

EFREI

Segmentation en Points d'Intérêt

Patrick Bonnin

Segmentation en Points d'Intérêt

Points de Moravec :

La Corrélation :

La *Corrélation*¹, comme l'*Auto Corrélation* (Corrélation sur l'image elle même) consistent à comparer une imagerie² de référence à une autre imagerie de même taille extraite d'une autre ou de la même image. La ressemblance est évaluée par un score à maximiser : à base, par exemple de somme de produits. La différence est évaluée par un coût à minimiser, à base par exemple de somme de carrés ou de valeurs absolues de différences.

Parmi toutes les positions possibles de la seconde imagerie, celle donnant le score ou le coût optimal désigne la position de l'imagerie de référence dans la seconde image.

Algorithme :

L'algorithme de Moravec utilise une fonction d'Auto Corrélation pour détecter les Points d'Intérêt. En effet :

- un point à l'intérieur d'une région a une auto corrélation faible dans toutes les directions ;
- un point situé sur un contour a une auto corrélation faible dans la direction du contour ;
- seul, un point de contour anguleux, ou point d'intérêt a une auto corrélation élevée dans toutes les directions.

Ainsi, en chaque point de l'image :

$$P(x, y) \in [2, N_{lig} - 3] \times [2, N_{col} - 3]$$

la fonction $E(x, y)$ est le minimum des huit auto corrélations dans les directions des pixels voisins du voisinage 3×3 centré.

$$E(x, y) = \min\{E(x, y)_{(d_u, d_v)} / \forall (d_u, d_v) \in [-1, 1]^2, (d_u, d_v) \neq (0, 0)\}$$

avec

$$E(x, y)_{(d_u, d_v)} = \sum_{j=0}^{j=2} \sum_{i=0}^{i=2} |I(x + d_u + i - 1, y + d_v + j - 1) - I(x + i - 1, y + j - 1)| / 9$$

$P(x, y)$ point d'intérêt de Moravec **si et seulement si** $E(x, y)$ est maximum local.

¹vue dans le cours sur le Mouvement, dans le cadre de la poursuite de cible

²Petite image extraite d'une image

Implantation :

Le but de ce paragraphe est de comprendre le code source fournit, et de valider qu'il réalise bien l'algorithme précédent.

1°) Valider l'étendue du balayage vidéo.

```
for(POINT_Y(point) = 2; POINT_Y(point) < NLIG(image) - 2;  
    POINT_Y(point)++)  
for(POINT_X(point) = 2; POINT_X(point) < NCOL(image) - 2;  
    POINT_X(point)++)
```

2°) Que représentent les différents points `EdPoint : point, pointv, pointDepl, pointvv` ?

3°) Quel est le rôle de la boucle : `for (k = 0; k < 8; k++)`

4°) Dans quelle partie du code s'effectue la comparaison pixel à pixel entre les deux imagerie ? Quel opérateur est-il utilisé ? S'agit-il d'un score ou d'un coût ?

5°) Où l'expression $E(x, y)_{(d_u, d_v)}$ est-elle évaluée ?

Idem pour $E(x, y) = \min\{E(x, y)_{(d_u, d_v)}\}$

6°) Conclure sur la simplicité de l'opérateur !

Résultats :

La figure 1 présente les résultats obtenus par l'opérateur de Moravec sur l'image du Bureau.



Figure 1: Résultats sur l'image Bureau.

Remarquons que les angles des fenêtres sont très correctement détectés. Le seuil a été choisi sur la valeur de la fonction $E(x, y)$, ici pris à 5, avec la taille 5×5 pour le voisinage local, ce qui permet d'obtenir les deux coins de part et d'autre des barreaux horizontaux des fenêtres, ce que ne permet pas le voisinage de taille 7×7 .

Malheureusement certains points d'intérêt persistent sur les obliques, dû à la pixelisation (effet d'aliasing) (cf Figure n° 2).



Figure 2: Résultats sur une Image de Maison.