

# Effiziente Berechnung eines voxelbasierten Umgebungsmodells für eine echtzeitfähige Informationsverteilung in der Robotik

## Efficient Calculation of a voxel-based Environment Model for an Online Information Distribution in Robotics

Heiko Renz, Technische Universität Dortmund, Lehrstuhl für Regelungssystemtechnik, 44227 Dortmund, Deutschland, heiko.renz@tu-dortmund.de

Univ.-Prof. Dr.-Ing. Prof. h.c. Dr. h.c. Torsten Bertram, Technische Universität Dortmund, Lehrstuhl für Regelungssystemtechnik, 44227 Dortmund, Deutschland, torsten.bertram@tu-dortmund.de

### Kurzfassung

Umgebungsmodelle sind ein essentieller Bestandteil für verschiedene Aufgaben in der Robotik. Verschiedene Ansätze zur Planung einer Robotertrajektorie oder zur Navigation benötigen ein Umgebungsmodell, um eine kollisionsfreie Lösung zu bestimmen. Eine weitere Applikation ist die fortwährende Beobachtung von Änderungen zur Detektion von ausgeführten Aufgaben. Je nach Applikation werden verschiedene Anforderungen an das Modell gestellt. Während ein Umgebungsmodell zur Kollisionsvermeidung oftmals nur eine rudimentäre Abschätzung von Hindernissen benötigt, ist für die Detektion von Änderungen eine detaillierte Modellierung notwendig. Zusätzliche Anforderungen werden, je nach Anwendung, an die Laufzeit zur Bestimmung des Modells gestellt. Der Beitrag stellt daher verschiedene Filtervariationen zur Verbesserung der Laufzeit unter Berücksichtigung des Genauigkeitsverlustes vor. Des Weiteren wird eine Methode zur Bestimmung einer 3D Informationsverteilung in einem voxelbasierten Umgebungsmodell zur Diskussion gestellt.

Verschiedene Umgebungsmodelle nutzen zur Darstellung im Arbeitsraum variierende Datenstrukturen. Eine verbreitete Methode sind voxelbasierte Umgebungsmodelle wie *OctoMap* [1]. Dabei wird eine Octree-Repräsentation gewählt, welche es ermöglicht effizient variierende Auflösungen zu verwenden, in dem weitere Voxel hinzugefügt werden, sobald dies nötig wird. Eine Variation der *OctoMap* stellt die *UFOMap* dar. Diese erlaubt zusätzlich zur Modellierung freier und belegter Voxel auch die explizite Modellierung unbekannter Voxel [2]. Die Kenntnis über unbekannte Voxel ist für eine Informationsverteilung von Bedeutung, da dort ein höherer Informationsgewinn erwartet wird. In der Voxelkarte werden anschließend mit Hilfe eines Raycastings Perspektiven ausgewertet, um einen Informationsgewinn für jede Perspektive zu bestimmen. Dabei werden mögliche Perspektiven oftmals zufällig gesampelt und die Informationsgewinne der Perspektiven sequentiell bestimmt, was zu hohen Laufzeiten führt [3]. Der Informationsgewinn wird auch zur Bestimmung explorativer Trajektorien für Drohnen verwendet, wobei der Fokus dabei auf der Berechnung des lokal erreichbaren Informationsgewinns liegt [6]. Für die Nutzung in Online-Anwendungen der Robotik ist eine effiziente Berechnung der Informationsverteilung nötig, um eine aktuelle Verteilung bei der iterativen Trajektorienplanung bereitzustellen.

Im Folgenden werden Filtervariationen präsentiert und hinsichtlich der Genauigkeit sowie der Laufzeit einer *OctoMap* evaluiert. Eine hohe Frequenz zur Erzeugung des voxelbasierten Umgebungsmodells ist dabei von Bedeutung, da die Umgebung eines Roboters sich fortwährend ändert und für eine echtzeitfähige Informationsverteilung ein aktuelles Modell vorliegen muss. Zur Reduktion der Laufzeit werden daher ein Voxel-Filter und ein Outlier-Filter aus der *Point Cloud Library* [4] auf die Punktwolke einer Azure Kinect RGBD Kamera angewendet, um die Anzahl der zu verarbeitenden Daten zu reduzieren. Des Weiteren ist die wählbare Auflösung der *OctoMap* eine zentrale Größe zur Beeinflussung der Laufzeit. Die zur Berechnung der *OctoMap* verwendete Hardware ist mit einem Intel i5-12600KF mit 3,7 GHz und 32 GB RAM ausgestattet. Für jede Variation wird ein zuvor aufgezeichnetes Datenset abgespielt und die Laufzeit zur Aktualisierung der *OctoMap* gemessen und die finale Voxelkarte gespeichert. Variiert werden dabei die Auflösung der *OctoMap*  $s_{Oct}$  sowie die Anwendung der Filter und die Filterparameter. Für das Voxel-Filter wird die Voxelgröße  $s_{Vox}$  und für das

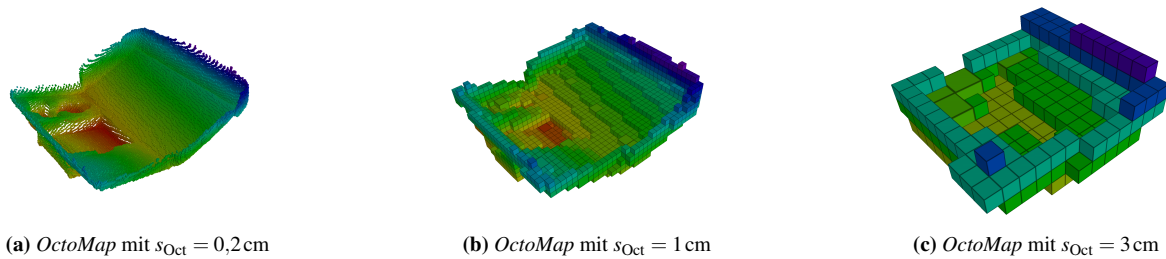
**Tabelle 1** Laufzeit- und Genauigkeitsergebnisse für verschiedene *OctoMap* Auflösungen  $s_{Oct}$  ohne Filter

$s_{Oct}$ [cm]	$f_{Oct}$ [Hz]	$N_{OL}$ [%]	$V_{Add}$ [%]
0,2	0,27		
0,5	0,43	98,90	63,7
1	0,67	99,98	82,2
3	1,33	100,00	94,4
5	1,71	100,00	96,6
7,5	2,05	100,00	98,0
10	2,31	100,00	98,5

**Tabelle 2** Laufzeit- und Genauigkeitsergebnisse für verschiedene Filterparameter bei  $s_{Oct} = 1$  cm

$s_{Vox}$ [cm]	$N_{Out}$ [ ]	$r_{Out}$ [cm]	$f_{Filt}$ [Hz]	$f_{Oct}$ [Hz]	$N_{OL}$ [%]	$V_{Add}$ [%]
0,5			30	2,27	99,93	81,98
1			30	6,96	99,89	81,82
2			30	21,30	44,16	33,47
	2	10	6	0,71	99,96	82,23
	10	10	4	0,71	99,97	82,25
	2	1	5	0,71	99,97	82,26
	10	1	4	0,71	99,92	82,14
1	2	10	30	6,96	99,89	81,86





**Abbildung 1** Vergleich verschiedener *OctoMap* Auflösungen

Outlier-Filter werden die Anzahl der Nachbarn  $N_{\text{Out}}$  sowie der Radius der Nachbarschaft  $r_{\text{Out}}$  variiert. Tabelle 1 zeigt die Laufzeit- und Genauigkeitsergebnisse unter Variation der Auflösung der *OctoMap*  $s_{\text{Oct}}$  ohne Nutzung der Filter. Die Frequenz der *OctoMap*  $f_{\text{Oct}}$  steigt mit wachsendem  $s_{\text{Oct}}$ , da die Erstellung der *OctoMap* weniger Voxel bestimmt. Zur Evaluation der Genauigkeit werden die *OctoMaps* mit  $s_{\text{Oct}} > 0,2\text{ cm}$  mit der *OctoMap* mit  $s_{\text{Oct}} = 0,2\text{ cm}$  verglichen.  $N_{\text{OL}}$  bestimmt dabei den Anteil an Voxelmittelpunkten der *OctoMap* mit  $s_{\text{Oct}} = 0,2\text{ cm}$ , welche in Voxeln der niedrig auflösenden *OctoMap* liegen.  $V_{\text{Add}}$  berechnet den Anteil des zusätzlichen Volumens der niedrig auflösenden *OctoMap* im Vergleich zum Referenzvolumen  $V_{\text{Ref}} = 0,417\text{ dm}^3$  der hochauflösenden *OctoMap* mit  $s_{\text{Oct}} = 0,2\text{ cm}$  relativ zum Gesamtvolumen  $V_k$  der niedrig auflösenden *OctoMap*:  $V_{\text{Add}} = (V_k - V_{\text{Ref}})/V_k$ . Bei einer Auflösung von  $s_{\text{Oct}} \geq 3\text{ cm}$  werden alle Referenzvoxel abgedeckt. 94,4 % des Gesamtvolumens  $V_k$  bei  $s_{\text{Oct}} = 3\text{ cm}$  entfallen aber auf ein zusätzliches Volumen. Ein wachsendes  $s_{\text{Oct}}$  deckt zwar mehr Referenzvoxel ab, belegt aber zusätzlichen Arbeitsraum. Dies kann beispielsweise dazu führen, dass ein Roboter sich zu beobachtenden Objekten nicht mehr weiter nähern kann, da das Umgebungsmodell einen größeren Bereich als belegt markiert. Bild 1 zeigt die *OctoMap* für verschiedene Auflösungen  $s_{\text{Oct}}$ . Zur Untersuchung des Filtereinflusses werden die Filterparameter  $s_{\text{Vox}}$ ,  $N_{\text{Out}}$  und  $r_{\text{Out}}$  für  $s_{\text{Oct}} = 1\text{ cm}$  variiert und Tabelle 2 zeigt die Laufzeit- und Genauigkeitsergebnisse. Die Metriken  $N_{\text{OL}}$  und  $V_{\text{Add}}$  werden dabei auf die Referenzdarstellung mit  $s_{\text{Oct}} = 0,2\text{ cm}$  bezogen. Die Ergebnisse zeigen, dass das Voxelfilter die Frequenz der *OctoMap* erhöht, da weniger Punkte verarbeitet werden müssen. Bei der Auslegung des Voxelfilters sollte beachtet werden, dass die Voxelgröße  $s_{\text{Vox}}$  kleiner als die Auflösung der *OctoMap*  $s_{\text{Oct}}$  gewählt wird, da ansonsten Punkte gefiltert werden, welche zu fehlenden Voxeln in der *OctoMap* führen (vgl. Tab. 1 Zeile 3 und Tab. 2 Zeile 1-3). Das Outlier-Filter reduziert die Frequenz  $f_{\text{Oct}}$ , da bereits die Frequenz  $f_{\text{Filt}}$  der gefilterten Punktwolke für alle vier Konfigurationen zwischen 4 Hz bis 6 Hz liegt. Somit ist das Outlier-Filter für eine online Umgebungsmodellierung nicht geeignet, da die Frequenz deutlich sinkt (vgl. Tab. 1 Zeile 3 und Tab. 2 Zeile 4-8). Weitere Experimente zeigen, dass sich vergleichbare Resultate für die Variation der Filterparameter auch bei der Nutzung der *UFOMap* ergeben. Die Auswahl einer geeigneten Voxelauflösung ist an die entsprechende Applikation sowie Umgebung anzupassen.

Zur Bestimmung einer Informationsverteilung wird ein Raycasting für verschiedene Perspektiven durchgeführt und die Anzahl der unbekanntem Voxel in der Perspektive bestimmt. Eine mögliche Erweiterung der Informationsverteilung ist die Berücksichtigung des Sensormodells unter Anwendung der Fisher-Information [5]. Für eine effiziente Implementierung, die eine hohe Anzahl an Perspektiven in kurzer Zeit auswerten kann, wird eine parallele Implementierung des Raycastings angestrebt. Durch Nutzung einer GPU kann eine solche Parallelisierung erreicht werden, indem rechenintensive Operationen parallelisiert auf der GPU ausgeführt werden. Erste Experimente zeigen dabei eine Zeitreduktion um bis zu Faktor 70 im Vergleich zur sequentiellen Ausführung auf der CPU zur Berechnung einer Informationsmetrik in einer *UFOMap*.

Die Autoren bedanken sich für die finanzielle Unterstützung des Projektes durch die Deutsche Forschungsgemeinschaft (DFG, Projektnummer 497071854).

## Literatur

- [1] Hornung, A.; Wurm, K. M.; Bennewitz, M.; Stachniss, C.; Burgard, W.: *OctoMap: An efficient probabilistic 3D mapping framework based on octrees*. Autonomous Robots. (2013), Volume 34-3, S. 189-206.
- [2] Duberg, D.; Jensfelt, P.: *UFOMap: An efficient probabilistic 3D mapping framework that embraces the unknown*. IEEE Robotics and Automation Letters. (2020), Volume 5-4, S. 6411-6418.
- [3] Santos, J.; Oliveira, M.; Arrais, R.; Veiga, G.: *Autonomous scene exploration for robotics: A conditional random view-sampling and evaluation using a voxel-sorting mechanism for efficient ray casting*. Sensors. (2020), Volume 20-15, S. 4331.
- [4] Rusu, R. B.; Cousins, S.: *3D is here: Point cloud library (PCL)*. In IEEE International Conference on Robotics and Automation (ICRA). 9-13 Mai 2011 Shanghai, China. S. 1-4.
- [5] Zhang, Z.; Scaramuzza, D.: *Fisher information field: An efficient and differentiable map for perception-aware planning*. <https://arxiv.org/pdf/2008.03324>. (2020), S.1-18.
- [6] Liang, L.; Redondo, C.; Campoy P.: *Optimal frontier-based autonomous exploration in unconstructed environment using RGB-D sensor*. Sensors. (2020), Volume 20-22, S. 6507.

# DuEPublico

Duisburg-Essen Publications online

UNIVERSITÄT  
DUISBURG  
ESSEN

*Offen im Denken*

ub | universitäts  
bibliothek

In: Zehnte IFToMM D-A-CH Konferenz 2024

Dieser Text wird via DuEPublico, dem Dokumenten- und Publikationsserver der Universität Duisburg-Essen, zur Verfügung gestellt. Die hier veröffentlichte Version der E-Publikation kann von einer eventuell ebenfalls veröffentlichten Verlagsversion abweichen.

**DOI:** 10.17185/duepublico/81614

**URN:** urn:nbn:de:hbz:465-20240304-101810-0



Dieses Werk kann unter einer Creative Commons Namensnennung 4.0 Lizenz (CC BY 4.0) genutzt werden.