

---

# INF555 – PROJET

## Sketch-Based Shape Retrieval

---

L’article étudié présente une méthode permettant de retrouver un modèle en trois dimensions (dans une base de données de 1800 objets) à partir d’un croquis.

### 1 Méthode

La première étape consiste en la génération de croquis, à partir d’un modèle, tel qu’un humain le ferait.

Ensuite, plutôt que de comparer des images entières, l’article préfère travailler avec des features correspond à des morceaux des images.

On découpe les croquis générés en morceaux, ce qui fait un ensemble de plusieurs milliers de features, qu’on réduit à un millier de représentants. Ces 1000 petites images vont constituer notre alphabet.

Enfin on représente chaque croquis par un histogramme correspond au nombre de fois que chaque mot de l’alphabet apparaît dans celle-ci. Cet histogramme est une sorte de carte d’identité de l’image.

Pour trouver les croquis générés (et donc les modèles) les plus proches d’un dessin, on compare simplement les histogrammes. Des histogrammes proches indiquent des dessins proches.

### 2 Implémentation

L’algorithme, que nous avons choisi d’implémenter en **C++**, se découpe en deux grandes étapes :

- Un pré-traitement des 1800 modèles est d’abord effectué, permettant d’obtenir une nouvelle base de données, composé de l’alphabet de 1000 mots et des histogrammes des croquis générés ;
- La recherche à partir d’un croquis dans cette nouvelle base de données.



FIGURE 1 – Une orientation a été choisie, et seule certaines lignes sont dessinées

## 2.1 Pré-traitement des modèles

Lorsqu'une personne réalise un croquis (en deux dimensions) d'un objet, elle effectue plusieurs choix (figure 1) :

- Elle choisit une direction depuis laquelle regarder, ou selon laquelle projeter l'objet ;
- Elle choisit de quelle façon transformer l'objet texturé qu'elle observe en simple lignes.

Pour la première étape, pour chaque modèle 3D on génère un ensemble de 50 directions au hasard (mais uniformément réparties) selon lesquelles projeter. Comme suggéré par l'article on tire 50 points aléatoirement sur une sphère (provenant d'un fichier .off composé de 16 000 sommets) puis on applique l'algorithme k-means permettant de séparer la sphère en 50 régions de même taille uniformément réparties. On prend ensuite le centre de chaque région.

Pour la deuxième étape, le choix que nous avons fait est de calculer une carte de profondeur de la projection (une image de l'objet où la couleur d'un point dépend de sa distance à l'observateur : un point proche sera noir et un point au loin sera blanc), puis d'appliquer à l'image obtenue un algorithme de détection des contours (Canny). Nous ouvrons le modèle au format .off avec **OpenGL**, qui nous permet d'obtenir la carte de profondeur. Nous utilisons **OpenCV** pour ensuite y appliquer Canny.

On obtient ainsi un ensemble d'images pour chaque modèle qui ressemblent à des croquis (figure 2 page suivante).

Nous effectuons ensuite les étapes suivantes :

- Afin d'améliorer les contours, on applique 4 filtres de Gabor à chaque image, suivant 4 directions. Cela permet de renforcer les contours dans la direction choisie et d'atténuer les autres. On calcule pour ça la transformée de Fourier d'une image (avec OpenCV), on la multiplie par le filtre, puis on revient dans le domaine spatial. L'article restait assez flou sur les détails techniques de cette partie, qui s'est révélée chronophage et difficile ;
- On découpe l'image obtenue en 1024 cellules d'aire 7,5% l'aire de l'image. Chaque cellule est ensuite divisée en 64 cases. Pour chaque case on somme les pixels qu'elle contient. Ces  $64 \times 4$  valeurs forment un mot ;
- On élimine aléatoirement une partie de ces mots afin de n'en garder qu'un million. Cela représente néanmoins un volume d'environ 256 Mo ;

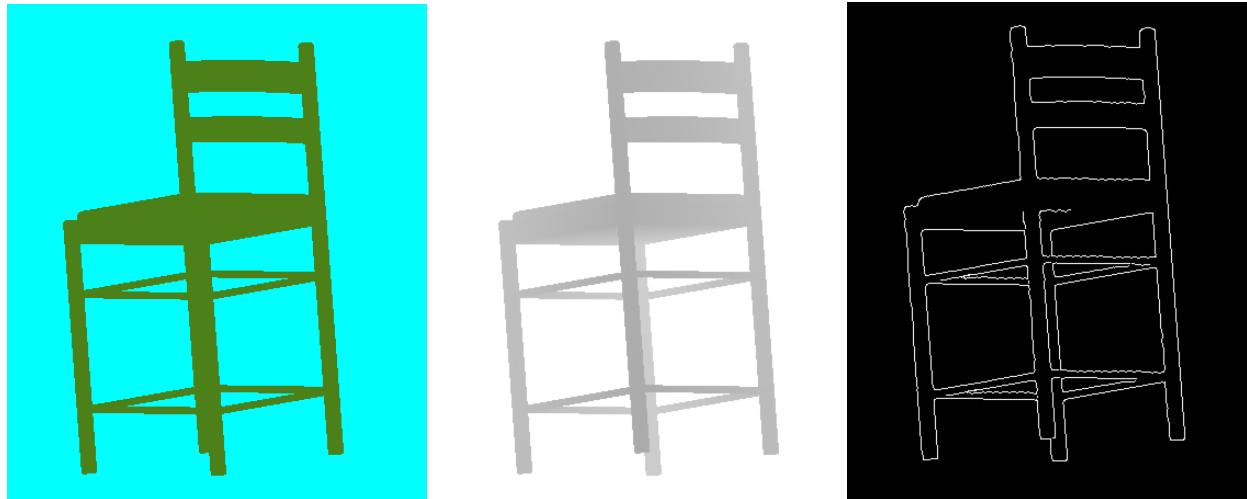


FIGURE 2 – Le modèle 3D, la carte de profondeur, puis le résultat après la détection des contours via Cany

- Grâce à l'algorithme k-means on réduit ce million de mots à un millier, qui vont composer notre alphabet ;
- Enfin on calcule l'histogramme de chaque image en regardant pour chaqu'une de ses 1024 cellules de quel mot de l'alphabet elle est le plus proche.

Cet ensemble d'étapes pouvant s'avérer très long (une vingtaine d'heures pour les 1800 modèles), nous stockons le résultat obtenu (l'alphabet et les histogrammes) dans un fichier. De façon générale, l'article passait sous silence toute question relative à l'efficacité, mais c'est un problème auquel nous avons dû être très attentifs. En effet, avec 1800 modèles, 50 projections par modèles et 1024 cellules de longueur 256 par projection, on peut facilement atteindre 20 Go en mémoire.

## 2.2 Recherche d'un modèle

On commence par charger l'alphabet et les histogrammes des modèles à partir du fichier dans lequel nous avions sauvégarde ces données.

Lorsqu'on donne un croquis au programme, on calcule son histogramme comme précédemment (on applique Cany, Gabor et on déceoupe en cellules). Ensuite on cherche simplement de quels histogrammes il est le plus proche.

## 3 Résultats

À venir...

## 4 Pistes d'amélioration

Tout au long du projet, nous avons fait des choix, car l'article proposait souvent plusieurs solutions.

Nous choisissons les directions selon lesquelles projeter le modèle de manière aléatoire mais uniforme. Une autre possibilité est d'essayer de calculer la probabilité d'une projection en utilisant plusieurs facteurs tels que la longueur de la projection, l'aire de la projection, et les variations de profondeur.

Pour l'obtention des croquis à partir des modèles 3D nous avons décidé d'appliquer Canny à la carte de profondeur. Cela donne de bons résultats, mais l'algorithme Suggestive Contours qui s'applique directement au modèle 3D pourrait donner de meilleurs résultats.

Nous pourrions également augmenter la résolution des projections, qui est actuellement limitée à 800x800. Généralement les contours sont assez fins (4-5 pixels) et peu lisses.

Il est également possible d'agir sur le nombre de projections réalisées pour chaque modèle. Nous nous sommes limités à 50 pour des raisons d'efficacité, mais les auteurs suggèrent 100.