

---

# Map Metrics Documentation

*Release 0.0.4*

**Anastasiia Kornilova**

**Mar 08, 2022**



**CONTENTS:**

<b>1</b>	<b>Installation</b>	<b>1</b>
1.1	Latest release . . . . .	1
<b>2</b>	<b>Usage</b>	<b>3</b>
2.1	Mean Map Entropy (MME) . . . . .	3
2.2	Mean Plane Variance (MPV) . . . . .	3
2.3	Mutually Orthogonal Metric (MOM) . . . . .	4
<b>3</b>	<b>Indices and tables</b>	<b>5</b>



## INSTALLATION

### 1.1 Latest release

To install Map Metrics, run this command in your terminal:

```
$ pip install -U pip  
$ pip install -U map_metrics
```



To use Map Metrics in a project:

```
import map_metrics
```

## 2.1 Mean Map Entropy (MME)

Mean Map Entropy calculates average entropy of every point vicinity in the aggregated map:

$$h(p_k) = \frac{1}{2} \det(2\pi e \Sigma(W(p_k)))$$
$$H(P) = \frac{1}{|P|} \sum_{k=1}^{|P|} h(p_k)$$

To use it, provide *pcs* — a list of point clouds pcs in the form of *open3d.PointCloud* and *Ts* — a list of corresponding poses in the trajectory in the form of *4x4* transformation matrices.

```
map_metrics.mme(pcs, Ts)
```

## 2.2 Mean Plane Variance (MPV)

Mean Plane Variance calculates average plane variance of every point vicinity in the aggregated map:

$$V(P) = \frac{1}{|P|} \sum_{k=1}^{|P|} v(p_k) = \frac{1}{|P|} \sum_{k=1}^{|P|} \lambda_{min}$$

To use it, provide *pcs* — a list of point clouds pcs in the form of *open3d.PointCloud* and *Ts* — a list of corresponding poses in the trajectory in the form of *4x4* transformation matrices.

```
map_metrics.mpv(pcs, Ts)
```

## 2.3 Mutually Orthogonal Metric (MOM)

Mutual Orthogonality is a concept of considering not all points in the map but only ones from mutually orthogonal surfaces. Mean Plane Variance over those points provides (as described in our paper) correlation with Relative Pose Error (RPE) — one of the popular full-reference metrics for trajectories.

To use it, provide *pcs* — a list of point clouds *pcs* in the form of *open3d.PointCloud* and *Ts* — a list of corresponding poses in the trajectory in the form of  $4 \times 4$  transformation matrices.

```
map_metrics.mom(pcs, Ts)
```

The default usage of the method assumes extraction of points from mutually orthogonal surfaces during the method execution and therefore increase calculation time. One can extract those points manually one time for specific set of point cloud and use it as parameter to calculate MOM faster.

```
orth_list, _, _ = map_metrics.extract_orthogonal_subsets(pcs[0])
print(map_metrics.orth_mme(pcs, Ts_gt, orth_list=orth_list))
```



## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`