

Description de réalisation du projet

1ère étape :

- Application des filtres sur les différents images de la base de données afin d'éliminer le bruit et faciliter le repérage des lignes.
- Détection des lignes horizontale en utilisant la Transformée de Hough. (on définit **Z** le nombre des lignes horizontales).

Principe de la Transformée de Hough :

Le but est de détecter les contours et les représenter dans un espace de paramètre a deux dimensions (une matrice par exemple), pour cela l'algorithme se résume sur ces principales instructions:

- ★ détermination de la courbe correspondante dans l'espace des paramètre pour chaque point de contour détecté.
- ★ Stocker dans un vecteur à deux dimensions les valeurs des deux paramètres (angles et radiant (ρ , θ))
- ★ détection de pics dans la matrice d'accumulation.
- Dessiner les lignes détectés sur les images de façon rectangulaire qui désignera la marche de l'escalier.
- Chaque rectangle est une composante connexe.
- Chaque niveau de ligne horizontales indique une marche de l'escalier.

2ème étape:

Détection des composantes connexes (les marches de l'escalier).

3ème étape:

Calculer la GFD (Generic Fourier Descriptors) de chaque composante connexes détecté, afin d'éliminer les composantes connexes qui n'ont pas une forme rectangulaire.

Le principe de la GFD (Generic Fourier Descriptors):

On va s'appuyer sur la méthode des chercheurs D. Zhang et G. Lu expliqué dans leur document : [Abstract Research for GFD](#)

- Application de deux Transformé de Fourier Polaire PFT:
 - PF1 : en passant l'image et la transformée de fourier dans le domaine polaire
 - PF2 : Faire la transformation de fourier 2D discrète de l'image passée en coordonnées polaire.

Exemple illustré en image:

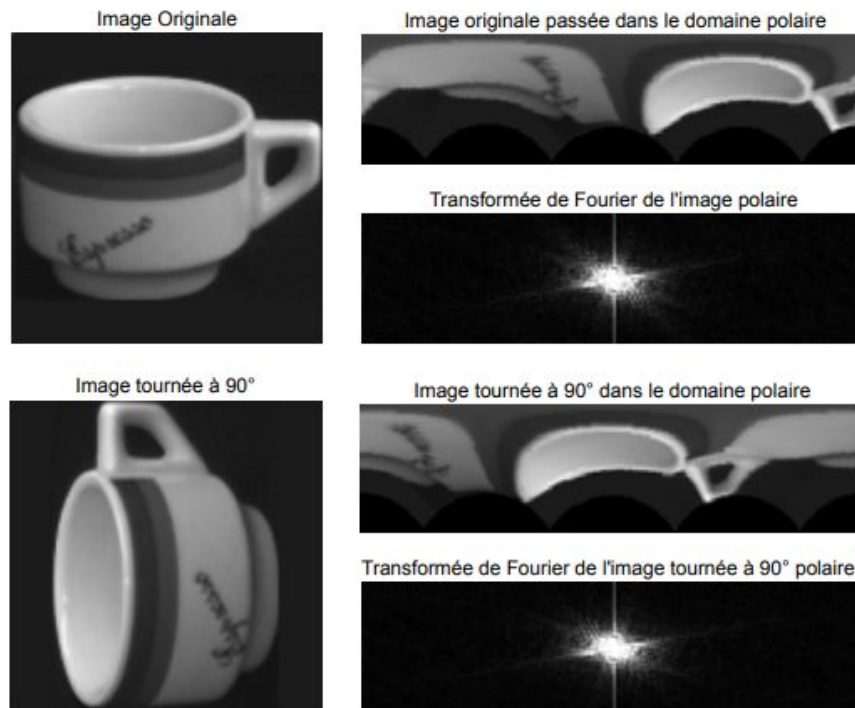


FIG. 1 – La transformée de Fourier Polaire

Le but est d'obtenir des descripteurs de Fourier invariant, dans l'exemple de Fig. 1 l'invariance de translation est obtenue en prenant le centre de gravité de l'objet en image comme l'origine de l'espace polaire.

Puis l'invariance en échelle et en rotation sont vérifiées par l'ensemble des descripteurs réel (les descripteurs sont définis par des équations qu'on ne citera pas ici).

Résultat : Des vecteurs accumulant les caractéristiques de chaque composante; Exemple d'un vecteur:

[256.45, 0.74, 155.9412 ...]

4ème étape:

Calculer la **GFD** d'une image qui contient un rectangle afin d'obtenir un vecteur de caractéristiques du rectangle (**Vect**).

5ème étape:

Comparaison entre les vecteurs : Entre (**Vect**) et les autres vecteurs des composants connexes, En utilisant la distance de **Manhattan**

6ème étape:

Classification: Compter les composant connexe (**X**) ayant une distance inférieure à un seuil (**S**) définie.

7ème étape:

Décision: en fonction de nombre de marches = $((X/2) + (Z/2))/2$

le nombre des contre-marches ($X/2$)

Référentiel de documentation :

- Algorithme de Transformée de Hough :
https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_houghlines/py_houghlines.html
- Generic Fourier Descriptors :
<https://www.labri.fr/perso/schlick/mimse/Zhang%20&%20Lu%20-%20Generic%20Fourier%20Descriptor%20-%202002.pdf>
- Transformée de Fourier Polaire :
<https://www.sciencedirect.com/science/article/pii/S1063520305001065>