homology classes. However, since the selected path traverses through texture-poor region, it resulted in a higher odometry error and a lower success rate (3/10) compared to the proposed method. For sampling-based planners, since the environment is even larger and more complicated than the previous one, sample efficiency dropped and required more samples to find a visually advantageous path. With 1000 and 8000 samples, the planner often selected path traversing texture-poor regions. Sampling-based planner with abundant sample number (15000) generated paths with lowest estimation error, but it took over 100 s to generate the path at each trial. On the contrary, the proposed algorithm generated paths traversing texture-rich area in all trials, within a much shorter computation time. Results of each simulation environment are summarized in Table II.



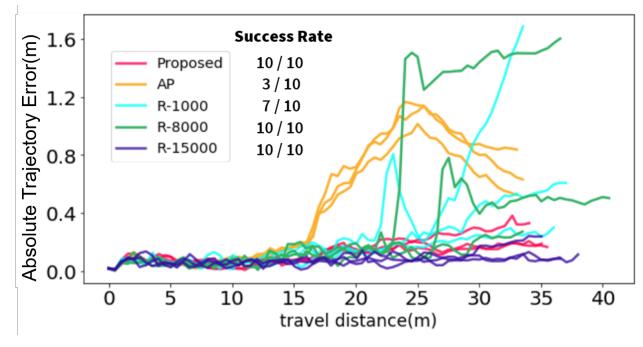
(a)



(b)

Fig. 7. Resulting trajectories from the tested planners in (a) *storage* and (b) *gallery* environments.

(a)



(b)

Fig. 8. Absolute trajectory error of VO with respect to travel distance, and success rate for each planner at (a) *storage* and (b) *gallery* environment. Only 3 successful trials for each planner are shown for clear visualization.

TABLE II: Simulation results for *storage* and *gallery* environments. Mean values of length of the groundtruth trajectory, distance between the goal and the estimated goal position, and computation time are measured. Note that goal estimation error was evaluated only for successful trials.

|                               | Planner  | Length | Goal Error | Comp. Time |
|-------------------------------|----------|--------|------------|------------|
| <i>storage</i><br>(12m × 10m) | Proposed | 15.9 m | 0.118 m    | 1.58 s     |
|                               | AP       | 14.5 m | 0.391 m    | 0.402 s    |
|                               | R-300    | 9.86 m | 0.083 m    | 3.04 s     |
|                               | R-1000   | 13.0 m | 0.136 m    | 9.31 s     |
|                               | R-2000   | 14.3 m | 0.082 m    | 19.5 s     |
| <i>gallery</i><br>(22m × 20m) | Proposed | 35.8 m | 0.235 m    | 7.92 s     |
|                               | AP       | 33.9 m | 0.664 m    | 1.66 s     |
|                               | R-1000   | 36.8 m | 0.619 m    | 9.52 s     |
|                               | R-8000   | 36.4 m | 0.598 m    | 90.2 s     |
|                               | R-15000  | 36.7 m | 0.163 m    | 164.1 s    |

### B. Real-world Experiment

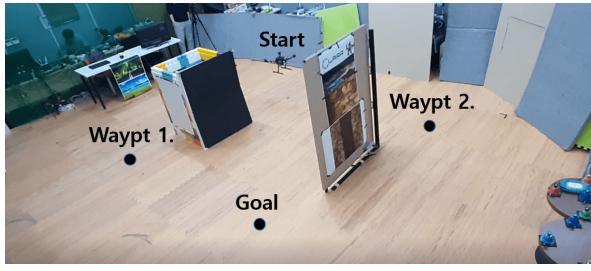
For experiments, S-500 frame quadrotor equipped with forward-facing Realsense D435 camera was used. We used Pixhawk4 flight controller and Intel i7 NUC computer.

As in Fig. 9(a), the environment has two obstacles parallel to each other, with only a small number of features visible

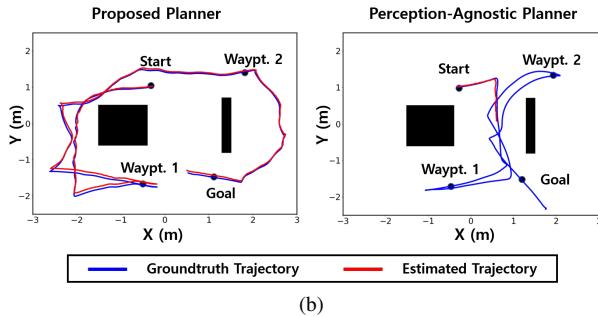
in the middle region and feature-rich objects visible from the rear sides. Two algorithms, the proposed algorithm and the perception-agnostic planner were mounted on MAV. We used OptiTrack motion capture system to provide state information to the controller in order to prevent control failure. It also provided the groundtruth poses. Estimated trajectories from visual odometry were compared with the groundtruth trajectories. As Fig. 9(b). shows, the proposed planner created a path with a low odometry error through the region, but the perception-agnostic planner showed failure on VO since it did not take the visual information of the environment into account.

### VII. CONCLUSIONS & FUTURE WORKS

Based on the observation that the homology class affects the visual information which MAVs can obtain, we proposed a topological path planner for perception-aware navigation. By creating a sparse topological graph from 2D GVD, multiple paths within distinctive homology classes are obtained. Then, the best path within each homology class is searched using graph search. Finally, among the paths selected within each homology, the path with minimum travel cost and perception cost is selected. We validated the effectiveness of our planner in multiple simulation environments and



(a)



(b)

Fig. 9. Result from hardware experiments. (a) MAV generates a path pass to move through two waypoints and arrives at the goal point. (b) The resulting trajectories with the proposed planner (left) vs. the perception-agnostic planner (right). Blue lines indicate the groundtruth trajectory of MAV, and red lines are estimated position trajectory. With the perception-agnostic planner, the position estimate could be obtained during a part of the trajectory only, because of the early VO failure.

experiment. Future works would include an extension to online planning in unknown environments, reducing the need of tuning parameters (i.e. perception quality threshold  $c_{p,thr}$ ) via data-driven methods.

## REFERENCES

- [1] M. W. Achtelik, S. Lynen, S. Weiss, M. Chli, and R. Siegwart, “Motion-and uncertainty-aware path planning for micro aerial vehicles,” *Journal of Field Robotics*, vol. 31, no. 4, pp. 676–698, 2014.
- [2] S. Å. Sadat, K. Chutskoff, D. Jungic, J. Wawerla, and R. Vaughan, “Feature-rich path planning for robust navigation of mavs with monoslam,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 3870–3875.
- [3] G. Costante, J. Delmerico, M. Werlberger, P. Valigi, and D. Scaramuzza, “Exploiting photometric information for planning under uncertainty,” in *Robotics research*. Springer, 2018, pp. 107–124.
- [4] A. A. Makarenko, S. B. Williams, F. Bourgault, and H. F. Durrant-Whyte, “An experiment in integrated exploration,” in *IEEE/RSJ international conference on intelligent robots and systems*, vol. 1. IEEE, 2002, pp. 534–539.
- [5] P. Fogliaroni, J. O. Wallgrün, E. Clementini, F. Tarquini, and D. Wolter, “A qualitative approach to localization and navigation based on visibility information,” in *International Conference on Spatial Information Theory*. Springer, 2009, pp. 312–329.
- [6] G. J. Stein, C. Bradley, V. Preston, and N. Roy, “Enabling topological planning with monocular vision,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1667–1673.
- [7] S. M. LaValle, *Planning algorithms*. Cambridge university press, 2006.
- [8] A. Bry and N. Roy, “Rapidly-exploring random belief trees for motion planning under uncertainty,” in *2011 IEEE international conference on robotics and automation*. IEEE, 2011, pp. 723–730.
- [9] D. Falanga, P. Foehn, P. Lu, and D. Scaramuzza, “Pampc: Perception-aware model predictive control for quadrotors,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–8.
- [10] V. Murali, I. Spasojevic, W. Guerra, and S. Karaman, “Perception-aware trajectory generation for aggressive quadrotor flight using differential flatness,” in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 3936–3943.
- [11] L. Bartolomei, L. Pinto Teixeira, and M. Chli, “Perception-aware path planning for uavs using semantic segmentation,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2020)(virtual)*, 2020.
- [12] Z. Zhang and D. Scaramuzza, “Perception-aware receding horizon navigation for mavs,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 2534–2541.
- [13] Y. Jang, Y. Lee, and H. J. Kim, “Navigation-assistant path planning within a mav team,” 2020.
- [14] M. Collins and N. Michael, “Efficient planning for high-speed mav flight in unknown environments using online sparse topological graphs,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 11450–11456.
- [15] C. Rösmann, F. Hoffmann, and T. Bertram, “Integrated online trajectory planning and optimization in distinctive topologies,” *Robotics and Autonomous Systems*, vol. 88, pp. 142–153, 2017.
- [16] B. Lau, C. Sprunk, and W. Burgard, “Efficient grid-based spatial representations for robot navigation in dynamic environments,” *Robotics and Autonomous Systems*, vol. 61, no. 10, pp. 1116–1130, 2013.
- [17] H. Oleynikova, Z. Taylor, R. Siegwart, and J. Nieto, “Sparse 3d topological graphs for micro-aerial vehicle planning,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1–9.
- [18] R. Mur-Artal and J. D. Tardós, “Orb-slam2: An open-source slam system for monocular, stereo, and rgbd cameras,” *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.
- [19] C. Forster, M. Pizzoli, and D. Scaramuzza, “Svo: Fast semi-direct monocular visual odometry,” in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 15–22.
- [20] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, “Octomap: An efficient probabilistic 3d mapping framework based on octrees,” *Autonomous robots*, vol. 34, no. 3, pp. 189–206, 2013.
- [21] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, “Voxblox: Incremental 3d euclidean signed distance fields for onboard mav planning,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 1366–1373.
- [22] S. Bhattacharya, M. Likhachev, and V. Kumar, “Topological constraints in search-based robot path planning,” *Autonomous Robots*, vol. 33, no. 3, pp. 273–290, 2012.
- [23] C. Richter, A. Bry, and N. Roy, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments,” in *Robotics research*. Springer, 2016, pp. 649–666.
- [24] J. Park, J. Kim, I. Jang, and H. J. Kim, “Efficient multi-agent trajectory planning with feasibility guarantee using relative bernstein polynomial,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 434–440.
- [25] M. Zucker, N. Ratliff, A. D. Dragan, M. Pivtoraiko, M. Klingensmith, C. M. Dellin, J. A. Bagnell, and S. S. Srinivasa, “Chomp: Covariant hamiltonian optimization for motion planning,” *The International Journal of Robotics Research*, vol. 32, no. 9-10, pp. 1164–1193, 2013.
- [26] B. Zhou, F. Gao, L. Wang, C. Liu, and S. Shen, “Robust and efficient quadrotor trajectory generation for fast autonomous flight,” *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3529–3536, 2019.
- [27] J.-L. Blanco, J. Gonzalez, and J.-A. Fernandez-Madrigal, “Consistent observation grouping for generating metric-topological maps that improves robot localization,” in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. IEEE, 2006, pp. 818–823.
- [28] Z. Zhang and D. Scaramuzza, “Fisher information field: an efficient and differentiable map for perception-aware planning,” *arXiv preprint arXiv:2008.03324*, 2020.
- [29] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*. MIT press, 2009.
- [30] S. Shah, D. Dey, C. Lovett, and A. Kapoor, “Airsim: High-fidelity visual and physical simulation for autonomous vehicles,” in *Field and service robotics*. Springer, 2018, pp. 621–635.