Solution de génération de formes 2D d'objets basée sur des connaissances préalables pour la cartographie sémantique

Abdessalem ACHOUR[1,2] Hiba AL ASSAAD[1] Yohan DUPUIS[3] Madeleine EL ZAHER[1]

[1]CESI LINEACT, Campus de Toulouse, 31670 Labège, France.

[2]École doctorale SMI, HESAM Université, 75013 Paris, France.

[3]CESI LINEACT, Paris La Défense, 92074 Paris, France.

Résumé – Dans ce travail, nous proposons une solution de cartographie sémantique en temps réel basée sur les données RGBD. Nous nous focalisons sur la proposition d'une approche d'association pour générer les formes 2D des objets sémantiques en utilisant des connaissances préalables. Notre approche est évaluée dans un environnements bureautique à l'aide du robot mobile MIR. Les résultats expérimentaux et une comparaison avec une approche existante montrent que notre proposition permet de générer des cartes proches de la vérité terrain.

Abstract – This study presents an RGBD-based real-time semantic mapping solution. We focus on presenting a novel association approach to generate 2D semantic object shapes based on prior knowledge. We evaluated our approach in an office environment using the mobile robot MIR. Our experimental results and comparison with an existing approach show that our proposal generates maps that are very close to the ground truth.

1 INTRODUCTION

Afin d'accomplir efficacement des tâches complexes telles que la manipulation d'objets et la navigation sémantique, un robot mobile doit disposer d'une compréhension cognitive de son environnement. Les cartes sémantiques répondent à ce besoin en introduisant des informations sémantiques dans la carte, telles que les catégories d'objets, leurs formes, leurs modèles 3D et les différentes relations entre eux [1]. Dans nos travaux, nous nous intéressons particulièrement à la navigation sémantique, pour laquelle une représentation 2D des objets peut s'avérer suffisante [7].

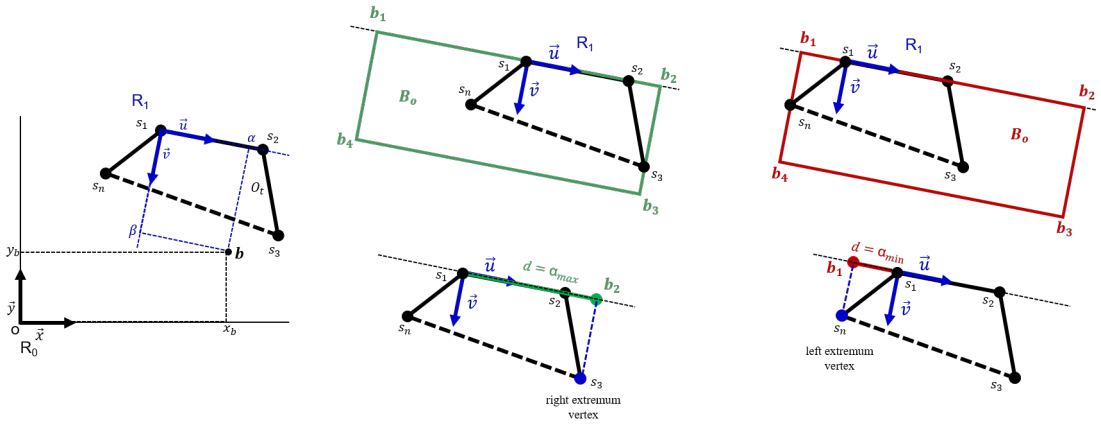
Dans la littérature, certains travaux ont abordé la question de la cartographie sémantique 2D. Par exemple, [5] utilise l'odométrie et la vision stéréo pour la détection et la triangulation des objets dans un environnement domestique. Les nuages de points d'objets sont étiquetés, ensuite regroupés à l'aide des rectangles englobants minimaux pour définir l'espace topologique de chaque objet. L'approche proposée dans [7] s'intéresse à l'exploration sémantique autonome. Cette approche se base sur l'algorithme Quickhull [2] pour donner une approximation des formes des objets à partir du nuage de points projeté au sol. De même, Dengler *et al.* [3] utilisent un algorithme de segmentation géométrique pour segmenter le nuage de points et un modèle de détection basé sur le CNN pour la détection des instances. Chaque segment de point est assigné à une détection, ensuite la forme 2D des objets est représentée par des polygones. Les points du modèle 3D sont projetés sur le plan 2D et l'algorithme Quickhull est appliqué pour calculer le polygone correspondant. La structure R-tree [4] est utilisée pour identifier les correspondances et combiner les polygones. Les travaux cités utilisent principalement des données de capteurs pour déterminer la zone d'occupation d'un objet. Cependant, dans de nombreux environnements, des connaissances préalables sur les modèles d'objets sont disponibles. Par exemple, dans les environnements industriels, les modèles d'objets sont souvent construits pour être utilisés dans des jumeaux numériques.

Dans cet article, nous présentons une solution de cartographie sémantique basée sur les données RGBD qui utilise les dimensions réelles des objets pour améliorer l'approximation de leur zone d'occupation en 2D. Notre méthode fournit une représentation polygonale simplifiée de la carte sémantique, puis utilise les dimensions extraites des modèles 3D des objets pour les représenter avec des boîtes englobantes de taille exacte. Pour déterminer la position optimale de la boîte englobante, nous proposons un algorithme d'association qui trouve la meilleure association avec le polygone représentant l'objet. Cette méthode offre une approximation précise des zones d'occupation des objets partiellement visibles ou occultés, permettant ainsi d'obtenir une carte sémantique plus précise. Nous évaluons notre méthode dans un environnement bureautique et la comparons à une autre approche [3].

2 Description de la méthode

2.1 Description de la problématique

La solution de cartographie sémantique présentée dans cette étude suppose que la position globale du robot est connue et qu'une carte d'occupation a déjà été créée pour déterminer sa position. Tout d'abord, les objets sont identifiés à partir de l'image RGB à l'aide d'un modèle de détection, et leurs modèles 3D sont déterminés à partir de l'image de profondeur à l'aide d'un algorithme de segmentation rapide. Ensuite, pour représenter leur étendue spatiale, le modèle 3D de chaque objet est projeté sur le sol et l'algorithme Quickhull [2] est appliqué au nuage de points résultant pour générer une représentation polygonale convexe de l'objet. Le polygone obtenu ne représente que la partie visible de l'objet au moment de la détection.



(a) Représentation du polygone et du référentiel local défini pour la première arête.

(b) Création de la boîte englobante décalée à gauche.

(c) Création de la boîte englobante décalée à droite.

FIGURE 1 : Les étapes de la création des boîtes englobantes décalées à gauche et à droite pour la première arête.

En intégrant la connaissance préalable des dimensions de la boîte englobante associée à l'objet, ce polygone est augmenté pour créer une représentation plus précise basée sur une boîte englobante orientée. La solution de cartographie est progressif pour compléter les parties manquantes et mettre à jour la forme et l'emplacement de l'objet dans son ensemble. Dans la suite de cette étude, le polygone représentant l'objet à chaque instant est considéré comme déjà construit à l'aide de l'algorithme Quickhull, et l'accent est mis sur la méthode d'association pour générer la représentation de la boîte englobante.

2.2 La base de connaissances préalables

La base de connaissances préalables est une liste d'objets présents dans l'environnement, chacun décrit par la longueur l et la largeur w de la boîte rectangulaire englobant la projection de son modèle 3D au sol. Les boîtes englobantes sont couramment utilisées dans les approches de cartographie pour représenter les zones d'occupation des objets, car elles peuvent représenter une grande variété d'objets dans le monde réel, allant de formes géométriques simples à des structures complexes. L'environnement considéré dans notre étude est un environnement bureautique, donc des objets tels que des tables, des chaises, des bureaux, etc. peuvent être présents. Pour l'implémentation initiale de notre solution, un seul modèle par catégorie d'objet est considéré, représenté par une boîte rectangulaire définie par les sommets b_1 à b_4 dans le sens des aiguilles d'une montre, soit $B_o = \{b_1, b_2, b_3, b_4\}$.

2.3 Méthode d'association géométrique en 2D

La méthode vise à déterminer la position et l'orientation d'un objet en associant sa boîte englobante prédéfinie B_o à son polygone partiel $O_t = \{s_i, i = 1, ..., n\}$ à l'instant t. Les sommets s_i sont donnés par les coordonnées (x_{s_i}, y_{s_i}) dans le référentiel global $R_0(o, \overrightarrow{x}, \overrightarrow{y})$ et sont disposés dans le sens des aiguilles d'une montre (Fig. 1.a). Deux boîtes englobantes sont créées pour chaque arête, l'une décalée vers la gauche (boîte verte Fig. 1.b) et l'autre vers la droite (boîte rouge Fig. 1.c), afin d'identifier les connexions potentielles entre la boîte englobante de l'objet et les bords de son polygone. Chaque association est évaluée en calculant un score d'association, et la boîte ayant le meilleur score est sélectionnée pour représenter l'objet.

2.4 Description de l'algorithme

2.4.1 Génération des boîtes englobantes

Chaque arête $e_i = \{s_i, s_{i+1}\}$ est représentée par deux points s_i et s_{i+1}. Pour chaque arête, un repère local $R_i(s_i, \overrightarrow{u}, \overrightarrow{v})$ est défini, comme illustré sur Fig. 1.a. Le vecteur $\overrightarrow{u}$ est le vecteur unitaire directeur de e_i, tandis que le vecteur $\overrightarrow{v}$ est le vecteur unitaire normal à e_i pointant vers l'intérieur du polygone.

Soit $\overrightarrow{u} = \begin{bmatrix} \Delta_x & \Delta_y \end{bmatrix}^T$ et $\overrightarrow{v} = \begin{bmatrix} \Delta_y & -\Delta_x \end{bmatrix}^T$, avec :

$$\Delta_x = \frac{x_{s_{i+1}} - x_{s_i}}{\|e_i\|}, \quad \Delta_y = \frac{y_{s_{i+1}} - y_{s_i}}{\|e_i\|}$$

où $\|e_i\|$ représente la norme de l'arête e_i. $\overrightarrow{v}$ est obtenu en effectuant une rotation de $-\frac{\pi}{2}$ autour de l'axe $\overrightarrow{u}$.

La solution proposée utilise le repère local R_i pour déterminer les paramètres d'association et les coordonnées des sommets de la boîte englobante. Cependant, pour mettre à jour ultérieurement la boîte sur la carte, il est nécessaire d'exprimer ses coordonnées dans le repère global de la carte. L'équation 1 permet de transformer les coordonnées locales (α, β) d'un point b défini dans R_i en coordonnées globales (x_b, y_b) dans le repère global R_0, comme illustré sur Fig. 1.a.

$$\begin{bmatrix} x_b \\ y_b \end{bmatrix} = \begin{bmatrix} x_{s_i} \\ y_{s_i} \end{bmatrix} + \begin{bmatrix} \Delta_x & \Delta_y \\ \Delta_y & -\Delta_x \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (1)$$

Considérons $A = \begin{bmatrix} \Delta_x & \Delta_y \\ \Delta_y & -\Delta_x \end{bmatrix}$ et $B = \begin{bmatrix} x_b - x_{s_i} \\ y_b - y_{s_i} \end{bmatrix}$, l'équation 2 permet, par la transformation inverse, de déterminer les coordonnées locales (α, β) de b à partir des coordonnées globales (x_b, y_b).

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} = A^{-1}B = \begin{bmatrix} \Delta_x & \Delta_y \\ \Delta_y & -\Delta_x \end{bmatrix} \begin{bmatrix} x_b - x_{s_i} \\ y_b - y_{s_i} \end{bmatrix} \quad (2)$$

Pour créer les boîtes à décalage dans le repère global R_0, il faut tout d'abord, déterminer la distance de décalage $d \in \mathbb{R}$ le long de l'axe u. Pour un décalage vers la gauche (Fig.1.b), la distance de décalage est égale à α_{max}, le α du sommet

extrême droit, et pour un décalage vers la droite (Fig.1.c), la distance de décalage est égale à α_{min}, le α du sommet extrême gauche. Pour ce faire, les coordonnées locales (α, β) de tous les sommets du polygone sont calculées à l'aide de l'équation 2. Ensuite, les sommets ayant α_{min} et α_{max} sont retenus.

Ensuite, il faut déterminer les coordonnées de b_1 pour un décalage vers la droite et de b_2 pour un décalage vers la gauche, tout en alignant l'arête $b_{12} = \{b_1, b_2\}$ avec e_i, et en faisant passer l'arête perpendiculaire, soit $b_{14} = \{b_1, b_4\}$, soit $b_{23} = \{b_2, b_3\}$, par α_{min} ou α_{max} respectivement. Les coordonnées de b_1 ou b_2 peuvent être calculées à l'aide de l'équation 3. La valeur de d dans l'équation dépend de l'indice m, qui correspond au sommet que l'on veut déterminer. Pour $m = 1$, $d = \alpha_{min}$, et pour $m = 2$, $d = \alpha_{max}$.

$$\begin{bmatrix} x_{b_m} \\ y_{b_m} \end{bmatrix} = \begin{bmatrix} x_{s_i} \\ y_{s_i} \end{bmatrix} + d\overrightarrow{\mathrm{u}} \tag{3}$$

Une fois que les coordonnées d'un des sommets de la boîte englobante rectangulaire, b_1 ou b_2, ont été déterminées, l'équation 4 est utilisée pour trouver les coordonnées des autres sommets en se basant sur les dimensions préalables de la boîte, l et w. Les valeurs de a et b dans l'équation dépendent de l'indice m, qui correspond au sommet que l'on veut déterminer. Pour $m = 2$, on prend $a = l$ et $b = 0$, pour $m = 3$, $a = l$ et $b = w$, et pour $m = 4$, $a = 0$ et $b = -w$.

$$\begin{bmatrix} x_{b_m} & y_{b_m} \end{bmatrix}^T = \begin{bmatrix} x_{b_1} & y_{b_1} \end{bmatrix}^T + a\overrightarrow{u} + b\overrightarrow{v} \tag{4}$$

Tout d'abord, des boîtes englobantes de dimensions $\{l, w\}$ sont générées. Ensuite, les dimensions sont inversées et des boîtes englobantes de dimensions $\{w, l\}$ sont générées. Ainsi, toutes les candidates boîtes englobantes sont générées.

2.4.2 Étapes de pré-traitement pour l'association

Avant le processus d'association, deux étapes de pré-traitement sont appliquées à la représentation polygonale générée par l'algorithme Quickhull pour améliorer l'efficacité. La première étape consiste à simplifier la représentation en combinant les segments avec des angles supérieurs à 178 degrés pour éviter la sur-segmentation due à la densité du nuage de points. La deuxième étape sélectionne uniquement les arêtes de premier plan pour l'association, qui sont plus susceptibles d'être observées par le robot. Pour déterminer si une arête est au premier plan, un triangle est calculé à partir de la position du robot et des sommets de l'arête. Si le triangle ainsi obtenu n'a pas d'intersection avec le polygone, alors il s'agit d'une arête de premier plan. L'algorithme d'écrêtage de Weiler-Atherton [6] est utilisé pour évaluer l'intersection.

2.4.3 Calcul du score d'association

Pour choisir la meilleure boîte englobante pour représenter un objet, il est nécessaire de calculer une fonction de score $\mathcal{S}$ pour chaque boîte englobante générée afin d'évaluer la qualité de l'association. Les boîtes dont le score dépasse un seuil de sélection ϵ sont conservées, et celle avec le score le plus élevé est choisie comme représentation finale de l'objet. Cette fonction est définie par l'équation $\mathcal{S} = 1 - (w_1 f_1 + w_2 f_2 + w_3 f_3)$, où f_1, f_2 et f_3 mesurent des caractéristiques liées respectivement à l'écart d'angle, de longueur et de largeur entre le polygone et la boîte englobante. w_1, w_2 et w_3 sont les poids associés respectant la contrainte $w_1 + w_2 + w_3 = 1$. Pour la première version de la solution, ces poids sont définis par des tests exhaustifs. Cependant, une approche basée sur l'apprentissage automatique est envisagée pour améliorer la méthode et la rendre plus robuste.

Définition de f_1 : Afin de réduire l'écart entre l'angle du polygone et celui de la boîte englobante, l'angle du polygone doit tendre vers $90°$. Pour cela, il faut minimiser les valeurs de α_{min} pour un déplacement vers la droite et $\alpha_{max} - \|e_i\|$ pour un déplacement vers la gauche. La fonction f_1 est définie comme suit : $f_1 = (|\alpha_i| - i * \|e_i\|)/l$, où i est un paramètre qui prend les valeurs 0 et 1, correspondant respectivement à un décalage vers la droite et vers la gauche. L'angle α_i est égal à α_{min} lorsque $i = 0$ et à α_{max} lorsque $i = 1$.

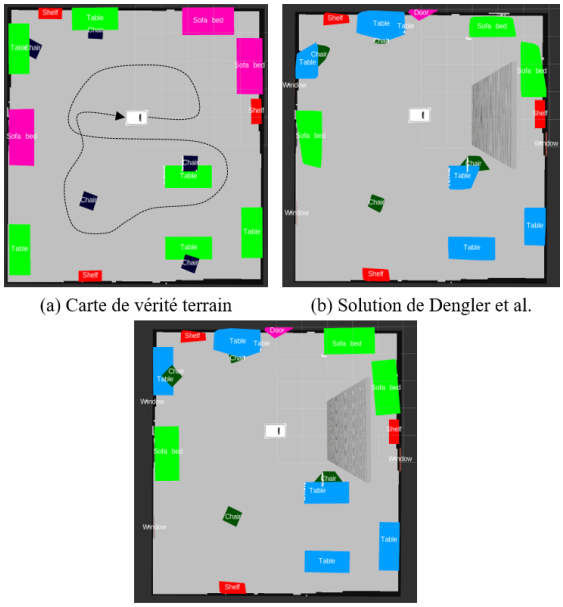
Définition de f_2 : Pour minimiser l'écart entre la longueur du polygone et celle de la boîte englobante, la valeur de $|\alpha_{min}| + \alpha_{max}$ doit tendre vers la longueur de la boîte englobante. La fonction f_2 est définie comme suit : $f_2 = (l - (|\alpha_{min}| + \alpha_{max}))/l$.

Définition de f_3 : Pour minimiser l'écart entre la largeur du polygone et celle de la boîte englobante, la valeur de β_{max} correspondant au sommet inférieur extrême du polygone doit tendre vers la largeur de la boîte englobante. La fonction f_3 est définie comme suit : $f_3 = (w - \beta_{max})/w$.

Le score $\mathcal{S}$ n'est calculé que pour les boîtes qui respectent deux critères : $|\alpha_i| + (1 - i) * \|e_i\| < l + \psi$ et $\beta_{max} < w + \psi$. ψ est une constante définie pour prendre en compte les erreurs de mise à l'échelle causées par les erreurs d'observation.

3 Expérimentation

Nous avons comparé les performances de notre approche de cartographie sémantique à celles de l'approche open source de Dengler *et al.* [3]. Cette dernière utilise des données RGBD et l'algorithme Quickhull pour produire une représentation polygonale des objets sémantiques. Le robot mobile MiR100, équipé d'une caméra Asus Xtion Pro, d'une résolution de 640x480 pixels, placée à une hauteur de $1m$ au-dessus du robot et inclinée à un angle de 5 degrés par rapport au sol, est utilisé pour la cartographie. Pour une comparaison appropriée, nous avons adopté le modèle de détection utilisé dans [3] sans entraînement supplémentaire et créé un environnement bureautique simulé similaire contenant les mêmes modèles d'objets. La base de connaissances créée contient les dimensions de quatre objets : Chaise, Table, Etagère et Canapé. Ainsi, l'environnement de test créé est d'environ $100m^2$ et contient différentes instances de ces objets, avec des zones espacées et d'autres où les objets sont partiellement cachés. Nous avons effectué 12 séquences de cartographie, où la trajectoire du robot varie légèrement d'une séquence à l'autre, afin de fournir une performance indépendante des points de vue des objets. Fig. 2.a montre une des trajectoires suivies par le robot. Les paramètres de la solution ont été déterminés à l'avance à l'aide de plusieurs expériences dans un environnement de validation. Les poids de la fonction $\mathcal{S}$ ont été fixés à $w_1 = 0.5$, $w_2 = 0.3$ et $w_3 = 0.2$. La constante ψ a été fixée à $\psi = 0,1m$, et le seuil de sélection a été fixé à $\epsilon = 0,6$. Les résultats présentés dans Tab. 1 ont été obtenus avec ces paramètres.



(a) Carte de vérité terrain (b) Solution de Dengler et al.

(c) Notre solution

FIGURE 2 : Visualisation des cartes (a) de vérité terrain, (b) de la méthode de Dengler *et al.* [3], et (c) de notre approche.

3.1 Évaluation de l'algorithme d'association

Les étapes de simplification des polygones et de sélection des arêtes de premier plan, présentées dans la section 2.4.2, réduisent significativement le nombre d'arêtes à associer, ce qui permet d'améliorer le temps de traitement. Par exemple, lors d'une séquence, le nombre total d'arêtes générées est passé de 12060 à 4137 après simplification et seulement 1369 arêtes de premier plan, soit une réduction de 10% du nombre d'arêtes initiales. Le temps de traitement de l'association est ainsi divisé par 10. En moyenne, le temps de traitement d'une étape d'association pour une séquence est d'environ 4 ms.

Les performances de notre approche sont comparées à celles de [3] en étudiant les indicateurs ci-après pour chaque catégorie d'objet sur 12 séquences. La similarité entre la forme créée d'un objet et sa boîte englobante de vérité terrain est évaluée en utilisant la valeur moyenne de l'intersection par rapport à l'union (IoU). Le déplacement des objets par rapport à la position de référence est évalué à l'aide de la valeur moyenne du déplacement du centre de masse (CoM). Ces deux indicateurs ne prennent en compte que les objets correctement cartographiés (TP), pour lesquels l'IoU est > 0,2. Tab. 1 montre les résultats moyens obtenus pour les 12 séquences, soit un total de 142 objets cartographiés en environ 24 minutes (2 minutes/séquence). Notre approche intégrant l'étape d'augmentation est plus performante que celle de Dengler *et al.* pour tous les objets, à l'exception de l'étagère, pour laquelle les résultats sont presque équivalents (Fig. 2). Notre solution montre de bonnes performances dans l'approximation des objets de grande taille, tels que les tables ou les canapés, avec de grandes parties non vues, ce qui conduit à des améliorations significatives à la fois de l'IoU moyen et du déplacement du CoM moyen. Nous constatons également une meilleure approximation de la forme des objets de premier plan, en particulier les chaises. Notre solution est également performante pour l'objet étagère, mais ses résultats sont relativement inférieurs à ceux des autres objets et similaires à [3]. Cela est dû au fait que, bien que l'orientation de la boîte ait été correctement estimée, le côté de décalage n'a pas été bien choisi dans certains cas. Comme l'objet est petit, ce décalage a une influence significative sur les valeurs de l'IoU et le déplacement du CoM. De plus, notre approche réduit systématiquement l'écart-type, ce qui la rend plus stable.

TABLE 1 : Résultats moyens des indicateurs de cartographie.

Indicateurs	**TP**	**Intersection sur l'union (IoU)**		**Déplacement du CoM (m)**	
Catégorie		**Notre solution**	**Dengler *et al.* [3]**	**Notre solution**	**Dengler *et al.* [3]**
Chaise	48	**0.8216 (0.04)**	0.6559 (0.07)	**0.0455 (0.02)**	0.0782 (0.02)
Table	58	**0.8825 (0.03)**	0.7521 (0.05)	**0.0672 (0.02)**	0.1799 (0.05)
Etagère	12	0.6477 (0.10)	**0.7044 (0.10)**	0.1030 (0.03)	**0.0914 (0.05)**
Canapé	24	**0.8241 (0.04)**	0.6709 (0.04)	**0.1078 (0.02)**	0.1770 (0.04)

4 Conclusion

Cet article présente une nouvelle méthode pour la localisation d'objets sémantiques en utilisant des connaissances préalables sur leurs dimensions. Les résultats obtenus dans un environnement bureautique, comparés à [3], ont montré une réduction significative de la complexité de la représentation polygonale et une amélioration considérable de l'approximation de presque tous les objets, en particulier les objets partiellement visibles et les objets occultés. La méthode sera testée sur d'autres jeux de données dans nos prochains travaux.

Références

[1] Abdessalem ACHOUR, Hiba AL-ASSAAD, Yohan DUPUIS et Madeleine EL ZAHER : Collaborative mobile robotics for semantic mapping : A survey. *Applied Sciences*, 12(20):10316, 2022.

[2] C Bradford BARBER, David P DOBKIN et Hannu HUHDANPAA : The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483, 1996.

[3] Nils DENGLER, Tobias ZAENKER, Francesco VERDOJA et Maren BENNEWITZ : Online object-oriented semantic mapping and map updating with modular representations. *CoRR*, abs/2011.06895, 2020.

[4] Antonin GUTTMAN : R-trees : A dynamic index structure for spatial searching. *In Proceedings of the ACM SIGMOD international conference on Management of data*, pages 47–57, 1984.

[5] Xianyu QI, Wei WANG, Mei YUAN, Yuliang WANG, Mingbo LI, Lin XUE et Yingpin SUN : Building semantic grid maps for domestic robot navigation. *International Journal of Advanced Robotic Systems*, 17(1), 2020.

[6] Kevin WEILER et Peter ATHERTON : Hidden surface removal using polygon area sorting. *ACM SIGGRAPH computer graphics*, 11(2):214–222, 1977.

[7] Tobias ZAENKER, Francesco VERDOJA et Ville KYRKI : Hypermap mapping framework and its application to autonomous semantic exploration. *In IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 133–139, 2020.