



FUEL: Fast UAV Exploration using Incremental Frontier Structure and Hierarchical Planning

Abstract

- 본 논문에서는 복잡한 unknown environments에서 빠른 UAV Exploration이 되도록 도움을 주는 FUEL을 제안하였다.

Introduction

Unmanned aerial vehicles은 많은 분야에서 인기를 얻어왔다. 많은 method가 제안되어 왔지만, 거의 대부분은 큰 공간에 적용하지 못할만큼 좋은 결과를 내지는 못하였다. 하지만 본 논문에서 제안한 방법은 structure이 효율적으로 업데이트 될 수 있도록 한다.

System Overview

본 논문의 framework는 voxel grid map에서 작동된다. 이 method는 FIS의 증가하는 업데이트와 hierarchical exploration planning approach로 구성되어 있다. 센서를 통해 map이 업데이트 될 때에 frontier cluster이 영향을 받았는지 확인한다.

Incremental Frontier Information Structure

Frontier은 unknown voxel에 인접한 known-free voxel로 정의된다. 본 논문에서는 frontier에서 많은 정보를 뽑아내어 정교한 planning이 가능하게 하였다.

Frontier Information Structure

Frontier information structure인 FI_i 는 새로운 frontier cluster인 F_i 가 생성될 때 계산된다. FI_i 은 cluster 안에 속하는 모든 cell인 C_i 를 저장하고 평균 위치인 $p_{avg,i}$ 를 뽑아낸다. Axis-aligned bounding box인 B_i 도 frontier 변화의 detection을 가속 하기위해 연

산된다. Exploration planning을 위해 candidate viewpoints인 VP_i 가 cluster 근처에서 생성된다. 그리고 F_i 와 모든 cluster 간의 connection cost를 포함하는 $L_{cost,i}$ 가 계산된다.

Data	Explanation
C_i	Frontier cells that belong to the cluster
$\mathbf{p}_{avg,i}$	Average position of C_i
B_i	Axis-aligned bounding box of C_i
VP_i	Viewpoints covering the cluster
$L_{cost,i}$	Doubly linked list of connection costs to all other clusters

Incremental Frontier Detection and Clustering

map은 센서를 통해 업데이트 된다. 또한 그에 따라 B_m 도 업데이트 되는데, 먼저 모든 cluster를 통과하고 새로운 B_i 와 겹치는 부분만이 반환된다. 그 후, precise check가 겹치는 부분에 대해 시행되어 더 이상 frontier이 아닌 cell을 제거한다. 그 후에 새로운 frontier이 region growing 알고리즘을 통해 탐색되고 그룹으로 cluster 된다. 그 그룹 중에서 cell 수가 적은 그룹은 무시한다. 남은 그룹이 너무 큰 cluster를 가지게 될 수도 있다. 이를 해결하기 위해 cluster에 PCA를 진행하여 여러 작은 cluster으로 나눈다.

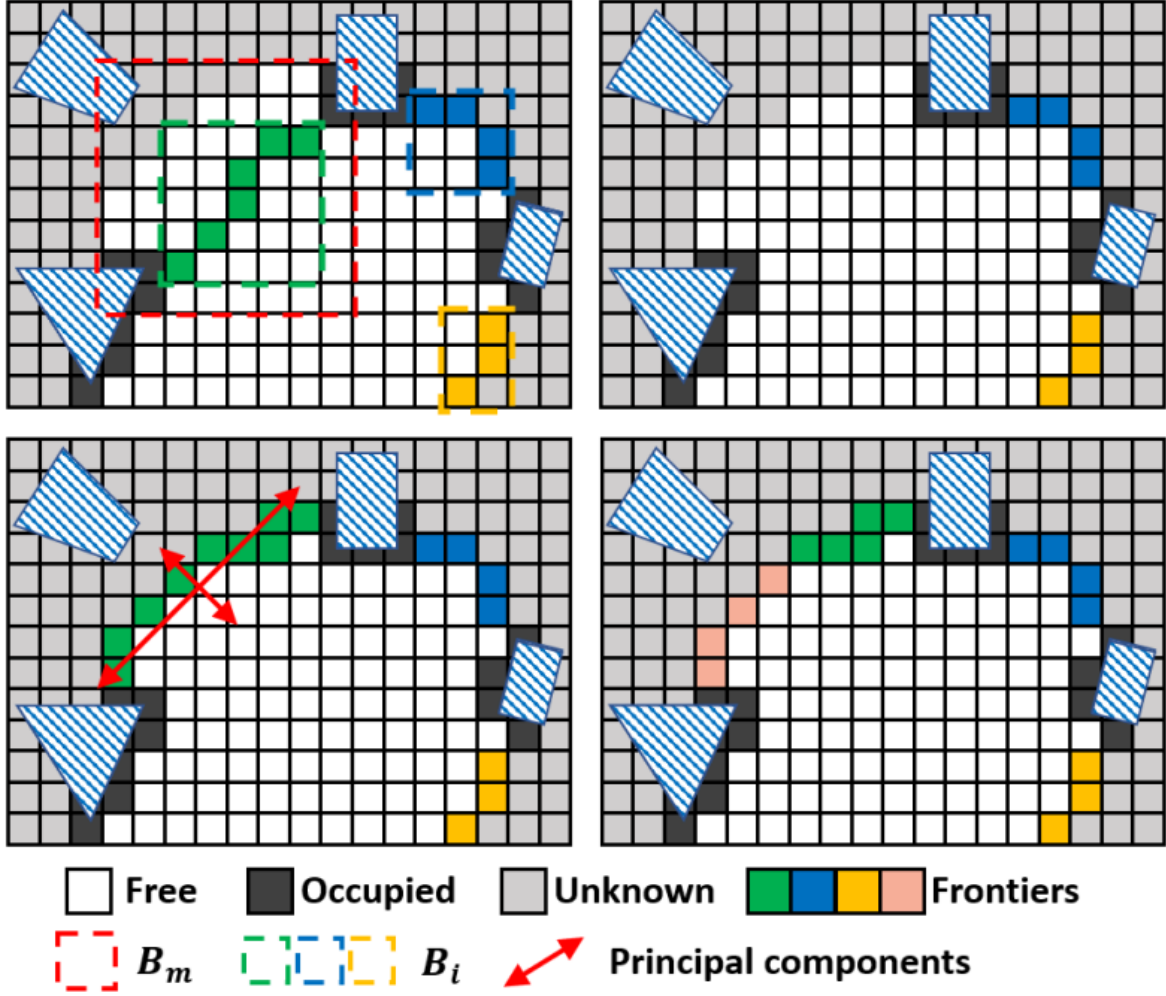


Fig. 3. Incremental frontier detection and clustering. Top: detecting and removing outdated frontiers. Bottom: new frontier is detected (left) and PCA is performed, the large cluster is split into two smaller ones (right).

Viewpoint Generation and Cost Update

본 논문의 frontier cluster이란 직관적으로 탐색할 잠재적 공간을 의미한다. cluster의 중심 부분으로 찾아가던 이전의 method와는 다르게, 본 논문의 저자들은 더욱 정교한 decision making을 추구하였다. 이를 위해 cluster인 F_i 가 생성될 때에 viewpoints의 집합인 $VP_i = \{x_{i,1}, x_{i,2}, \dots, x_{i,n_i}\}$, where $x_{i,j}(p_{i,j}, \xi_{i,j})$ 를 생성한다. 이는 cluster의 중심을 기반으로 원통형 좌표계에서 점을 균일하게 sampling하여 구한다.

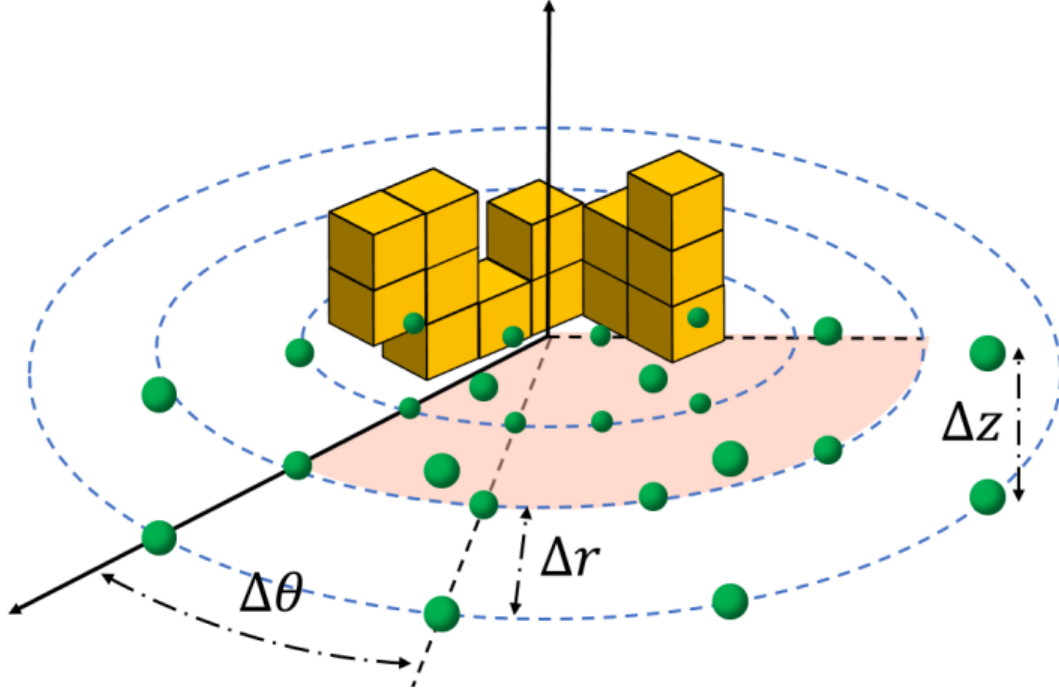


Fig. 4. Generating candidate viewpoints for a frontier cluster. Within the cylindrical coordinate system centered at the average position of the cluster, points are sampled uniformly.

각각의 sampled point인 p 에 대해 yaw angle인 ξ 는 cluster에 대한 센서 coverage를 yaw optimization을 통해 최대화 하는 것으로 결정된다. Coverage는 occupied voxel에 가려지지 않은 frontier 수와 sensor model을 준수하는 voxel 수로 평가된다. 그 후 threshold 보다 높은 viewpoint가 내림차순으로 정렬된다.

Global planning of exploration tour을 수행하기 위해, 각 cluster의 pair 사이의 connection cost가 필요하다.

$$t_{lb}(\mathbf{x}_{k_1, j_1}, \mathbf{x}_{k_2, j_1}) = \max \left\{ \frac{\text{length}(P(\mathbf{p}_{k_1, j_1}, \mathbf{p}_{k_2, j_2}))}{v_{\max}}, \frac{\min(|\xi_{k_1, j_1} - \xi_{k_2, j_2}|, 2\pi - |\xi_{k_1, j_1} - \xi_{k_2, j_2}|)}{\dot{\xi}_{\max}} \right\}, \quad (1)$$

- $t_{lb}(x_{k_1, j_1}, x_{k_2, j_1})$: 두 viewpoint인 $x_{k_1, j_1}, x_{k_2, j_1}$ 를 이동할 때의 time lower bound
- $P(p_{k_1, j_1}, p_{k_2, j_1})$: $p_{k_1, j_1}, p_{k_2, j_1}$ 사이의 충돌이 없는 경로
-