

Adaptation du détecteur de Harris pour l'indexation de textures

Jérôme DA RUGNA¹, Hubert KONIK¹

¹Laboratoire LIGIV, Equipe Ingénierie de la vision EA 3070
3 rue Javelin Pagnon, BP 505, 42007 Saint Etienne, Cedex 01, France
darugna@vision.univ-st-etienne.fr, konik@vision.univ-st-etienne.fr

Résumé – Les points d'intérêt ont été introduits dans le but de réduire le maximum d'information d'une image en quelques points. Quelques applications existent, le plus souvent dans un objectif de reconnaissance de formes ou d'appariement d'images. Dans un contexte d'indexation de textures, l'objectif de ce travail est de montrer l'apport réel de la méthode d'extraction de points d'intérêt dite de Harris. Une étude approfondie de celle-ci nous amène à proposer une adaptation pyramidale et itérative de cette méthode. Nous montrerons la perspicacité de ce choix pour l'indexation de textures à travers une série de tests sur des bases de références.

Abstract – Interest Points have been introduced in order to reduce the amount of information necessary to approximately describe entire image. The main goal of existent applications is pattern recognition or pattern matching. Our purpose is to show that a multi-resolution and iterative adaptation of the classical Harris detector can improve performance in texture classification. Some tests on some reference texture bases will demonstrate the ability of this approach in an indexation application.

L'indexation de textures est un problème complexe en analyse d'images. Il existe de nombreux descripteurs de textures dont certains font office de référence. Il est vrai qu'une telle diversité s'explique, non seulement par manque de définition, mais aussi par la diversité des objectifs recherchés. Souvent non suffisamment caractéristiques de la texture qu'ils doivent représenter, ou bien alors trop gourmands en puissance de calculs, les descripteurs classiques ne peuvent pas encore s'appliquer indifféremment sur des problèmes concrets. Certains comme la dimension fractale ou la co-occurrence cherchent à quantifier la texture par une valeur. D'autres sont plus orientés pour la comparaison de textures et ne fournissent donc pas une valeur à une image, mais seulement une distance entre deux textures, modélisant les histogrammes par exemple. Dans un contexte d'extraction de descripteurs numériques d'une texture, notre objectif est alors de montrer l'apport de notre adaptation itérative et pyramidale du détecteur de Harris. Commençons donc par présenter ce détecteur un contexte d'extraction de points d'intérêt pour l'indexation d'images[2].

1 Points d'intérêt

Les points d'intérêt sont à l'origine dérivés d'une volonté de caractériser les zones contenant le plus d'information visuelle. Néanmoins les différentes méthodes sont souvent éloignées de cela, dans le sens où elles recherchent plutôt à être insensibles à certaines transformations géométriques. Le choix de sélectionner les coins étant d'ailleurs le plus répandu. La qualité principale de ces méthodes est d'être relativement stables par les transformations classiques comme la rotation, la translation, le sur ou sous

éclairage ou encore les effets d'échelle, pour amener une certaine robustesse lors d'une approche indexation.

On peut noter alors trois types principaux de méthodes d'extraction de points d'intérêt.

Méthodes Différentielles Cette classe de méthodes utilisent des invariants différentiels pour extraire les points d'intérêt d'une image. La plupart sont des adaptations du détecteur dit de "Harris"[3]. Il s'agit de méthodes adaptées à la reconnaissance de formes, par exemple par appariement des points [6].

Méthodes pyramidales L'utilisation de pyramides de contraste, permet d'extraire des points qui, s'ils ne sont plus forcément des coins, renforcent par contre l'information visuelle de chaque point. Ce sont des méthodes particulièrement robustes face à la compression et au bruit[5].

Méthodes par "filtrage" En utilisant des filtres circulaires adaptatifs (SUSAN [7]) ou des filtres directionnels [4], ces méthodes ont pour avantage de bien réagir en présence de bruit important.

2 Le détecteur de Harris

Cette méthode est largement utilisée en matière d'extraction de points d'intérêt. Parler de méthode de Harris reste malgré tout délicat tant le nombre de variantes est large, toutes se spécifiant sur tel ou tel type de problème particulier. Basée sur le détecteur de Plessey, la méthode extrait donc surtout les "coins" comme points clés via une méthode différentielle reposant sur la théorie de l'information que nous ne détaillerons pas ici.

Définissons une image par sa fonction intensité $I(x, y)$ à valeurs dans \mathbb{R} . On ne considère ainsi que des images en niveaux de gris.

On note I_x et I_y les dérivées respectivement en x et y .

Les nombreuses variantes utilisent chacune différents calculs pour arriver à une fonction dont les points à extraire correspondent à des maxima locaux. Nous étudions ici la méthode de Harris originale, les autres versions restant assez proches et possédant sans aucun doute les mêmes qualités et défauts.

On définit alors la matrice C :

$$C = \begin{bmatrix} I_x^2 & I_x \cdot I_y \\ I_x \cdot I_y & I_y^2 \end{bmatrix}$$

La fonction H_k , qui dépend du paramètre k , est ensuite définie par :

$$H_k = \text{Det}[C] - k \cdot \text{Trace}^2[C]$$

En définitive, les points d'intérêt (ou coins) sont les maxima locaux (positifs) de la fonction H_k .

La complexité de cet algorithme est assez faible, puisqu'il ne s'agit que de calculer des dérivées, leur appliquer des opérations binaires et ensuite extraire les maxima locaux. Tout d'abord le premier problème intrinsèque de cette méthode est la liberté qu'elle offre pour sa mise en oeuvre. En effet, différents facteurs influent grandement sur le résultat final :

- Le choix des filtres et de leur application.
- Le calcul des dérivées.
- L'extraction des maxima locaux et l'élimination de maxima produit par du bruit.

Les figures 1 et 2 montrent un exemple de l'application de la méthode avec deux valeurs différentes de k pour une même image.

En plus de tous ces choix laissés à la programmation, l'interprétation du paramètre k est très difficile à réaliser. En effet, la figure 3 l'illustre, si la densité de la fonction H_k est croissante avec k , il n'en est pas du tout de même pour le nombre des maxima locaux, donc de points d'intérêt. Il est impossible de dire par exemple : "Pour obtenir plus de points d'intérêt il suffit d'augmenter (ou de diminuer) k ". On n'a donc pas la possibilité de régler le nombre de points d'intérêt a priori.

On notera ici que ces calculs différentiels sont sensibles à des fortes compressions type JPEG, ce qui peut se révéler gênant dans une optique indexation, pour des bases réelles généralement compressées[5].

Ainsi, bien que le détecteur de Harris soit devenu incontournable quand on parle de détection de coins ou de points d'intérêt, il n'en reste pas moins qu'il est difficile à maîtriser, surtout dans un objectif indexation de bases de données complexes et volumineuses.

3 Approche itérative du détecteur

On voit clairement que calculer une seule fonction H_k ou bien plusieurs fonctions H_k (k variant) n'est pas très coûteux en temps, les calculs les plus lourds étant fait avant

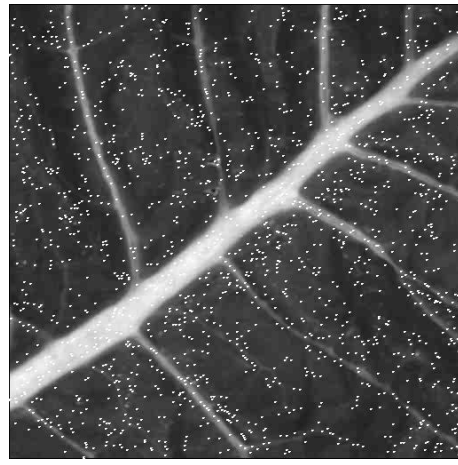


FIG. 1: $k = 0.4$

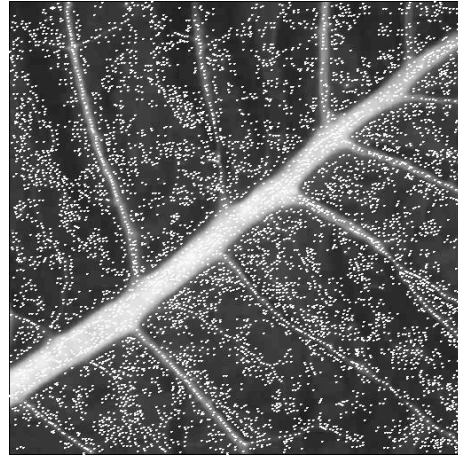


FIG. 2: $k = 0.05$

le calcul final. Ainsi, la nouvelle approche que nous allons décrire utilise en fait plusieurs valeurs de k . L'itération du calcul sur plusieurs valeurs de k permet alors de lever certaines ambiguïtés inhérentes à l'introduction du paramètre. En effet, même si sa compréhension ne semble pas immédiate, son itération permet de définir plusieurs ensembles de points d'intérêts qui n'expriment pas le même type de focalisation.

Le calcul d'invariants locaux sur les points d'intérêt permet d'extraire des descripteurs de texture. Néanmoins, comme un point d'intérêt représente l'information visuelle de la région qui l'entoure, le calcul de descripteurs statistiques sur la zone d'influence d'un point d'intérêt est plus judicieuse. Pour calculer un vecteur caractéristique relatif à chaque image il suffit ensuite de globaliser l'information portée par chaque point, par les différents moments de ceux-ci par exemple. La distance entre chaque vecteur donne en définitive la distance inter-images.

4 Apport multi-résolution

Parallèlement à ceci, dans une démarche de classification sur un continuum micro-macro, il semble intéressant d'étudier le comportement du vecteur caractéristique en

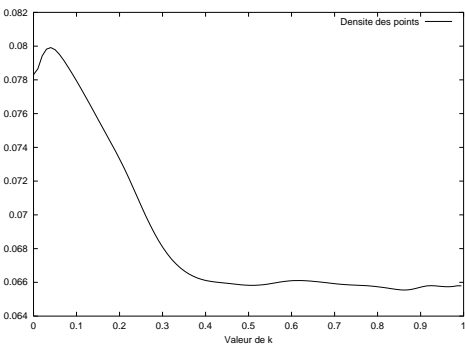


FIG. 3: Détecteur de Harris : influence de k

fonction de l'échelle d'étude. Pour cela, nous avons calculé la fonction H_k sur les premiers niveaux d'une pyramide gaussienne[1] et intégré cet aspect "multi-résolution" dans le moteur d'indexation.

Compte-tenu de la définition usuelle des pyramides gaussiennes pour des images $2^n \times 2^n$, il faut adapter cette dernière dans un contexte plus global où les tailles ne sont pas toujours calibrées. Pour cela, une nouvelle construction doit être mise en place. En effet, puisque seuls les premiers niveaux de la pyramide sont utiles, les derniers niveaux étant trop peu discriminants, on ne va calculer que les premiers niveaux de la pyramide pour chaque image, en adaptant la construction pyramidale la plus proche possible de l'image source (non forcément de type $2^n \times 2^n$). On obtient donc une pyramide de type $a.2^n \times b.2^p$ calculée uniquement sur les premiers niveaux.

5 Apport en indexation de texture

L'outil *i*COBRA¹ (Image Content Based Retrieval Application), développé au sein du laboratoire, est un outil performant et modulaire, capable d'indexer n'importe quelle base d'images discriminée par des valeurs réelles. L'insertion de méthodes est aisée et permet de juger rapidement l'apport potentiel de celles-ci en terme d'indexation d'images ou de textures. Comme le montrent les figures 4 et 5, où sont présentées les images les plus similaires à une image requête, son interface est basée sur du HTML, donc utilisable par n'importe quel navigateur.

Le principe général des tests du système *i*COBRA est d'utiliser deux ensembles, *DATA*, la base de connaissances, et *TEST*, l'ensemble test. Chacun de ces ensembles est constitué d'échantillons de texture formés à partir de la base originale. Différents moyens ont été utilisés pour former cette base, d'une part en isolant des imagerie aléatoires sur les images haute définition, d'autre part en utilisant des transformations classiques ou bien encore dans l'étape d'acquisition de l'image elle-même. Ce qui conduit à différents effets et perturbations sur les bases d'images du système tels: Zoom, compression, bruit, éclairage. Il s'agira ensuite de classer les éléments de *TEST* ayant connaissance de l'ensemble *DATA*. Une classe étant donc définie par toutes les images issues de la même texture.

1. En cours de publication sur internet, vous pouvez demander son lien à l'adresse darugna@vision.univ-st-etienne.fr

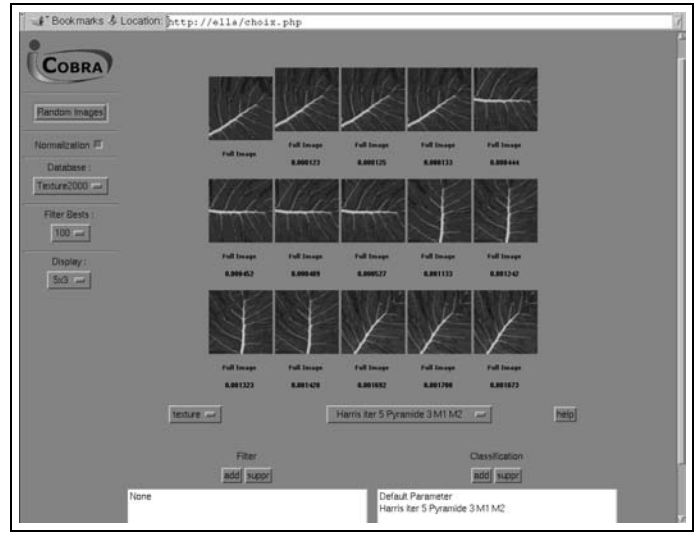


FIG. 4: Système *i*COBRA

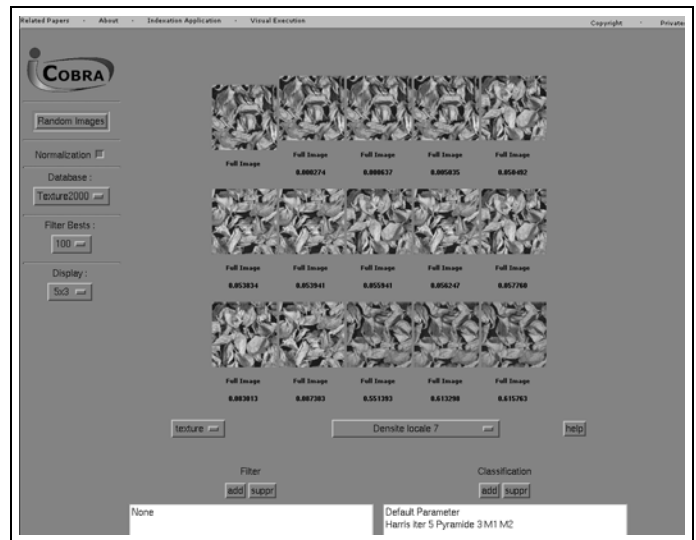


FIG. 5: Système *i*COBRA

Deux méthodes de classification sont possibles pour l'instamment : "Distance à la moyenne" et "k plus proches voisins", chacune pouvant utiliser différentes distances.

Les textures sont issues principalement de :

- La base de Brodatz
- Une base locale : "Texture2000"²
- La base VisTex³

Méthodes des k plus proches voisins Dans le contexte de la classification, la méthode des k plus proches voisins est non seulement une des plus anciennes, mais aussi une des plus usitées. Le principe se résume à affecter à un objet le type majoritaire parmi les k objets les plus similaires à lui-même. De nombreuses adaptations de cette méthode existent mais nous nous limiterons à la version de base que nous allons modéliser à présent.

2. Bases de diverses textures construite à l'aide de différents modes d'acquisitions et compressées JPEG avec différents taux

3. <http://www-white.media.mit.edu/vismod/imagery/>

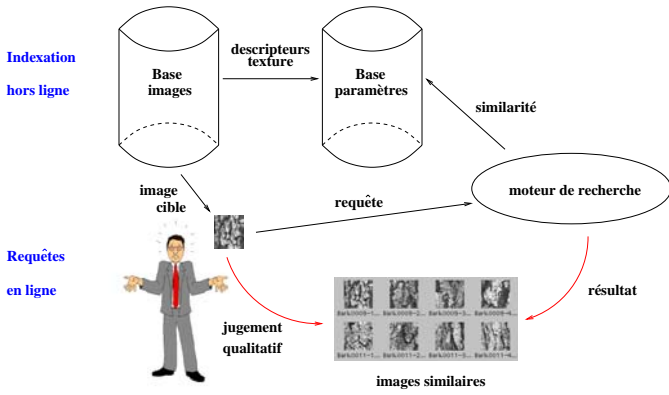


FIG. 6: Schéma d'indexation de textures

Soit $DATA = (A^i)_{1 \leq i \leq N}$ où N est le nombre d'éléments de $DATA$.

Soit X à classifier, l'ensemble des k plus proches voisins E est défini par :

$$\begin{cases} |E| = k \\ \text{Si } \zeta = \arg \max_{\gamma \in E} D(X, \gamma) \\ \forall \alpha \in \{DATA \setminus E\}, D(X, \alpha) \geq D(X, \zeta) \end{cases}$$

En pratique nous utilisons une valeur de k égale à 15. Les descripteurs ne sont pas normalisés a priori. Or il semble plus juste de les normaliser pour avoir des attributs de poids équivalents dans la décision finale. On considérera donc par la suite que les descripteurs sont normalisés.

Résultats Le tableau 1 illustre l'apport de notre itération du paramètre k sur les bases de textures. Ce tableau montre le pourcentage de bonne classification en fonction de k . Il apparaît ainsi qu'augmenter le nombre des différentes cartes de points augmente significativement la qualité du descripteur.

TAB. 1: Qualité de la classification: Itération de k

# k	Brodatz	Texture 2000	Vistex
1	75.5	68.5	70
2	76	70	72
3	78	71	73,5
5	81	71.5	74

L'apport de l'adaptation pyramidale décrite précédemment est validée par le tableau 2, son pouvoir discriminant étant meilleur par rapport aux précédentes méthodes. Ce tableau, en fonction de la méthode, donne le taux de bon classement des images, ainsi que la moyenne de l'indice dans l'ordre des plus proches voisins de la première image erreur (Le cardinal des classes est de 38). En moyenne, on note ainsi que non seulement le taux de classement est plus probant, mais aussi généralement les premières images sont toutes de la bonne classe, ce qui améliore plus encore la prédiction.

De plus, on a réalisé une comparaison avec d'autres méthodes implémentées dans le système *i*COBRA. Le calcul des différents moments d'une part sur l'image source (d'ordres 1,2,3 et 4), d'une méthode basée sur les matrices de co-occurrence et une méthode modélisant l'évolu-

tion du moment d'ordre 2 de de l'image dans les différents niveaux de la pyramide. Notre méthode se montre plus performante que ces descripteurs, ce qui justifie le couple pyramide-itération du paramètre k utilisé.

TAB. 2: Apport de la pyramide

Méthode	1ere Erreur	Taux
Iteratif	13	78
Iteratif Pyramidale	19	92
Moments	12	73
Matrice co-occurrence	15	81
Sigma Pyramidale	14	78

6 Perspectives

Notre approche pyramidale et itérative, bien qu'intéressante en de nombreux points, possède, par sa définition en niveau de gris, un défaut majeur dans l'indexation de textures: la non prise en compte de l'aspect couleur d'une texture. Des deux voies possibles, soit d'associer un descripteur couleur, soit d'intégrer une dimension couleur au détecteur de points d'intérêt; la deuxième nous paraît la plus intéressante, au sens où elle semble capable de mieux refléter l'aspect visuel d'une texture. Cette approche nous permettra donc a priori d'étendre ce descripteur aux images dites de scènes.

Références

- [1] Peter BURT et Edward ADELSON. « The Laplacian Pyramid as a compact image code ». *IEEE Transactions on communications*, 31(4):532+, 1983.
- [2] Alberto Del DIMBO. *Visual Information Retrieval*. Morgan Kaufmann publishers, inc, 1999.
- [3] C. HARRIS et M. STEPHENS. « A combined Corner and Edge Detector ». *4th Alvey Vision Conf. Manchester*, pages 189+, 1988.
- [4] F. HEITGET, L. ROSENTHALER, R. Von det HEYDT, E. PETERHANS, et O. KUEBLER. « Simulation of neural contour mechanism: from simple to end-stopped cells ». *Vision Research*, 32(5):963–981, 1992.
- [5] Jean Michal JOLION et Stéphane BRES. « Influence du codage JPEG sur des descripteurs d'images ». *Traitement du signal*, 15(4):309–320, 1999.
- [6] Cordelia SCHMID. « *Appariement d'images par invariants locaux de niveaux de gris* ». PhD thesis, Institut national polytechnique de grenoble, 1996.
- [7] S.M. SMITH et J.M. BRADY. « SUSAN - A New Approach to Low Level Image Processing ». *Int. Journal of Computer Vision*, 23(1):45–78, May 1997.