

Image

UNIVERSITÉ DE TECHNOLOGIE DE BELFORT-MONTBÉLIARD

Serious game de rééducation

Rapport de IN52 - IN54

BOUCHEREAU Thomas - BRUNET Pierre - PERREZ Stéphane

Département Informatique - Imagerie, Interaction et Réalité Virtuelle

Université de Technologie de Belfort Montbéliard
90 010 - BELFORT CEDEX

Responsable TX52
GECHTER Frank

Enseignant VI50
CHEVRIAU Sébastien

Table des matières

1	Objectifs du projet	3
2	Développement	3
2.1	Récupérer la face du dessus de la boîte de jeu	3
2.2	De la 3D à la 2D	4
2.3	Analyse des fréquences	5
2.4	Apprentissage	6
3	Conclusion	6

1/ Objectifs du projet

Le but du projet est de faire de la reconnaissance de jeux de plateaux à partir de leur boîte de jeu et d'une base de données. Il nous faut donc procéder en plusieurs étapes. Ainsi, nous devons recentrer la photo sur la face du dessus puis la recadrer afin d'avoir la meilleure analyse possible. Ensuite, nous allons comparer le pourcentage de couleur entre les jeux de la base de données et la photo. Cette base de données est censée se créer au fur et à mesure à l'aide d'une interface.

Nous avons décidé de diviser le projet en trois parties. Ainsi, le premier étudiant s'occuperait du recadrage de la face du dessus de la boîte. Le second prendrait en charge la comparaison de deux images données. Enfin, le dernier créerait la base de données et l'interface graphique afin que l'application soit utilisable en dehors de tests divers.

2/ Développement

2.1/ Récupérer la face du dessus de la boîte de jeu

Pour récupérer la face du dessus, il est nécessaire de pouvoir distinguer les contours dans l'image. Nous avons donc décidé d'utiliser la transformée de Hough pour cela. Mais avant tout, il nous faut enlever le bruit de l'image afin d'éviter des contours inutiles. Nous avons donc utiliser la technique d'érosion puis avons rogné l'image en fonction de la densité restante dans les lignes de l'image.

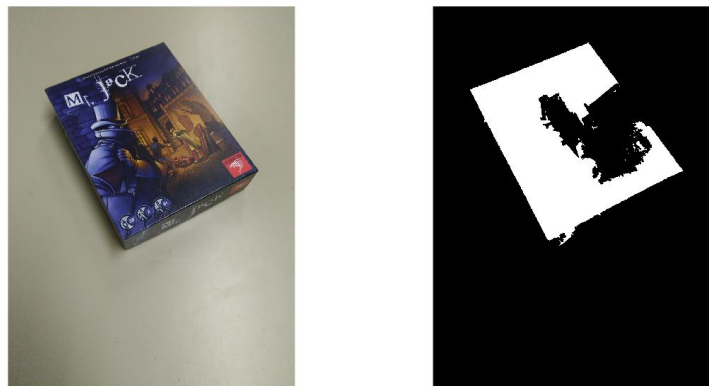


FIGURE 1 – Image binaire et érodée

Après ce pré-traitement, on applique la transformée de Hough à des lignes faisant au moins une certaine longueur pour éviter d'avoir des lignes du dessin du jeu. On allonge ensuite ces lignes et on récupère les points d'intersection.



FIGURE 2 – Image binaire rognée

Ces points d'intersection ont été produits grâce aux fonctions *polynomes* et *polyval*. Cependant, les lignes générées ne sont pas forcément dans le bon ordre. Par conséquent, les points ne le sont pas non plus. De plus, on génère toutes les intersections de lignes possibles. Il faut donc faire un tri pour éviter les duplicas et également les trier dans un ordre donné. C'est le dernier traitement que j'applique avant de passer à l'analyse de l'image.

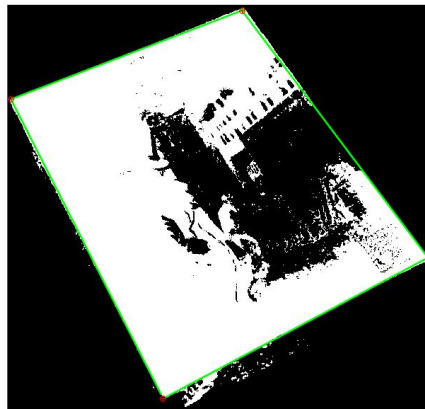


FIGURE 3 – Contours de la face du dessus

2.2/ De la 3D à la 2D

Une fois les points récupérés, il nous faut adapter l'image pour la rendre comparable avec l'image de base. On utilise donc les points d'intersection précédemment calculés.

La fonction `fitgeotrans` permet de rapprocher une image d'une autre grâce à un tableau de points mouvants et un tableau de points fixes. Le tableau de points mouvants doit *rentrer dans*

(*fit* en Anglais) le tableau de points fixes.

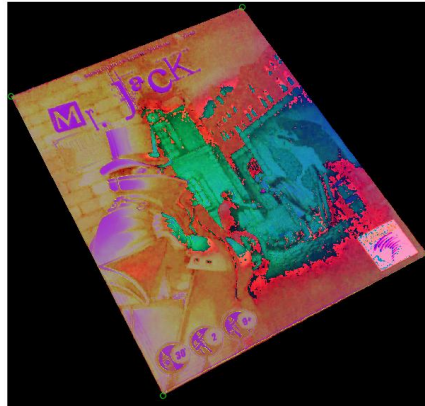


FIGURE 4 – Points mouvants en vert sur l'image

C'est à partir de ce moment-là que nous avons rencontré des problèmes. En effet, la fonction n'est pas aussi efficace qu'elle le sous-entend. Malgré tous nos efforts, nous n'avons pu que redresser la boîte sans pouvoir changer les angles de l'image.

Vous verrez sur les figures 5 et 6 les différents résultats obtenus grâce à la fonction Matlab *fitgeotrans*.

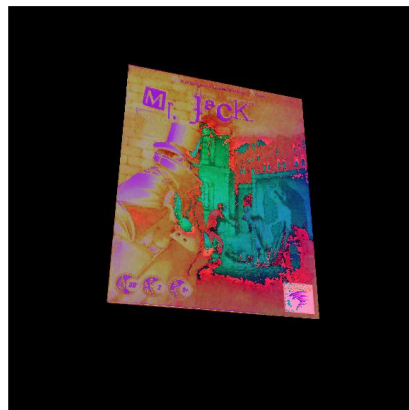


FIGURE 5 – Résultat de la fonction *fitgeotrans*

2.3/ Analyse des fréquences

Nous avons donc décidé de revenir à l'image pré-traitée et ne comparer que les fréquences de couleurs afin de pouvoir rapidement reconnaître un jeu. Cela s'est révélé plus logique et plus



FIGURE 6 – Recadrage maximal obtenu

optimisé en prenant comme couleurs le HSL. Nous n'avions alors qu'à récupérer le Hue (à savoir la nuance) de chaque pixel et comparer les densités par rapport à la base de données.

2.4/ Apprentissage

3/ Conclusion