

Approche sémantique de segmentation et de recherche interactive par le contenu issu d'une caméra de profondeur

Elliot Vanegue

2 août 2016

Table des matières

1	Introduction	3
1.1	Contexte	3
1.2	Sujet	3
1.3	Problématique	4
1.4	Organisation	4
2	Etat de l'art	5
2.1	Segmentation	5
2.1.1	Scène intérieur	5
2.1.2	Corps humain	6
2.2	Reconnaissance d'objets	6
2.2.1	Descripteur	6
2.2.2	Apprentissage automatique	8
2.2.3	Bag of word	8
2.3	Positionnement de modèle	8
3	Modification des membres du corps humain	8
3.1	Objectif	8
3.2	Délimitation du corps humain	8
3.3	Calcule de la distance géodésique	8
3.4	Calcule de descripteurs	8
3.5	SDK de la Kinect	8
3.6	Positionnement	8
4	Reconstruction d'un environnement intérieur	8
4.1	Objectif	8
4.2	SVM	8
4.3	Descripteur	8
5	Conclusion	8

1 Introduction

1.1 Contexte

Durant notre master IVI¹, nous avons l'occasion de réaliser un stage de fin d'étude. J'ai choisi de réaliser ce stage dans le laboratoire 3D-SAM spécialisé dans l'acquisition et le traitement d'image 3D à partir de capteur 3D de type Microsoft Kinect. Leur principaux travaux porte sur l'analyse de forme d'objet 3D et la modélisation des variation des formes dans des vidéo 3D. Réaliser mon stage dans un laboratoire était une priorité, car mes projets d'avenir ne sont pas encore tout à fait planifié et ce stage me permet de me décider sur l'environnement de travail qui me convient le mieux. Durant mes deux derniers stage en IUT et en licence 3, j'ai pu observer les atouts et les contraintes de chaque environnement, cependant le stage que j'ai effectué durant mon année en IUT relevé plus de l'ingénierie que de la recherche. Durant ce stage de deuxième année d'étude je n'ai pas eu à effectuer un état de l'art, ni à tester des méthodes proposé par d'autre laboratoire de recherche.

Ces deux années de master nous ont beaucoup appris sur le traitement d'image et la reconnaissance de forme 2D. Ces sujets mon particulièrement intéressé et je souhaite, durant ce stage, approfondir ces notions sur des types de donnée plus complet comme sur des images 3D. C'est pourquoi je réalise mon stage dans l'équipe 3D-Sam avec qui j'avais déjà travaillé sur mon projet de fin d'étude. Cette équipe dirigé par M. Daoudi comprend cinq membre permanent, un post-doctorant et quatre doctorant. Mes encadrant durant ce stage sont M. Vandeborre et M. Wannous. Ce stage ce déroule dans le cadre du projet CrABEx qui concerne la production et l'édition de produit 3D pour des applications de type loisir. L'enjeu de ce projet est d'aider les designer 3D dans la création et l'édition de ressources graphiques en suggérant des éléments approprié durant leur processus de création ou en générant automatiquement de nouvelles ressources à partir d'éléments existant.

1.2 Sujet

Comme nous pouvons le constater depuis quelques années, les approches 3D interactive sont de plus en plus présentent dans nos vies, que ce soit dans le domaine de la médecine avec l'utilisation de simulateur pour apprendre à réaliser des opérations complexe, dans le loisir avec les jeux vidéo dont le revenu mondial en 2015 est de plus de 90 milliard de dollars ou encore dans l'industrie pour visualiser des produit. L'engouement pour cette technologie requiert des outils de plus en plus efficace et rapide permettant à des personnes de profession plutôt artistique de laisser place à leur imagination sans s'inguiéter de l'aspect technique.

Le fait de pourvoir modéliser dans un monde virtuelle des objects du monde réel, et de pouvoir modifier ces objets facilement permettrait au designer de se soustraire de certaines tâches. On peut par exemple modéliser une scène ou une personne facilement grâce à une caméra Kinect, afin d'effectuer des modifications. Il serait intéressant de récupérer le modèle 3D d'une personne et de modifier quelques unes des parties de son corps avec des membre improbable comme un bras de robot. Le principe est le même dans une pièce, si nous souhaitons remplacer des meubles d'une pièce intérieur par d'autre plus improbable. L'intérêt de cela est d'éviter à un designer de devoir remodeliser une personne ou une pièce existante qui serait déjà idéal pour ce que l'on

1. Le master Image Vision Interaction est une spécialité du master informatique de l'université de Lille 1

souhaite réaliser.

Ce genre de technologie existe déjà pour modéliser des visages ou des gestes dans des jeux vidéo ou des films d'animation. Il y a par exemple des capture optique basé sur des caméras infrarouge avec des marqueurs réfléchissant. La Kinect fait également partie des outils de capture de mouvement, mais elle reste très peu utilisé dans le milieu professionnel.

1.3 Problématique

Les caméras 3D nous permettent d'obtenir un nuage de point de l'environnement qu'elles enregistrent. Cependant, ce type de caméra est peu précise et elles sont souvent bruité et comporte des valeurs qui n'existe pas dans le monde réel. Ce bruit dépend de plusieurs facteur comme la luminosité de l'environnement dans lequel nous effectuons l'acquisition ou tout simplement la qualité et la précisions du capteur utilisé. La première difficulté lors de ce projet est de réussir à filtrer les données de sorte à ce qu'il ne reste que les données réellement présente. L'ensemble des données ne nous intéresse pas forcément. En effet, si nous travaillons sur le corps humain nous n'avons pas besoin de l'ensemble de la pièce intérieure, ce surplus de données nous dérange lors de nos traitements. Il faut donc trouver une solution permettant de segmenter les données que nous recevons afin de ne garder que certains objets.

Une fois que les données ont été trié, nous avons besoin de reconnaître les objets présent dans notre scène. Par exemple, dans le cas du corps humain, si nous souhaitons modifier un membre comme la main il faut d'abord savoir quel membre représente la main. La reconnaissance d'objet dépend fortement du type de donnée et nécessite généralement d'avoir une base d'apprentissage assez volumineuse.

Lors de la modification d'une partie de la scène, il est nécessaire de connaître la position de l'objet remplacé, afin de pouvoir placer le nouvel objet au même endroit et dans le même sens. Cette partie est très délicate surtout pour des membres humain, car si le repositionnement n'est pas parfait est qu'il y a un écart entre deux membres, la scène perd toute sa crédibilité et son réalisme.

Nous avons donc trois problématique majeur dans ce projet. Comment filtrer les données de manière à ne traiter que celles qui nous intéresse? Comment reconnaître un objet dans une scène 3D? Comment connaître la position exacte des objets présents dans notre scène?

1.4 Organisation

Pour répondre aux questions précédentes, je vais travailler sur deux applications mettant en avant des problématique similaire, mais avec deux façon différentes d'aborder ces problèmes. Je vais travailler dans un premier temps sur une application qui se focalise sur le corps humain. Cette application aura pour but d'identifier les membres du corps afin que l'utilisateur puisse les modifier grâce à des modèles 3D existant. Les seuls interactions de l'utilisateur sont donc de cliquer sur le membre à modifier et de sélectionner un modèle parmi ceux que l'application lui proposera. La seconde application se concentre sur une pièce intérieure comportant plusieurs objets. Ici le but est de segmenter la pièce afin de reconnaître les objets présents afin de les ajouter dans une scène. Encore une fois, l'utilisateur doit sélectionner un meuble et sélectionner le modèle 3D qui lui convient dans une liste proposé par l'application.

Pour ce projet j'utilise la caméra Microsoft Kinect v2, qui est l'une des caméras les plus utilisé dans la littérature. Le SDK fournit avec cet outil comporte une segmentation du corps humain, la position du squelette de l'utilisateur et l'algorithme Kinect fusion qui permet de

construire un modèle 3D à partir du nuage de point fournit par la caméra.

Pour réaliser les deux application, je vais utiliser un ensemble de bibliothèque tel que la bibliothèque graphique de Microsoft pour la réalisation de l'interface, la bibliothèque PCL²[1] pour les calculs sur les nuages de points et opencv pour travailler sur les images couleurs et les images de profondeur fournit par la Kinect.

2 Etat de l'art

2.1 Segmentation

La première étape lors de notre projet va être de segmenter les images que nous recevons de notre caméra. Les informations contenues dans une image 3D sont nombreuses et nous devons déterminer les éléments important pour nos traitements. Dans notre scène, nous avons besoin des objets proches ou du corps de la personne en face de la caméra, mais l'environnement autour des ces objets clés n'est pas important et doit être supprimer pour gagner du temps lors de nos traitements en supprimant de l'information à traiter. Une second segmentation est nécessaire pour le traitement du corps humain. Pour cette étape du projet, nous devons segmenter le corps en plusieurs partie pour pouvoir par la suite les reconnaitres. Si cette seconde segmentation n'est pas réalisé il ne nous sera pas possible de reconnaître les mains ou encore la tête si nous ne savons délimité les parties du corps.



FIGURE 1 – Exemple de segmentation recherché pour une pièce intérieur et pour le corps humain

2.1.1 Scène intérieur

De nombreux travaux ont été réalisé dans la segmentation d'image 2D avant que les caméras 3D ne soit ouvert au grand public. Les premières méthode de segmentation reposaient sur la détection de contour comme pour la méthode de P. Arbelaez et al[2]. Leur méthode repose sur le détecteur de contour gPb qui est composé de d'un seuillage sur la luminance et sur la couleur et d'une détection de texture. La fermeture des contours se fait ensuite en utilisant les superpixels. D'autres méthodes 2D utilise un simple seuillage en utilisant par exemple la méthode de N. Otsu[3] pour binariser l'image et ainsi la segmenter.

Avec l'arrivé des caméras 3D de nombreuses recherche ont été effectué sur la segmentation d'image à partir des information extraite de ce type de caméra. S.A.A Shah et al[4] utilisent les

2. Point Cloud Librairie

informations de l'image de profondeur afin de calculer un vecteur sur chaque pixel. En appliquant un seuillage sur la différence des vecteurs ils obtiennent une segmentation de l'environnement qui leur permet de détecter des objets dans une pièce. Il est possible à partir de l'image de profondeur de créer un nuage de point, ce qui permet d'obtenir les coordonnées 3D des points présents dans l'image de profondeur. Les informations qu'il est possible d'extraire d'un nuage de point sont différentes et des méthodes de segmentation se sont développé autour de ces informations.

T. Rabbani et al[5] utilise les informations obtenus dans un nuage de point afin de calculer les normales de chaque point. Ils segmente ensuite l'image en comparant les normales et en appliquant un seuillage sur cette comparaison. Si l'angle formé par les normales de deux points est super au seuil alors les points appartiennent à deux régions différentes.

Nous pouvons voir que les méthodes cité précédemment sont efficace pour segmenter une scène comportant des objets, mais elles ne sont pas applicable à un corps humain. Le principal défaut de ces méthodes pour le corps humain est que celui-ci est trop lisse. La différence entre les normales ou entre les vecteurs de pixel n'est pas assez important et est trop instable pour que cela marche sur le corps humain.

2.1.2 Corps humain

La segmentation du corps humain est un sujet très complexe, car contrairement au objet celui-ci bouge et adopte des postures différentes. La méthode la plus souvent utilisé pour résoudre cette problématique est de déterminer la posture de l'utilisateur et lorsque cette posture est connu il est facile de déterminer les différentes partie du corps. Ces méthodes nécessitent d'avoir une base de connaissance contenant de nombreuses postures qui doivent être segmenter et labelisé avec les différentes parties du corps. J. Shotton et al[6] ont d'abord créé une base d'apprentissage en calculant un descripteur³ et une technique d'apprentissage automatique appelé forêt d'arbres décisionnels[7]. Lorsque l'utilisateur bouge, le descripteur utilisé précédemment est recalculé sur l'image courante et le résultat est comparé au posture de la base d'apprentissage. La posture ayant une valeur proche du résultat calculé précédemment est la posture de l'utilisateur. De nombreuses méthode utilise cette solution, mais elle modifie le descripteur utilisé et la technique d'apprentissage automatique comme nous le verrons dans la suite de ce rapport.

2.2 Reconnaissance d'objets

La reconnaissance d'objet est un sujet assez vaste dans le monde de l'imagerie et il existe de nombreuses méthodes dans le domaine que ce soit pour des images 2D ou 3D. Dans le cas d'image 3D la méthode la plus utilisé est le calcul de descripteur, ce qui correspond à un ensemble de caractéristiques représentant un objet spécifique.

2.2.1 Descripteur

Le nombre de descripteur qui existe dans le domaine de l'image 3D est assez important c'est pourquoi pour ce rapport, nous allons nous contenter de décrire seulement les plus utilisées. Le descripteur D2[8] est un des outils de comparaison de forme 3D les plus simple à réaliser. Il se repose sur le calcul de la distance euclidienne entre chaque point du modèle 3D. L'ensemble de ces distances permet de créer un histogramme 1D et de comparer ces histogramme afin de reconnaître un objet. Ce descripteur fournit de bon résultat lorsque les objets à reconnaître sont très différents.

3. Voir 2.2.1

Le descripteur PFH[9] (Point Feature Histograms) est un outil permettant de calculer la courbure moyenne d'un voisinage de point en utilisant un histogramme multi-dimensionnel. Le calcul de la courbure et le fait que ce soit une généralisation permet d'être invariant en translation et en rotation, et permet également d'être moins sensible au bruit présent dans le nuage de point. Le voisinage dépend de la distance des points avec le point centrale et il ne peut excéder un certain nombre de voisin (voir Fig. 2).

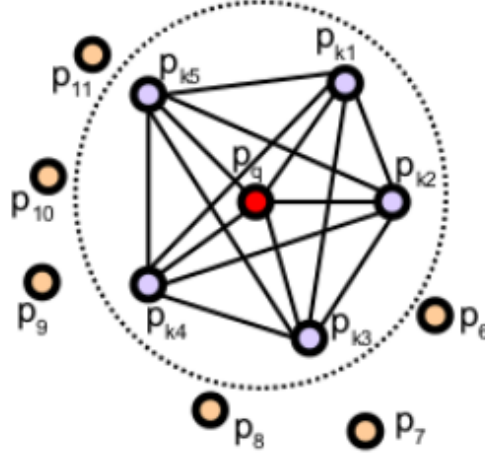


FIGURE 2 – Exemple de voisinage pris en compte dans le calcul de la courbure du descripteur PFH

Le descripteur PFH calculé en un point correspond à la relation que ce point a avec l'ensemble des points de son voisinage. Cette relation est la différence des normals entre deux points. Chaque classe de l'histogramme est composé de l'ensemble des points du voisinage dont la relation avec le point centrale est similaire. Une version améliorée du descripteur a été proposée par R. B. Rusu et al[10] appelé FPFH (Fast Point Feature Histograms). Cette version est plus rapide, car elle calcule un descripteur PFH simplifié, puis elle construit de nouveaux histogrammes à partir des histogramme simplifié précédent (voir Fig. 3).

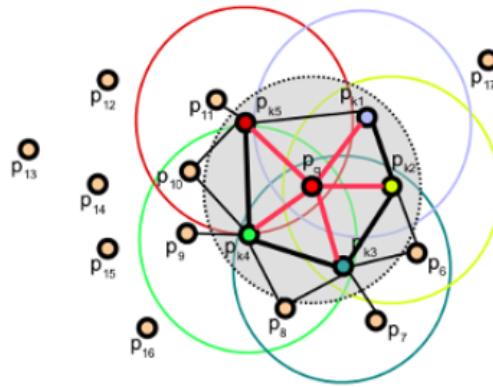


FIGURE 3 – Exemple de voisinage pris en compte dans le calcul de la courbure du descripteur FPFH

2.2.2 Apprentissage automatique

2.2.3 Bag of word

2.3 Positionnement de modèle

3 Modification des membres du corps humain

3.1 Objectif

3.2 Délimitation du corps humain

3.3 Calcule de la distance géodésique

3.4 Calcule de descripteurs

3.5 SDK de la Kinect

3.6 Positionnement

4 Reconstruction d'un environnement intérieur

4.1 Objectif

4.2 SVM

4.3 Descripteur

5 Conclusion

Références

- [1] A. Aldoma, Z.-C. Marton, F. Tombari, W. Wohlking, C. Potthast, B. Zeisl, R. B. Rusu, S. Gedikli, and M. Vincze, “Point cloud library,” *IEEE Robotics & Automation Magazine*, vol. 1070, no. 9932/12, 2012.
- [2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, “Contour detection and hierarchical image segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, pp. 898–916, May 2011.
- [3] N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, pp. 62–66, Jan 1979.
- [4] S. A. A. Shah, M. Bennamoun, and F. Boussaid, “A novel algorithm for efficient depth segmentation using low resolution (kinect) images,” in *Industrial Electronics and Applications (ICIEA), 2015 IEEE 10th Conference on*, pp. 603–607, June 2015.
- [5] T. Rabbani, F. Van Den Heuvel, and G. Vosselmann, “Segmentation of point clouds using smoothness constraint,” *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, no. 5, pp. 248–253, 2006.
- [6] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore, “Real-time human pose recognition in parts from single depth images,” *Communications of the ACM*, vol. 56, no. 1, pp. 116–124, 2013.
- [7] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [8] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, “Matching 3d models with shape distributions,” in *Shape Modeling and Applications, SMI 2001 International Conference on*, pp. 154–166, IEEE, 2001.
- [9] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz, *Persistent point feature histograms for 3D point clouds*. 2008.
- [10] R. B. Rusu, N. Blodow, and M. Beetz, “Fast point feature histograms (fpfh) for 3d registration,” in *Robotics and Automation, 2009. ICRA’09. IEEE International Conference on*, pp. 3212–3217, IEEE, 2009.