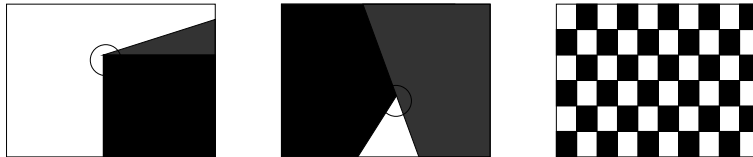


Détection de points d'intérêts

La détection de points d'intérêts (ou *coins*) est, au même titre que la détection de contours, une étape préliminaire à de nombreux processus de vision par ordinateur. Les points d'intérêts, dans une image, correspondent à des doubles discontinuités de la fonction d'intensités. Celles-ci peuvent être provoquées, comme pour les contours, par des discontinuités de la fonction de réflectance ou des discontinuités de profondeur. Ce sont par exemple : les coins, les jonctions en T ou les points de fortes variations de texture.



*Différents types de points d'intérêts :
coins, jonction en T et point de fortes variations de texture.*

Avantages des points d'intérêts :

1. Sources d'informations plus fiable que les contours car plus de contraintes sur la fonction d'intensité.
2. Robuste aux occultations (soit occulté complètement, soit visible).
3. Pas d'opérations de chaînage (-> contours !).
4. Présents dans une grande majorité d'images (\neq contours !).

1 Différentes approches

De nombreuses méthodes ont été proposées pour détecter des points d'intérêts. Elles peuvent être classées grossièrement suivant trois catégories :

1. Approches contours : l'idée est de détecter les contours dans une image dans un premier temps. Les points d'intérêts sont ensuite extraits le long des contours en considérant les points de courbures maximales ainsi que les intersections de contours.
 2. Approches intensité : l'idée est cette fois-ci de regarder directement la fonction d'intensité dans les images pour en extraire directement les points de discontinuités.
 3. Approches à base de modèles : les points d'intérêts sont identifiés dans l'image par mise en correspondance de la fonction d'intensité avec un modèle théorique de cette fonction des points d'intérêts considérés.
- ➔ Les approches de la deuxième catégorie sont celles utilisées généralement. Les raisons sont : indépendance vis à vis de la détection de contours (stabilité), indépendance vis à vis du type de points d'intérêts (méthodes plus générales).

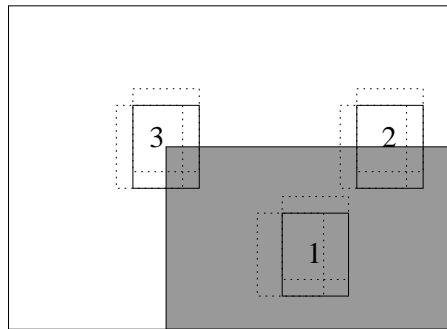
2 points d'intérêts : le détecteur de Moravec (1980)

L'idée du détecteur de Moravec est de considérer le voisinage d'un pixel (une fenêtre) et de déterminer les changements moyens de l'intensité dans le voisinage considéré lorsque la fenêtre se déplace dans diverses directions. Plus précisément, on considère la fonction :

$$E(x, y) = \sum_{u,v} w(u, v) |I(x + u, y + u) - I(u, v)|^2,$$

où :

- w spécifie la fenêtre/voisinage considérée (valeur 1 à l'intérieur de la fenêtre et 0 à l'extérieur);
- $I(u, v)$ est l'intensité au pixel (u, v) ;
- $E(x, y)$ représente la moyenne du changement d'intensité lorsque la fenêtre est déplacée de (x, y) .



Les différentes situations considérées par le détecteur de Moravec.

En appliquant cette fonction dans les trois situations principales suivantes (voir la figure ci-dessus), on obtient :

1. L'intensité est approximativement constante dans la zone image considérée : la fonction E prendra alors de faibles valeurs dans toutes les directions (x, y) .
2. La zone image considérée contient un contour rectiligne : la fonction E prendra alors de faibles valeurs pour des déplacements (x, y) le long du contour et de fortes valeurs pour des déplacements perpendiculaires au contour.

3. La zone image considérée contient un coin ou un point isolé : la fonction E prendra de fortes valeurs dans toutes les directions.

En conséquence, le principe du détecteur de Moravec est donc de rechercher les maxima locaux de la valeur minimale de E en chaque pixel (au dessus d'un certain seuil).

3 points d'intérêts : le détecteur de Harris (1988)

Le détecteur de Moravec fonctionne dans un contexte limité. Il souffre en effet de nombreuses limitations. Harris et Stephen ont identifié certaines limitations et, en les corrigeant, en ont déduit un détecteur de coins très populaire : le *détecteur de Harris*. Les limitations du détecteur de Moravec prises en compte sont :

1. La réponse du détecteur est anisotropique en raison du caractère discret des directions de changement que l'on peut effectuer (des pas de 45 degrés). Pour améliorer cet aspect, il suffit de considérer le développement de Taylor de la fonction d'intensité I au voisinage du pixel (u, v) :

$$I(x + u, y + v) = I(u, v) + x \frac{\delta I}{\delta x} + y \frac{\delta I}{\delta y} + o(x^2, y^2).$$

D'où :

$$E(x, y) = \sum_{u,v} w(u, v) [x \frac{\delta I}{\delta x} + y \frac{\delta I}{\delta y} + o(x^2, y^2)]^2,$$

En négligeant le terme $o(x^2, y^2)$ (valable pour les petits déplacements), on obtient l'expression analytique suivante :

$$E(x, y) = Ax^2 + 2Cxy + By^2,$$

avec:

- $A = \frac{\delta I^2}{\delta x} \otimes w$
- $B = \frac{\delta I^2}{\delta y} \otimes w$
- $C = (\frac{\delta I}{\delta x} \frac{\delta I}{\delta y}) \otimes w$

2. La réponse du détecteur de Moravec est bruitée en raison du voisinage considéré. Le filtre w utilisé est en effet binaire (valeur 0 ou 1) et est appliqué sur un voisinage rectangulaire. Pour améliorer cela, Harris et Stephen propose d'utiliser un filtre Gaussien :

$$w(u, v) = \exp -(u^2 + v^2)/2\sigma^2.$$

3. Enfin, le détecteur de Moravec répond de manière trop forte aux contours en raison du fait que seul le minimum de E est pris en compte en chaque pixel. Pour prendre en compte le comportement général de la fonction E localement, on écrit :

$$E(x, y) = (x, y) \cdot M \cdot (x, y)^t,$$

avec :

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix}.$$

La matrice M caractérise le comportement local de la fonction E , les valeurs propres de cette matrice correspondent en effet aux courbures principales associées à E :

- Si les deux courbures sont de faibles valeurs, alors la région considérée a une intensité approximativement constante.
- Si une des courbures est de forte valeur alors que l'autre est de faible valeur alors la région contient un contour.
- Si les deux courbures sont de fortes valeurs alors l'intensité varie fortement dans toutes les directions, ce qui caractérise un coin.

Par voie de conséquence, Harris et Stephen propose l'opérateur suivant pour détecter les coins dans une image :

$$R = Det(M) - kTrace(M)^2$$

avec : $Det(M) = AB - C^2$ et $Trace(M) = A + B$.

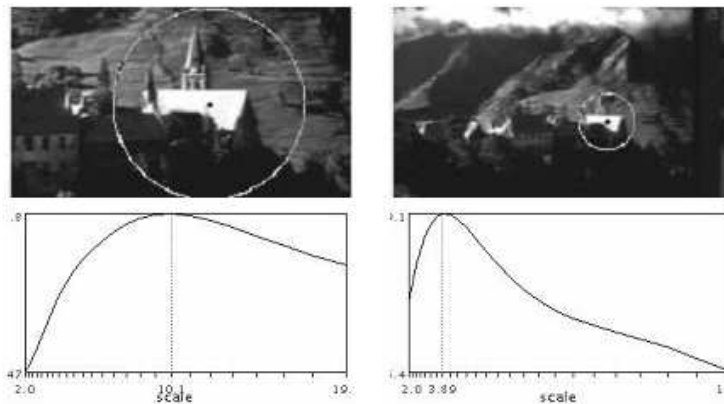
Les valeurs de R sont positives au voisinage d'un coin, négatives au voisinage d'un contour et faibles dans une région d'intensité constante.

4 points d'intérêts : SIFT (David Lowe 1999)

L'algorithme SIFT (Scale Invariant Feature Transform) a été proposé par David Lowe, université de British Columbia, en 1999 pour détecter et décrire des zones d'intérêts (local features) dans une image. A noter qu'il s'agit ici non seulement de détecter mais aussi de caractériser, par des valeurs, pour pouvoir reconnaître (**mettre en correspondance**) par la suite ces zones ou points d'intérêts dans d'autres images de la même scène. Cet algorithme a eu un succès très important au sein de la communauté vision, mais aussi en dehors de la communauté, et de nombreuses adaptations existent.

L'idée générale de SIFT est de trouver des *features* qui sont invariants à plusieurs transformations : rotation, échelle, illumination et changements mineurs du point de vue.

4.1 Détection



Mikolajczyk (2002) : les extrema du LoG (Laplacian of Gaussians) donne l'échelle intrinsèque.

Le principe de la détection est donc de trouver les extrema dans l'espace-échelle Gaussien. L'espace-échelle Gaussien d'une image $I(x, y)$ est défini par la fonction :

$$L(x, y, \sigma) = g_{\sigma} * I(x, y)$$

où g_{σ} est le filtre gaussien $g_{\sigma} = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$ et σ est le paramètre d'échelle.

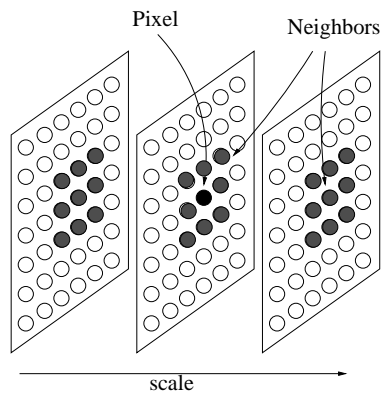


Filtrages Gaussiens successifs d'une image

Pour trouver ces extrema, au lieu de regarder la fonction LoG qui est coûteuse à calculer, on approxime avec la fonction DoG (pour Difference of Gaussians) suivante :

$$DoG(x, y) = L(x, y, k\sigma) - L(x, y, \sigma)$$

Les extrema sont alors les pixels qui présentent une intensité maximum, ou minimum, par rapport à leurs voisins directs dans l'image (8 voisins) ainsi qu'à ceux dans l'espace-échelle (9 voisins dans l'échelle précédente et 9 voisins dans l'échelle suivante).

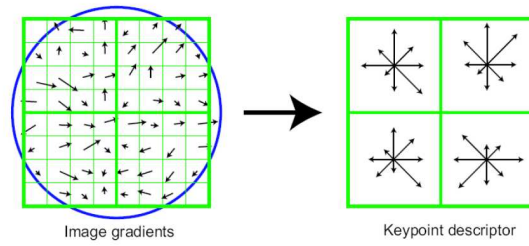


Les extrema de l'espace-échelle ainsi obtenus sont nombreux, pour les filtrer :

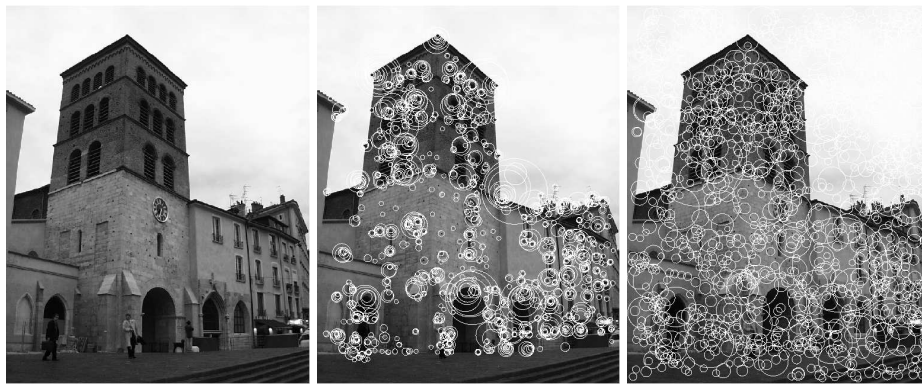
1. Les candidats avec peu de contraste sont éliminés.
2. Les réponses correspondant à des contours sont éliminés en considérant le Hessien de l'image DoG et un opérateur proche de celui de Harris.

4.2 Description

Le descripteur d'une région d'intérêt autour d'un point détecté représente l'histogramme des orientations des gradients dans la région.



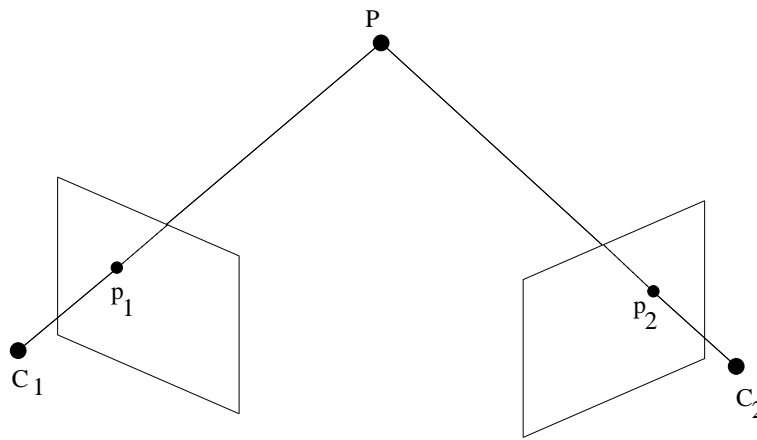
Dans le cas ci-dessus, la région est découpé en 4 zones avec 8 directions (la longueur dans chaque direction représente la somme des modules des gradients dans cette direction dans la zone considérée) soit un vecteur de description de 32 valeurs.



Résultats avec Harris et Sift.

Mise en correspondance

Considérons deux images 1 et 2 d'une scène, il s'agit ici de déterminer, pour un élément de l'image 1, l'élément qui lui correspond dans l'image 2 et éventuellement dans d'autres images. La mise en correspondance de primitives est un problème fondamental de la vision par ordinateur. C'est un processus intermédiaire entre les processus dit de *haut niveaux* : reconstruction, reconnaissance, etc., et ceux de *bas niveaux* : extraction d'indices.



Deux pixels p_1 et p_2 correspondants.

La mise en correspondance, ou appariement, peut s'appliquer à différents types de primitives :

1. points ;
2. segments ou contours ;
3. régions ;

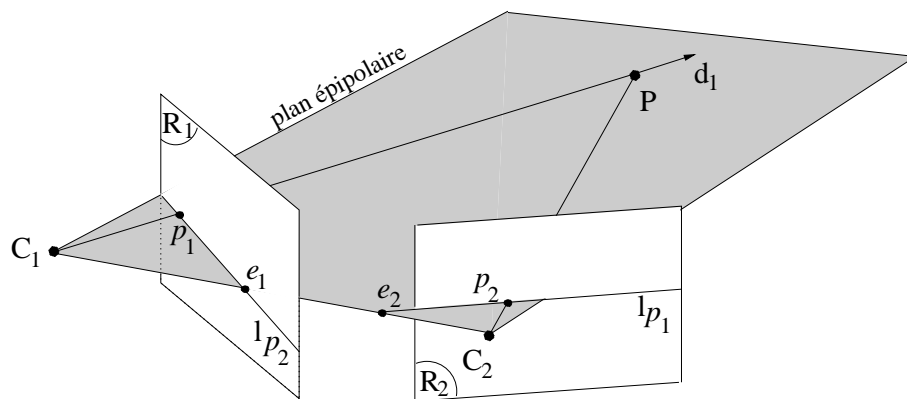
et repose sur l'introduction de mesure de similarité (ressemblance) entre primitives dans plusieurs images.

- ➔ La mise en correspondance est, en général, ambiguë. Des contraintes géométriques sur la position d'un correspondant dans l'image 2 sont utilisées pour réduire le nombre de correspondants potentiels.

1 Contraintes géométriques

1.1 Contrainte épipolaire

La contrainte épipolaire est une contrainte géométrique qui réduit l'ensemble des correspondants potentiels d'un point à une droite dans l'image. Considérons le point p_1 de l'image 2.1. Les points de l'espace ayant pour image le point p_1 sont situés sur la ligne de vue de direction d_1 . Les correspondants potentiels de p_1 dans l'image 2 sont donc nécessairement situés sur la projection de la droite d'origine C_1 et de direction d_1 dans l'image 2.

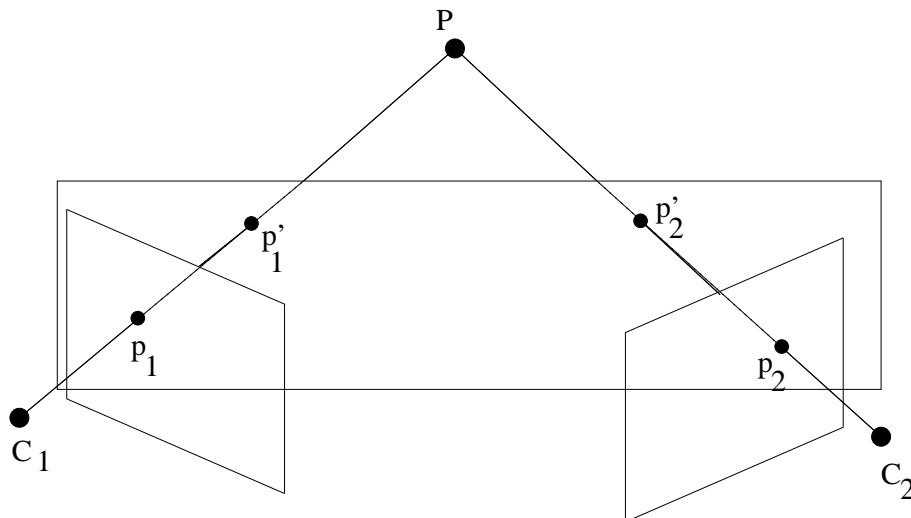


- La droite des correspondants potentiels de p_1 dans l'image 2 est appelée droite épipolaire de p_1 .
- Les intersections de la droite (C_1C_2) avec les plans rétinien R_1 et R_2 définissent les épipoles e_1 et e_2 des caméras 1 et 2.
- Les droites épipolaires dans une image s'intersectent à l'épipole.

1.2 Rectification d'images

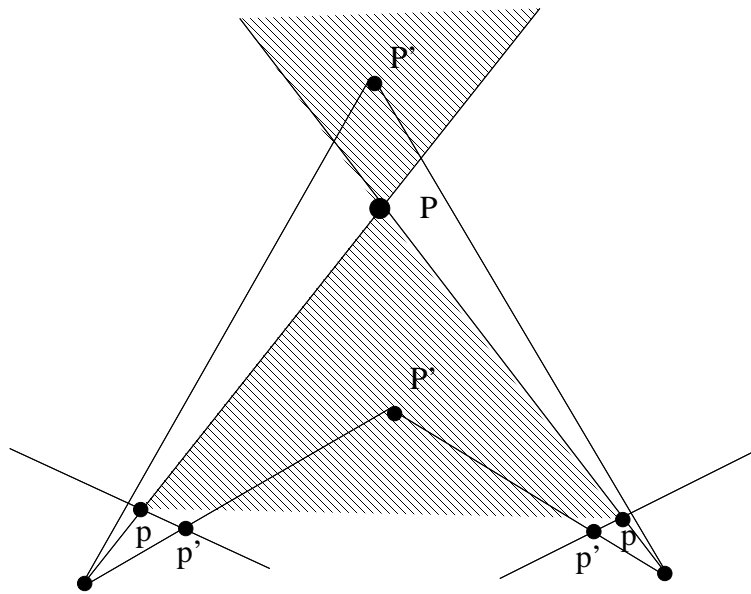
La rectification d'images consiste à recalculer, pour deux images en position générale, deux nouvelles images telles que la géométrie épipolaire de ces deux images soit *simple* ; c'est à dire que les droites épipolaires sont horizontales, ce qui implique que les deux nouveaux épipoles soient à l'infini.

Une solution consiste à garder les deux centres de projection comme nouveaux centres de projection et à utiliser comme nouveaux plans rétiniens un seul et même plan contenant la direction de la droite liant les deux centres de projection. Ce plan n'étant pas défini de manière unique, il faut choisir une orientation. On peut alors considérer que le nouveau plan rétinien contient la direction de la droite intersection des plans rétiniens des images originales.



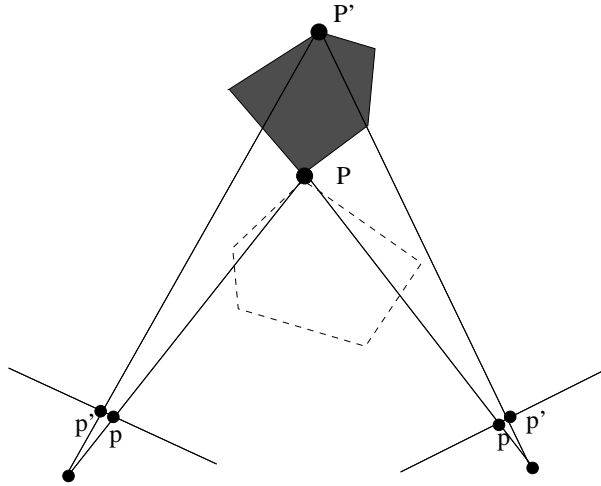
1.3 Contrainte d'ordre

La contrainte d'ordre spécifie que les projections images apparaissent dans le même ordre sur les droites épipolaires. Prenons deux points P et P' appartenant à un même plan épipolaire de deux vues 1 et 2, alors la contrainte d'ordre impose que leurs projections p et p' apparaissent dans le même ordre sur les droites épipolaires dans les images 1 et 2. Cette contrainte définit donc, pour un point P de l'espace une zone interdite.

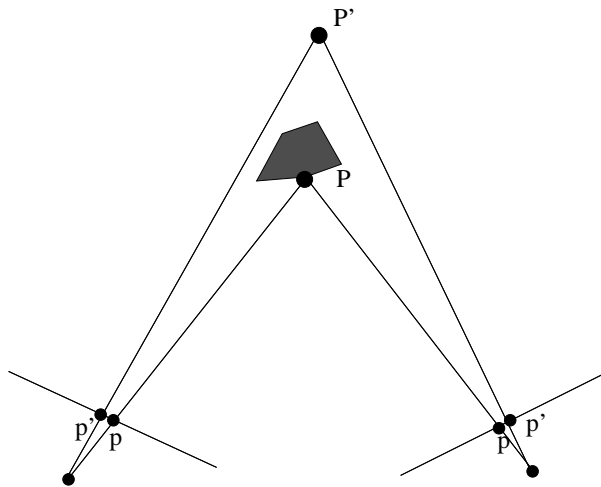


Zone interdite associée au point P .

- En pratique, la contrainte d'ordre n'est pas toujours vérifiée. Cela dépend de la taille des objets considérés.



La contrainte d'ordre est vérifiée.



La contrainte d'ordre n'est pas vérifiée.

2 Corrélation

Les méthodes de corrélation sont utilisées depuis longtemps, en particulier en photogrammétrie, pour mettre en correspondance des pixels sur la base d'informations d'intensités. L'idée est de définir une mesure de similarité entre les pixels de deux images.

- Les pixels sont les primitives les mieux adaptés pour la mise en correspondance. Les régions sont en effet mal adaptées à la mise en correspondance (la taille d'une région est différente d'une image à une autre). L'utilisation des segments quant à elle est limitée aux scènes polyédriques.

2.1 Le principe

Le principe est de considérer, pour un pixel p_1 de l'image 1, une fenêtre rectangulaire centrée en p_1 et de calculer sa corrélation/distance avec une fenêtre dans la deuxième image. La fonction de corrélation est alors maximum en p_2 correspondant de p_1 dans la deuxième image (distance minimum).



Les mesures de distance les plus classiques, pour une fenêtre de taille $2N + 1$ $2P + 1$, sont :

- Sum of Absolute Distances (SAD)

$$SAD(p_1(u_1, v_1), p_2(i_2, v_2)) = \sum_{i=-N}^N \sum_{j=-P}^P |I_1(u_1+i, v_1+j) - I_2(u_2+i, v_2+j)|.$$

- Sum of Squared Distances (SSD)

$$SSD(p_1(u_1, v_1), p_2(i_2, v_2)) = \sum_{i=-N}^N \sum_{j=-P}^P (I_1(u_1+i, v_1+j) - I_2(u_2+i, v_2+j))^2.$$

Le lien avec la corrélation se fait en considérant le produit de corrélation qui s'écrit pour SSD :

$$C_{12}(p_1(u_1, v_1), p_2(i_2, v_2)) = \sum_{i=-N}^N \sum_{j=-P}^P I_1(u_1 + i, v_1 + j) I_2(u_2 + i, v_2 + j).$$

La corrélation doit alors être maximum pour deux pixels correspondants. Dans la pratique, on normalise les fonctions d'intensités sur les fenêtres considérées pour limiter l'influence des caractéristiques propres des caméras :

$$C(p_1(u_1, v_1), p_2(i_2, v_2)) = \frac{1}{K} \sum_{i=-N}^N \sum_{j=-P}^P (I_1(u_1 + i, v_1 + j) - \overline{I_1(u_1, v_1)})(I_2(u_2 + i, v_2 + j) - \overline{I_2(u_2, v_2)}),$$

avec :

$$\overline{I_1(u_1, v_1)} = \frac{1}{(2N + 1)(2P + 1)} \sum_{i=-N}^N \sum_{j=-P}^P I_1(u_1 + i, v_1 + j),$$

$$K = (2N + 1)(2P + 1)\sigma_1(u_1, v_1)\sigma_2(u_2, v_2),$$

$$\sigma_1^2(u_1, v_1) = \frac{1}{(2N + 1)(2P + 1)} \sum_{i=-N}^N \sum_{j=-P}^P (I_1(u_1 + i, v_1 + j) - \overline{I_1(u_1, v_1)})^2.$$

Le principe s'applique aux distances SAD et SSD, on parle alors de Zero-Mean Sum of Square Distances (ZSAD) et de ZSSD.

Les limitations de ces approches restent importantes et résultent des hypothèses fortes qui sont faites :

- ➔ Les changements de points de vue n'altèrent pas l'aspect des surfaces (Lambertiennes !).
- ➔ Pas d'occultations lors de la recherche d'un correspondant.
- ➔ Une région rectangulaire dans l'image 1 correspond à une région rectangulaire dans l'image 2.
- ➔ Deux régions de couleurs constantes présentent une distance normalisée (ZSAD, ZSSD) nulle. Une solution consiste à normaliser non pas la région mais l'ensemble de l'image.

3 Méthodes par invariants

Les mesures de corrélation sont, par nature, peu robustes aux transformations (une zone rectangulaire reste une zone rectangulaire). D'autres mesures de ressemblance font intervenir des caractéristiques invariantes aux transformations telles que les dérivées, à différents ordres, de la fonction d'intensité.