Reconstruction 3D légère et basse résolution en environnements intérieurs à partir de caméra RGB-D

Bruce CANOVAS, Michèle ROMBAUT, Amaury NÈGRE, Serge OLYMPIEFF, Denis PELLERIN

Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, 38000 Grenoble, France
{f_author, s_author}@gipsa-lab.grenoble-inp.fr

Résumé – Les approches de reconstruction 3D existantes se focalisent sur la production de modèles complexes avec un haut niveau de détail. Nous proposons au contraire une méthode basse-résolution, mais rapide et légère, pour la reconstruction 3D d'environnements intérieurs avec caméra RGB-D, sur plateforme de puissance et de capacité mémoire limitée. Le système présenté se base sur la segmentation d'images en superpixels, pour reconstruire un modèle 3D sous forme de patchs elliptiques, encodant la géométrie et la couleur locale d'une surface, appelés supersurfels.

Abstract – In this paper we present a low cost RGB-D mapping system based on a novel lightweight form of 3D representation. Our approach builds and updates a low resolution 3D model of an observed scene as an unordered set of new primitives called supersurfels, which can be seen as elliptical planar patches, generated from superpixels segmented RGB-D live measurement. While most of the actual solutions focuses on the accuracy of the reconstructed 3D model, the implemented method is well-adapted to run on robots with limited computing capacity and memory storage, which do not need a highly detailed map of their environment but can settle for an approximate one.

1 Introduction

La reconstruction 3D dense d'environnements observés au moyen d'un capteur RGB-D, permettant de capturer simultanément la carte de profondeur et la texture, est un sujet de recherche actif en vision par ordinateur pour la robotique. En effet, afin de pouvoir intéragir dans son mileu, un robot doit avoir accès en temps réel à sa géométrie.

En raison des formes de représentation 3D qu'elles utilisent pour modéliser l'environnement, de nombreuses approches état de l'art se limitent à la reconstruction de milieux de petite taille et/ou nécessitent du matériel lourd et coûteux pour fonctionner en temps réel. Elles emploient pour la plupart des formes de représentation permettant un haut niveau de détail, mais très coûteuses, alors que dans beaucoup de situations un tel niveau de précision n'est pas nécessaire.

Parmi les pincipales solutions de reconstruction 3D dense existantes, beaucoup s'appuient sur une représentation volumétrique, comme KinectFusion [1]. Celle-ci reconstruit un unique modèle 3D de haute qualité à partir de données RGB-D courantes. La représentation utilisée consiste en une fonction de distance signée discrétisée dans un volume (grille 3D régulière englobant la scène à reconstruire) subdivisé en éléments cubiques appelés voxels. Si les approches volumétriques sont robustes aux bruits et produisent des résultats très précis, elles requièrent un volume mémoire très important. La fonction de distance signée ne peut par ailleurs pas être visualisée directement via le biais d'un pipeline de rendu graphique standard et doit être convertie en une autre forme de représentation.

D'autres approches, comme ElasticFusion [2], représentent l'environnement sous forme d'ensemble non ordonné de surfels. Un surfel est un élément de surface circulaire encodant comme principaux attributs sa position, sa normale et son rayon. Les surfels sont facilement générés à partir d'images RGB-D et peuvent être directement affichés via un pipeline de rendu graphique. De plus, contrairement aux méthodes volumétriques, les espaces libres n'ont pas à être représentés, ce qui rend ces approches plus légères et efficaces en termes de gestion de la mémoire. Les méthodes basées sur les surfels restent néanmoins très coûteuses, un modèle 3D pouvant compter plusieurs millions de surfels.

Au lieu d'associer à chaque pixel d'une image un surfel, l'approche [2] propose de générer des surfels à partir de superpixels, découpant l'image en sous-ensembles homogènes. Le but est de réduire le nombre de primitives 3D utilisées pour représenter la scène, afin de gagner en efficacité et pouvoir reconstruire des environnements plus grands. Contrairement à notre méthode, leur système de localisation est indépendant de leur système de reconstruction 3D et leur méthode se limite à l'utilisation de superpixels réguliers et de taille réduite.

Dans cet article, nous présentons une méthode de reconstruction 3D basée sur une nouvelle forme de représentation. Notre contribution, SupersurfelFusion, est un système de cartographie 3D compact et temps réel, réalisant une reconstruction de l'environnement basse résolution et légère, sous forme de patchs elliptiques appelés supersufels, à partir du flux vidéo d'une caméra RGB-D en mouvement. La méthode proposée vise à permettre une reconstruction 3D rapide sur des plateformes de puissance réduite, ou pour des applications nécessitant un haut niveau d'efficacité.

2 Aperçu du système

Le système développé, SupersurfelFusion, génère un unique modèle $\mathcal{G}$ défini dans le repère global $\mathfrak{R}_W$, sous la forme d'un ensemble de primitives 3D nommées supersurfels (figure 1). Le modèle global $\mathcal{G}$ est complété et mis à jour pour chaque nouvelle image acquise par une caméra RGB-D, à partir de l'ensemble de supersurfels $\mathcal{F}$ associé à cette image.

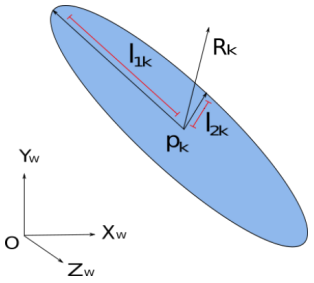

FIGURE 1 – Supersurfel défini dans le repère $\mathfrak{R}_W(O; X_w, Y_w, Z_w)$, de position $\boldsymbol{p_k}$, longueur $\boldsymbol{l_{1k}}$, largeur $\boldsymbol{l_{2k}}$ et d'orientation $\boldsymbol{R_k}$.

Un supersurfel peut être vu comme une approximation de la reprojection 3D d'un superpixel associé. Un superpixel définit un groupe de pixel connectés homogènes. Un supersurfel $\mathcal{G}^k \in \mathcal{G}$ est un patch elliptique de l'espace 3D, qui encode la position de son centre $\boldsymbol{p_k} \in \mathbb{R}^3$, son orientation $\boldsymbol{R_k} \in \mathbb{SO}^3$, sa couleur $\boldsymbol{c_k} \in \mathbb{R}^3$, sa largeur $\boldsymbol{l_{1k}} \in \mathbb{R}$ et sa longueur $\boldsymbol{l_{2k}} \in \mathbb{R}$, un marqueur temporel $\boldsymbol{t_k} \in \mathbb{N}$ qui représente le dernière instant où le supersurfel a été observé, un poids $\boldsymbol{w_k} \in \mathbb{R}_+$ pour quantifier la confiance que l'on a par rapport à la validité du supersurfel (état stable ou instable) et une matrice de covariance 3D $\boldsymbol{\Sigma_k} \in \mathcal{M}_3(\mathbb{R})$ décrivant sa forme (voir eq. 1).

Le système implémenté prend en entrée un flux d'images RGB-D. Une image RGB-D est définie par une paire $(\boldsymbol{C_t}, \boldsymbol{Z_t})$, avec $\boldsymbol{C_t} : \boldsymbol{\Omega} \rightarrow \mathbb{R}^3$ l'image couleur au temps $\boldsymbol{t}$, $\boldsymbol{Z_t} : \boldsymbol{\Omega} \rightarrow \mathbb{R}$ l'image de profondeur associée et $\boldsymbol{\Omega}$ le plan image. Le système procède ensuite en 3 étapes :

1. L'image RGB-D est segmentée en superpixels, qui sont utilisés pour générer l'ensemble de supersurfels $\mathcal{F}$ associé à la vue courante, dans le repère de la caméra $\mathfrak{R}_C$.
2. La pose de la caméra (position et orientation) est estimée dans l'espace global $\mathfrak{R}_W$.
3. Les supersurfels de l'ensemble $\mathcal{F}$ sont transformés du repère local $\mathfrak{R}_C$ au repère global $\mathfrak{R}_W$ et le modèle $\mathcal{G}$ est mis à jour.

3 Reconstruction basée supersurfels

3.1 Génération de supersurfels

Tout d'abord, la nouvelle paire RGB-D $(\boldsymbol{C_t}, \boldsymbol{Z_t})$ acquise est segmentée en N superpixels $C = \{\boldsymbol{C^h}, h = 1, ..., N\}$. Un superpixel est un groupe de pixels homogènes $\boldsymbol{u_j}$, pour $j = 1, 2, ..., M$ avec M la taille du superpixel, de couleurs similaires et dont les reprojections 3D se trouvent approximativement sur un même plan. Une implémentation GPU de l'approche décrite par [3] est appliquée pour partitionner l'image courante en superpixels adaptés, c'est à dire préservant autant que possible les contours et les discontinuités géométriques. Un critère de coplanarité est utilisé pour filtrer la profondeur $\boldsymbol{Z_t}$.

Un supersurfel $\boldsymbol{\mathcal{F}^i} \in \mathcal{F}$, où $\mathcal{F}$ désigne l'ensemble des supersurfels associés à la vue courante $(\boldsymbol{C_t}, \boldsymbol{Z_t})$, est généré pour chaque superpixel $\boldsymbol{C^h}$. La position $\boldsymbol{p_i}$ du supersurfel $\boldsymbol{\mathcal{F}^i}$ est la moyenne des reprojections 3D $\pi(\boldsymbol{u_j}, \boldsymbol{Z_t(u_j)})$ des pixels contenus dans $\boldsymbol{C^h}$. De même, la couleur $\boldsymbol{c_i}$ est définie comme la couleur moyenne des pixels du superpixel. La matrice de covariance $\boldsymbol{\Sigma_i}$ est calculée à partir de la formule suivante :

$$\Sigma_i = \frac{1}{M-1} \sum_{j=1}^{M} (\pi(u_j, Z_t(u_j)) - p_i)(\pi(u_j, Z_t(u_j)) - p_i)^T. \quad (1)$$

La décomposition en valeurs propres de la matrice de covariance $\boldsymbol{\Sigma_i}$ permet de déterminer l'orientation $\boldsymbol{R_i}$, correspondant à la matrice formée par les vecteurs propres $\boldsymbol{e_{1i}}, \boldsymbol{e_{2i}}, \boldsymbol{e_{3i}}$ de $\boldsymbol{\Sigma_i}$, avec pour vecteur normal celui associé à la plus petite des valeurs propres $\boldsymbol{\lambda_{1i}}, \boldsymbol{\lambda_{2i}}, \boldsymbol{\lambda_{3i}}$ ($\boldsymbol{\lambda_{1i}} > \boldsymbol{\lambda_{2i}} > \boldsymbol{\lambda_{3i}}$). La longueur et la largeur $\boldsymbol{l_{1i}}, \boldsymbol{l_{2i}}$ du supersurfel sont calculées à partir de $\boldsymbol{\lambda_{1i}}$ et $\boldsymbol{\lambda_{2i}}$ ($\boldsymbol{l_{xi}} = 2.4477\sqrt{\boldsymbol{\lambda_{xi}}}$, $x = 1$ ou 2). Le temps courant $\boldsymbol{t}$ est assigné à l'horodatage $\boldsymbol{t_i}$. Enfin, le poids de confiance $\boldsymbol{w_i}$ est initialisé avec une valeur basse, correspondant au ratio entre le nombre de pixels ayant une profondeur valide (comprise dans le champ de mesure du capteur) dans le superpixel $\boldsymbol{C^h}$ et le nombre total de pixels qu'il contient : $\boldsymbol{w_i} \leftarrow pixels_{valid}/pixels_{total}$.

3.2 Mise à jour du modèle

SupersurfelFusion génère et complète un unique modèle 3D $\mathcal{G}$ sous la forme d'un ensemble de supersurfels $\mathcal{G}^k$ stockés dans un tableau indexé par $\boldsymbol{k} \in \mathbb{N}$. Les nouveaux supersurfels générés, de l'ensemble $\mathcal{F}$, sont soit directement ajoutés dans le modèle, soit combinés avec des supersurfels du modèle, pour mettre à jour leurs attributs. Un supersurfel $\mathcal{G}^k$, fusionné avec un supersufel $\boldsymbol{\mathcal{F}^i} \in \mathcal{F}$ verra sa valeur de confiance $\boldsymbol{w_k}$ croître. Un supersurfel du modèle peut passer de l'état instable à l'état stable si il est observé de manière répétée. Les supersurfels vérifiant $\boldsymbol{w_k} > \boldsymbol{w_{stable}}$, avec $\boldsymbol{w_{stable}}$ un seuil fixé, sont considérés comme stables.

La fusion des supersurfels de la vue courante $\mathcal{F}$ avec les supersurfels similaires du modèle $\mathcal{G}$ a pour but d'affiner le modèle et de réduire les redondances. La première étape consiste à chercher pour chaque nouveau supersurfel $\boldsymbol{\mathcal{F}^i}$, si un supersurfel ressemblant $\mathcal{G}^k$ est présent dans le modèle. Connaissant la pose de la caméra dans le repère global, les supersurfels $\boldsymbol{\mathcal{F}^i}$ de $\mathcal{F}$ sont alignés avec le modèle. On réalise ensuite pour chaque $\boldsymbol{\mathcal{F}^i}$ une recherche du plus proche voisin dans $\mathcal{G}$, à travers une hiérarchie de volumes englobants. La hiérarchie de volumes englobants est une structure accélératrice permettant d'organiser la scène reconstruite sous la forme d'un arbre de recherche bi-

naire. La divergence de Kullback-Leibler symmétrisée est utilisée comme distance afin de déterminer le supersurfel $\mathcal{G}^k$ qui ressemble le plus à $\mathcal{F}^i$:

$$KLD(\mathcal{F}^i||\mathcal{G}^k) = \frac{1}{2}\{tr(\Sigma_i^{-1}\Sigma_k + \Sigma_k^{-1}\Sigma_i) + (p_i - p_k)^T(\Sigma_i^{-1} + \Sigma_k^{-1})(p_i - p_k)\} - 3. \quad (2)$$

Elle permet de mesurer la dissimilarité entre deux supersurfels en termes de position, de forme et d'orientation. La couleur des supersurfels et l'angle entre les normales sont aussi pris en compte pour rejeter de mauvaises correspondances.

Si aucun supersurfel correspondant $\mathcal{G}^k$ n'est trouvé, le supersurfel $\mathcal{F}^i$ est simplement ajouté dans le modèle. Sinon, pour chaque paire de correspondances $(\mathcal{G}^k, \mathcal{F}^i)$ trouvée, les attributs de $\mathcal{G}^k$ sont mis à jour à partir des données de $\mathcal{F}^i$.

Les nouvelles positions et covariances p'_k et Σ'_k sont calculées en appliquant la stratégie d'intersection de covariances :

$$\Sigma_k'^{-1} = \alpha\Sigma_k^{-1} + (1-\alpha)\Sigma_i^{-1}, \quad (3)$$

$$p'_k = \Sigma'_k(\alpha\Sigma_k^{-1}p_k + (1-\alpha)\Sigma_i^{-1}p_i), \quad (4)$$

$$\alpha = \frac{w_k}{w_k + w_i}. \quad (5)$$

La couleur mise à jour c'_k est obtenue par simple moyenne pondérée :

$$c'_k = \frac{w_k c_k + w_i c_i}{w_k + w_i}. \quad (6)$$

On applique la même procédure que celle utilisée lors de l'étape de génération des suspersurfels pour calculer les valeurs mises à jours de la longueur l'_{1k}, de la largeur l'_{2k} et de l'orientation R'_k. La valeur de confiance w'_k est incrémentée et l'horodatage est mis à jour avec la valeur du temps courant t.

Enfin, deux stratégies sont appliquées pour supprimer les aberrations du modèle. Premièrement, les supersurfels qui restent dans un état instable pendant trop longtemps sont supprimés après une période Δt.

Deuxièmement, les supersurfels du modèle qui se trouvent devant les supersurfels nouvellement mis à jour, par rapport à la caméra, sont détectés et supprimés. Ils représentent des violations de l'espace libre et correspondent probablement à des éléments en mouvement.

4 Estimation de la pose de la caméra

Soit $\mathbf{T} \in \mathbb{SE}3$ une transformation rigide, composée d'une matrice de rotation $\mathbf{R}$ et d'une translation $\mathbf{t}$. On note $\mathbf{T}(q) = \mathbf{R}q + \mathbf{t}$ l'application de la transformation à un point q. Pour estimer la transformation relative $\mathbf{T_{t,t-1}}$ de la caméra entre deux acquisitions (C_t, Z_t) et (C_{t-1}, Z_{t-1}), et mettre à jour sa pose $\mathbf{P}$ dans le repère global $\mathfrak{R}_W$, nous procédons en deux étapes suivant une approche similaire à [4].

4.1 Estimation initiale

La transformation relative $\mathbf{T_{t,t-1}}$ est calculée à partir des images courante et précédente. Des points d'intérêts sont détectés et appariés dans les images de couleur C_t, C_{t-1}. La transformation est alors estimée par minimisation de la distance entre les reprojecions 3D q_i^t, q_i^{t-1} des points d'intérêts appariés, issus respectivement de C_t et C_{t-1} :

$$\mathrm{T}_{\mathrm{t,t-1}} = \underset{\mathrm{T}}{argmin} \sum_i |\mathrm{T}(q_i^t) - q_i^{t-1}|^2. \quad (7)$$

Une estimation grossière de la nouvelle pose de la caméra dans le repère global est alors calculée par concaténation : $\mathbf{P} \leftarrow \mathbf{PT_{t,t-1}}$

4.2 Amélioration de l'estimation

Afin d'obtenir une estimation plus précise de la pose de la caméra, l'estimation initiale est passée en paramètre d'entrée d'un algorithme de type Iterative Closest Point. Cet algorithme calcule de manière itérative la transformation en minimisant la distance de Mahalanobis entre les centres des supersurfels $\mathcal{G}^k$ du modèle $\mathcal{G}$ et les supersurfels $\mathcal{F}^i$ de la vue courante $\mathcal{F}$:

$$\mathrm{P} = \underset{\mathrm{T}}{argmin} \sum_{\mathcal{A}_{i,k}} (p_k - \mathrm{T}(p_i))\Sigma_k^{-1}(p_k - \mathrm{T}(p_i)). \quad (8)$$

$\mathcal{A}_{i,k} = \{(i,k)_{1:L}\}$ est le jeu d'associations entre les supersurfels de $\mathcal{G}$ et $\mathcal{F}$ mis en correspondance. La couleur des supersurfels est utilisée pour contraindre la recherche de correspondances.

5 Résultats

Les tests ont été réalisées avec un ordinateur portable équipé d'une carte graphique Nvidia GTX 950M et d'un processeur Intel Core i5-6300HQ. Les supersurfels sont affichés sous forme de patchs rectangulaires pour simplifier la visualisation et des superpixels d'environ 16x16 pixels sont utilisés.

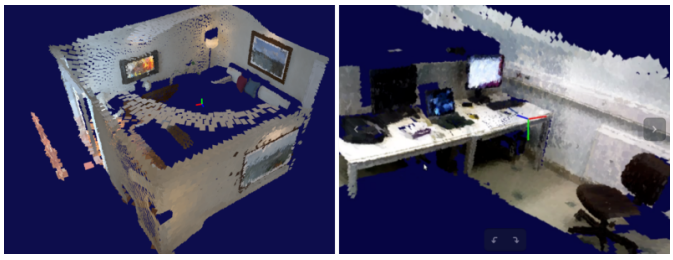

FIGURE 2 – Reconstruction 3D de l'environnement sous forme de supersurfels.

Les modèles 3D obtenus sur la figure 2 sont denses, même s'ils sont composés d'un nombre réduit de primitives. Certains éléments fins ou incurvés sont correctement modélisés (pieds de table, dossier de chaise...), soulignant la capacité de notre représentation, bien qu'approximative, à préserver l'information importante.

5.1 Performance

Pour évaluer les performances de notre système en termes de vitesse et de gestion de la mémoire, et montrer le gain d'efficacité qu'il permet par rapport aux solutions existantes, nous l'avons comparé à l'approche état de l'art ElasticFusion [2]. Des vidéos du TUM RGB-D benchmark [5] ont été utilisées (*fr1/room, fr1/plant, fr2/rpy, fr2/desk*). Les mesures de vérité terrain des trajectoires données avec chacune des vidéos ont été fournies aux deux systèmes afin de pouvoir comparer de manière adaptée les 2 méthodes de reconstruction 3D (sans prendre en compte le suivi de la caméra).

TABLE 1 – Temps d'exécution moyen (ms).

System	fr1/room	fr1/plant	fr2/rpy	fr2/desk
SFusion	23.7	23.8	21.2	22.2
EFusion	58.1	53.0	50.4	73.1

TABLE 2 – Empreinte mémoire maximale du modèle 3D (MB).

System	fr1/room	fr1/plant	fr2/rpy	fr2/desk
SFusion	5.23	4.41	1.45	2.37
EFusion	52.3	33.0	19.1	58.6

Les tableaux 1 et 2 présentent les temps d'exécution des reconstruction 3D et la taille mémoire maximale utilisée pour stocker le modèle. La processus de reconstruction 3D de SupersurfelFusion (SFusion dans le tableau) est environ 2 fois plus rapide et 10 fois plus léger que celui de ElasticFusion (EFusion dans le tableau) sur notre plateforme de tests. Ces résultats correspondent à nos attentes puisque ElasticFusion est une approche beaucoup plus précise, qui génère une primitive 3D pour chaque pixel d'une image RGB-D. Ils marquent aussi l'efficacité de la méthode.

5.2 Estimation de la trajectoire

L'odométrie visuelle de SupersurfelFusion a aussi été évaluée à partir de séquences vidéos du TUM. Les trajectoires estimées par notre système ont été comparées aux vérités terrains fournies par le jeu de données (tableau 3).

TABLE 3 – Erreur quadratique moyenne de translation entre la trajectoire estimée et la vérité terrain (mm).

fr1/xyz	fr1/plant	fr2/rpy	fr3/office
70.0	108.0	34.0	471.0

Si notre système est capable de correctement se localiser dans de nombreuses situations, il est amené à échouer lorsque les mouvements de la caméra sont trop brusques ou que la scène observée est trop pauvre en termes de géométrie. Par ailleurs, le suivi de la caméra n'est pour le moment efficace qu'à un niveau local. L'ajout d'une fermeture de boucle pourrait permettre de corriger l'erreur importante sur la séquence fr3/office.

6 Conclusion

Une nouvelle forme de représentation basse résolution, pour la reconstruction 3D en environnement intérieur statique, a été présentée dans cet article. Le modèle 3D, est défini par un ensemble supersurfels, générés à partir de la segmentation en superpixels d'images RGB-D. L'utilisation de superpixels, préservant l'information importante, garantit la fiabilité de la représentation approximative de l'environnement.

Le système de reconstruction 3D proposé, SupersurfelFusion, basé sur cette représentation, effectue une cartographie temps réelle grossière mais pertinente d'une scène observée, en utilisant peu de mémoire. L'utilisation de ce système est particulièrement intéressante pour des robots n'ayant pas besoin d'une reconstruction très précise de leur milieu, mais plutôt d'une reconstruction efficace, légère et rapide. Nous pensons que la faible quantitié de supersurfels requis pour modéliser un environnement peut rendre notre méthode adéquate pour de l'évitement accéléré d'obstacles.

Par la suite nous souhaiterions ajouter au système une fermeture de boucle permettant de corriger les éventuelles dérives dans l'estimation de la position de la caméra ainsi que dans la géométrie du modèle reconstruit. Il est aussi envisagé de rendre le système robuste aux milieux dynamiques en détectant les éléments en mouvement au niveau des superpixels.

Références

[1] S. Izadi, D. Kim et O. Hilliges et al., *KinectFusion : Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera*. Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technologyy, 2011.

[2] T. Whelan, S. Leutenegger et R. Salas Moreno et al., *ElasticFusion : Dense SLAM Without A Pose Graph*. Proceedings of Robotics : Science and Systems, 2015.

[3] K. Yamaguchi and D. McAllester and R. Urtasun et al., *Efficient Joint Segmentation, Occlusion Labeling, Stereo and Flow Estimation*. ECCV, 2010.

[4] P. Henry, M. Krainin, E. Herbst et al., *RGB-Dmapping : Using depth cameras for dense 3D modeling ofindoor environments*. Proceedings of the InternationalSymposium on Experimental Robotics (ISER), 2014.

[5] J. Sturm and N. Engelhard and F. Endres et al., *A Benchmark for the Evaluation of RGB-D SLAM Systems*. Proc. of the International Conference on Intelligent Robot Systems (IROS), 2012.

[6] K. Wang and F. Gao and S. Shen, *Real-time Scalable Dense Surfel Mapping*. Proc. of the International Conference on Robotics and Automation (ICRA), 2019.