

# Estimation Neuronale du Flot Optique dans une Séquence d'Images Débruitées par la Transformée en Curvelettes

Fella CHARIF

Département d'électronique et communication, FNTIC,  
UKMO,  
Ouargla, Algérie  
cherif.fe@univ-ouargla.dz

Abderrazak Benchabane

Département d'électronique et communication, FNTIC,  
UKMO,  
Ouargla, Algérie  
Benchabane.ab@univ-ouargla.dz

Noureddine DJEDI

Département d'informatique Université de Biskra,  
Algérie  
djedi\_nour@yahoo.fr

Abdelmalik TALEB-AHMED

IUT GE2I, Laboratoire LAMIH UMR CNRSUVHC 8530,  
Université de Valenciennes,  
France  
Abdelmalik Taleb-Ahmed@univ-valenciennes.fr

**Résumé**— Dans le problème de l'estimation de mouvement dans une séquence d'images, deux performances critiques sont à améliorer, le temps de calcul et la précision des résultats. La méthode itérative multi-échelles de Lucas et Kanade est l'une des meilleures méthodes dans ce domaine. Dans cet article on a introduit le réseau de neurones de Zhang pour accélérer le processus d'estimation suite à la nature parallèle de ces derniers. Du côté précision, on a proposé d'utiliser la transformée en curvelettes pour le pré-filtrage des images. Les simulations numériques sur des séquences d'images artificielles et réelles montrent la bonne performance de la méthode proposée.

**Mots clé**— Estimation de mouvement, Séquences d'images, Réseau de Zhang, Méthode de Lucas et Kanade.

## I. INTRODUCTION

L'estimation de mouvement connaît un intérêt croissant du fait du nombre considérable d'applications (contrôle du trafic routier, guidage de robot, compression de séquences d'images, génie biomédical, météorologie, reconstruction 3D...). Elle consiste à extraire un vecteur de mouvement à partir d'une séquence d'images en faisant l'hypothèse que l'intensité est conservée au cours du déplacement [1].

Les méthodes d'estimation de mouvement peuvent être classées en quatre catégories principales: les méthodes de corrélation, les méthodes spatio-temporelles basées sur l'énergie, les méthodes basées sur des modèles paramétrés de mouvement et les méthodes différentielles [1]. Bien que beaucoup de progrès aient été faits sur ces méthodes, quelques problèmes fondamentaux sont encore ouverts, les méthodes de corrélation sont coûteuses en temps de calcul pour une estimation dense du champ de déplacement, les méthodes basées sur l'énergie faisant intervenir la transformée de Fourier

des images, les calculs deviennent plus lourds pour des résultats équivalents [1]. Pour les méthodes paramétriques, les modèles de mouvement sont largement exploités pour l'analyse du mouvement apparent, toutefois, ils ne permettent pas toujours de fournir une représentation appropriée du mouvement dominant dans la scène induit par le déplacement de la caméra. Quant aux méthodes différentielles, elles sont plus robustes et précises que les autres méthodes. La technique de Lucas et Kanade multi-échelles est l'une des méthodes différentielles les plus utilisées [2-4]. Elle offre une robustesse au bruit et une grande capacité à estimer les grands déplacements.

Pour améliorer les performances de la méthode de Lucas et Kanade, deux points critiques sont considérés : la précision et le temps de calcul. Pour la précision, on a introduit la transformée en curvelettes pour un prétraitement des images. Cette transformée qui est un outil très intéressant, a fait ses preuves en traitement d'images [5-7]. Quant à la rapidité, on a utilisé un réseau de neurone nommé réseau de Zhang [8] qui estime le flot optique de tous les pixels simultanément au lieu de son calcul séquentiel dans la méthode de Lucas et Kanade.

L'article est organisé comme suit. Dans la section 1 on discute brièvement la transformée en curvelettes, la section 2 présente la méthode de Lucas et Kanade (LK), dans la section 3, on donne le principe de l'estimateur neuronal de Zhang. Les résultats obtenus sur des séquences synthétiques et réelles sont présentés dans la section 4. On achève cet article par une conclusion.

## II. TRANSFORMEE EN CURVELETTES

La transformée en curvelettes (TC) est une généralisation de la transformée en ridgelettes. Elle est récemment appliquée

au domaine du filtrage de bruit gaussien avec des résultats très prometteurs [5]. Dans un premier temps on exposera brièvement les relations de base des transformations en ridgelettes (TRid), puis on présentera la transformée en curvelettes et leur implémentation numérique. Enfin, on proposera d'utiliser un filtre non-linéaire pour réduire les bruits.

### A. Transformée en ridgelettes

La transformée en ridgelettes [5] est définie comme une ondelette  $\psi$  construite le long d'orientation  $\theta$  définie dans le plan cartésien  $(x_1, x_2)$  par l'équation :

$$\psi_{a,b,\theta}(x_1, x_2) = a^{-1/2} \psi((x_1 \cos \theta + x_2 \sin \theta - b)/a) \quad (1)$$

où  $a > 0$  est le facteur d'échelle,  $b \in \mathbb{R}$  le paramètre de translation,  $\theta \in [0, 2\pi[$  l'angle de projection et  $\psi$  la base d'ondelette.

Les coefficients de ridgelettes  $Rid_I$  d'une image  $I$  sont obtenus par projection sur cette base :

$$Rid_I(a, b, \theta) = \iint \psi_{a,b,\theta}(x_1, x_2) I(x_1, x_2) dx_1 dx_2 \quad (2)$$

Cette projection est intimement liée à la transformée de Radon (T.Rad) qui consiste à intégrer une image selon un ensemble de lignes :

$$Rad_I(\theta, t) = \iint I(x_1, x_2) \delta(x_1 \cos \theta + x_2 \sin \theta - t) dx_1 dx_2 \quad (3)$$

où  $\delta$  est la distribution Dirac.

Par conséquent, la transformée en ridgelettes (2) peut être vue comme la transformée en ondelette 1-D (TO1D) de la transformée de Radon (3) selon l'axe des translations  $t$ .

$$Rid_I(a, b, \theta) = \int Rad_I(\theta, t) \psi((t - b)/a) dt \quad (4)$$

La transformée de Radon peut être calculée à partir de la transformée de Fourier inverse de la transformée de Fourier de l'image le long des lignes radiales.

$$\begin{aligned} \hat{I}(\lambda \cos \theta, \lambda \sin \theta) &= TF[Rad_I(\theta, t)] \\ &= \int Rad_I(\theta, t) e^{-\lambda i t} dt \end{aligned} \quad (5)$$

Par conséquent l'équation (4) peut être écrite dans le domaine fréquentiel :

$$Rid_I(a, b, \theta) = TF_\lambda^{-1} \left[ \hat{I}(\lambda \cos \theta, \lambda \sin \theta) TF_t[\psi(t/a)] \right] \quad (6)$$

La transformée en ridgelettes discrète s'obtient en appliquant une transformée en ondelettes 1D le long de  $Rid_I(a, b, \theta)$ , en utilisant la variable d'intégration  $t$ . Pour cela, on suit les trois étapes suivantes [9] :

- Calcul de la transformée de Fourier 2D (TF 2D)

- Conversion des coordonnées cartésiennes en coordonnées polaires.
- Calcul de la transformée inverse 1D (ITFF 1D).

### B. Transformée en curvelettes

L'objectif de la transformée en curvelettes [5] est de décrire l'image comme des petites parties d'une certaine taille et d'orientation donnée. Pour cela, une analyse multi-résolutions est appliquée à l'image avant de lui appliquer la transformée en ridgelettes localement sur des blocs dyadiques.

L'algorithme de la transformée en curvelettes est le suivant :

- Effectuer l'analyse multi-échelles en différents niveaux  $J$

$$I(x, y) = A_I^J(x, y) + \sum_{j=1}^{J-1} HF_I^j(x, y) \quad (7)$$

où  $A_I^J$  est la version grossière de l'image  $I$ , et  $HF_I^j$  sont les détails de  $I$  à l'échelle  $2^{-j}$ .

- Initialiser la taille des blocs à  $B^1 = B_{\min}$  (on utilise  $B_{\min} = 16$ ).
- Pour chacune des images hautes fréquences  $HF_I^j, j=1:J$  :
- Effectuer la transformation en ridgelettes locale;
- Si  $j \bmod 2 = 1, B^{j+1} = 2B^j$  ;
- Sinon,  $B^{j+1} = B^j$ .

Les figures 1 et 2 présentent le principe de la transformée en curvelettes ainsi qu'un exemple de décomposition en curvelettes d'une frame de la séquence 'Hydrangea' en 3 niveaux.

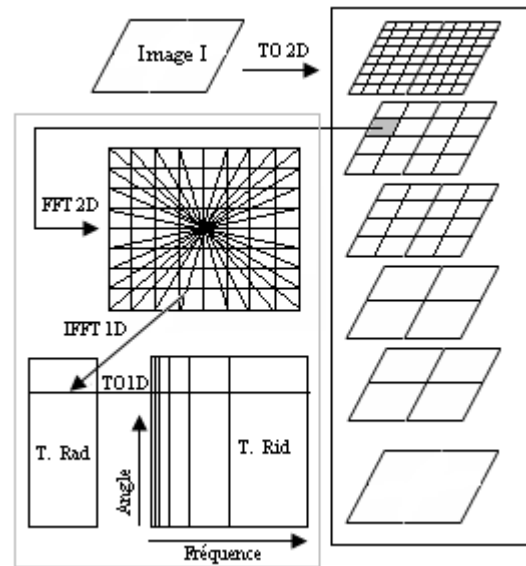


Figure 1. Principe de la transformée en curvelettes

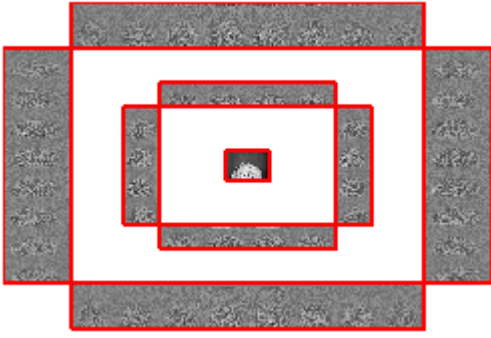


Figure 2. Décomposition en curvelettes de l'image Hydrangea en 3 niveaux

### C. Débruitage par la transformée en curvelettes

Premièrement on va calculer la transformée en curvelettes pour l'image d'entrée  $I$ , puis on lui applique un filtre non linéaire et, à la fin, on revient en calculant la transformée en curvelettes discrète inverse [10]. La relation entrée-sortie pour le filtre utilisé dans le domaine de la transformée en curvelettes est:

$$\tilde{R}_I = \begin{cases} R_I & \text{si } (|R_I| \geq \lambda_1) \\ R_I \cdot \varphi(\lambda, \lambda_1) & \text{si } (|R_I| < \lambda_1) \\ 0 & \text{si } (|R_I| < \lambda) \end{cases} \quad (8)$$

avec  $\varphi(\lambda, \lambda_1)$  est une fonction de pondération amortie de 1 à 0 dans l'intervalle  $[\lambda, \lambda_1]$ , où  $\lambda, \lambda_1$  sont des seuils. Ce filtre est robuste si les valeurs des seuils sont comprises dans l'intervalle  $[0, 4\sigma_I / (2\pi)^{1/2}]$ , où  $\sigma_I^2$  représente la variance de la variable en entrée  $R_I$ . Le meilleur affaiblissement du bruit, réalisé par ce filtre, est obtenu pour la valeur du seuil de  $\lambda_1 = 0.797\sigma_I$  [10].

Pour pouvoir vérifier et valider les résultats du filtrage, on mesure l'erreur quadratique moyenne (EQM) et le rapport signal sur bruit crête (PSNR en dB) entre chaque pixel de l'image  $\tilde{I}$  obtenue par le débruitage utilisant la TC et l'image d'entrée  $I$ .

$$EQM = \frac{1}{N.M} \sum_{i=1}^N \sum_{j=1}^M (I(i, j) - \tilde{I}(i, j))^2 \quad (9)$$

$$PSNR(dB) = 10 \cdot \log_{10} \left[ \frac{255^2}{EQM} \right] \quad (10)$$

où  $M$  et  $N$  étant les dimensions de l'image. La figure 3 présente les résultats du débruitage réalisé par la transformée en curvelettes et la transformée en ondelettes d'une frame de la séquence 'Hydrangea'. On remarque que le filtrage par la TC a de meilleurs résultats ( $PSNR=35.25dB$ ) que par le filtrage par la TO ( $PSNR=33.61dB$ ).

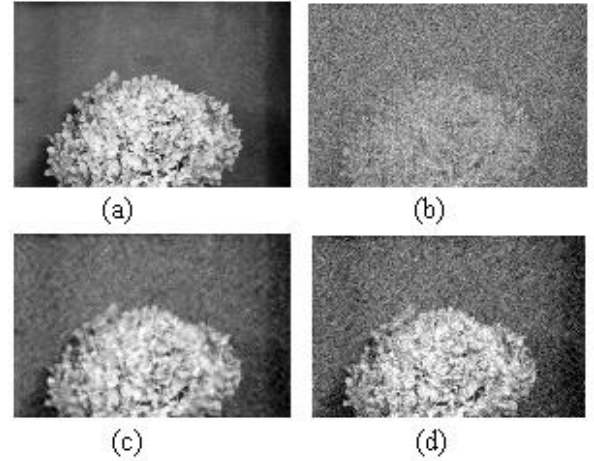


Figure 3. Débruitage de l'image Hydrangea. (a) image originale, (b) image bruitée(10%), (c) image débruitée par la TC et (d) image débruitée par la TO.

### III. METHODE ITERATIVE MULTI ECHELLES DE LK

Lucas et Kanade [2] ont proposé une méthode qui impose une contrainte locale de lissage. Ils ont supposé que l'équation du flot optique demeurait constante dans de petites régions de l'image  $\Omega$ . Ceci est exprimé par :

$$E = \min_{p \in \Omega} \sum w^2(p) (I_x(p)u + I_y(p)v + I_t(p))^2 \quad (11)$$

avec  $I(x, y, t)$  est une séquence d'image,  $V(p) = (u, v)^T$  le vecteur vitesse associé au point  $p = (x, y)^T$  à l'instant  $t$ . Les composantes  $u$  et  $v$  sont respectivement la vitesse selon les directions  $x$  et  $y$ .  $I_x$ ,  $I_y$  et  $I_t$  sont les dérivées spatio-temporelle de  $I(x, y, t)$ . Les valeurs  $w(p)$  sont définies de telle sorte qu'elles donnent plus d'influence aux pixels situés au centre de  $\Omega$  qu'à ceux situés à la périphérie du voisinage.

L'équation 11 se traduit sous forme matricielle de la manière suivante :

$$A^T W^2 A V = A^T W^2 b \quad (12)$$

où, pour  $n$  pixels dans le voisinage  $\Omega$ , nous notons :

$$A = (\nabla I(p_1), \dots, \nabla I(p_n))^T, \quad W = \text{diag}[w(p_1), \dots, w(p_n)] \text{ et}$$

$$b = -(I_t(p_1), \dots, I_t(p_n))^T. \quad \nabla I \text{ est le gradient spatial de } I.$$

En appliquant la méthode de calcul des moindres carrés, on obtient :

$$\vec{V} = G^{-1} d \quad (13)$$

avec :

$$G = A^T W^2 A = \begin{bmatrix} \sum W^2 I_x^2 & \sum W^2 I_x I_y \\ \sum W^2 I_x I_y & \sum W^2 I_y^2 \end{bmatrix} \quad (14)$$

$$d = A^T W^2 b = - \begin{bmatrix} \sum W^2 I_x I_t \\ \sum W^2 I_y I_t \end{bmatrix} \quad (15)$$

Lorsque les amplitudes des déplacements sont grandes ou inconnues a priori, il est judicieux d'effectuer l'analyse dans plusieurs bandes de fréquences, ce qui fait appel à une représentation multi-échelles. Les déplacements à amplitudes différentes peuvent donc être estimés à des niveaux de résolution différents [3].

Soit  $V_k = (u_k, v_k)^T$  le flot optique estimé au niveau  $k$  de la pyramide d'images. Le mouvement réel peut s'écrire comme la somme de  $V_k$  et d'un mouvement résiduel  $\Delta V_k$  à déterminer :

$$V = V_k + \Delta V_k \quad (16)$$

Connaissant  $V_k$ , l'hypothèse de la conservation de la luminance s'écrit :

$$I(p_i + V_k + \Delta V_k, t+1) - I(p_i, t) = 0 \quad (17)$$

à nouveau, l'équation (17) est linéarisée par un développement de Taylor, mais cette fois-ci autour de  $p_i + V_k$  :

$$\nabla I(p_i + V_k, t+1) \Delta V_k + I_t(p_i, t) = 0 \quad (18)$$

Tout algorithme différentiel de calcul du flux optique pourra être utilisé pour estimer le déplacement résiduel  $\Delta V_k$ . Pour que l'estimation soit plus précise, nous utilisons l'algorithme itératif de Lucas & Kanade [3-4] :

$$\Delta V_k = G^{-1} d \quad (19)$$

Durant chaque itération, La matrice  $G$  ne se change pas, donc elle est calculée une seule fois, et le vecteur  $d$  prend la valeur :

$$d^{it} = - \begin{bmatrix} \sum W^2 I_x I_t^{it} \\ \sum W^2 I_y I_t^{it} \end{bmatrix} \quad (20)$$

Dans l'itération suivante  $it+1$  :

$$V_k^{it+1} = V_k^{it} + \Delta V_k \quad (21)$$

On arrête le processus itératif quand  $\Delta V_k$  atteint un seuil fixé à l'avance ou pour un nombre d'itérations donné. Cette estimation incrémentale du mouvement est effectuée du sommet à la base de la pyramide, c-à-d on projette le flot optique estimé au niveau de la résolution  $k$  à un niveau de résolution plus fin  $k+1$  grâce à la formule de changement de base.

#### A. Position du problème

La méthode de LK nécessite l'inversion de la matrice  $G$  pour chaque pixel de l'image  $i \in \{1, 2, \dots, N\}$  avec  $N$  la taille de l'image. Donc on obtient le système suivant :

$$\begin{bmatrix} \sum W^2 I_x^2 & \sum W^2 I_x I_y \\ \sum W^2 I_x I_y & \sum W^2 I_y^2 \end{bmatrix} \begin{bmatrix} u_i \\ v_i \end{bmatrix} = - \begin{bmatrix} \sum W^2 I_x I_t \\ \sum W^2 I_y I_t \end{bmatrix} \quad (22)$$

Pour estimer le flot optique de tous les pixel simultanément, on pose pour  $i \in \{1, 2, \dots, N\}$ ,  $z = (z_1, \dots, z_{2N})^T \in \mathbb{R}^{2N}$  avec les cordonnés:  $z_{2i-1} = u_i$ ,  $z_{2i} = v_i$ , et  $d = (d_1, \dots, d_{2N})^T \in \mathbb{R}^{2N}$  avec les cordonnés  $d_{2i-1} = -\sum W^2 I_x I_t$ ,  $d_{2i} = -\sum W^2 I_y I_t$  et la matrice  $G \in \mathbb{R}^{2N \times 2N}$  :

$$G_{2i-1, 2i-1} = \sum W^2 I_x^2, \quad G_{2i, 2i} = \sum W^2 I_y^2, \\ G_{2i-1, 2i} = G_{2i, 2i-1} = \sum W^2 I_x I_y$$

On obtient le nouveau système :

$$GV = d \quad (23)$$

La nouvelle matrice  $G$  est de taille  $2N \times 2N$  dont les éléments de la diagonale sont les éléments de la matrice  $G$  de l'équation (14).

Motivé par la nature parallèle de calcul des réseaux de neurones, on a proposé d'éviter l'inversion de la matrice  $G$  en résolvant directement le système d'équations (23) par le réseau de neurones de Zhang.

#### B. Réseau de neurones de Zhang

Pour résoudre l'équation  $GV = d$  on utilise le réseau de Zhang en se basant sur la minimisation d'un vecteur-erreur de la forme [8,11] :

$$\epsilon(t) = GV - b \quad (24)$$

L'utilisation du gradient négatif de  $\epsilon(t)$  nous aboutit à :

$$G\dot{V}(t) = -\eta f(GV(t) - d) \quad (25)$$

avec  $\eta > 0$  est le paramètre de convergence,  $f(\cdot)$  dénote la fonction d'activation. Dans le présent travail on a utilisé la fonction signe-bi-puissance [12-13] :

$$f(x) = \frac{1}{2} \left( \text{sig}^r(x) + \text{sig}^{\frac{1}{r}}(x) \right), \quad 0 < r < 1 \quad (26)$$

$$\text{avec : } \text{sig}^r(x) = \begin{cases} |x|^r & \text{si } x > 0 \\ 0 & \text{si } x = 0 \\ -|x|^r & \text{si } x < 0 \end{cases}$$

Pour l'implémentation du réseau sur des calculateurs numérique, on exprime la dynamique du  $i^{\text{me}}$  neurone à partir de l'équation (25) sous la forme :

$$\dot{v}_i(t) = -\eta f \left( \sum_{j=1}^n g_{ij} v_j(t) - d_i \right) + (1 - g_{ii}) \dot{v}_i(t) - \sum_{j=1, j \neq i}^n g_{ij} \dot{v}_j(t) \quad (27)$$

puis on applique la formule d'Euler  $\dot{v}_i(t) = (v_i((k+1)\tau) - v_i(k\tau)) / \tau$  pour obtenir la version discrète du modèle de Zhang.  $k$  et  $\tau$  sont l'indice de l'itération et le pas de discrétisation respectivement. La Figure (4) présente l'architecture du réseau discret pour  $n = 2N$ .

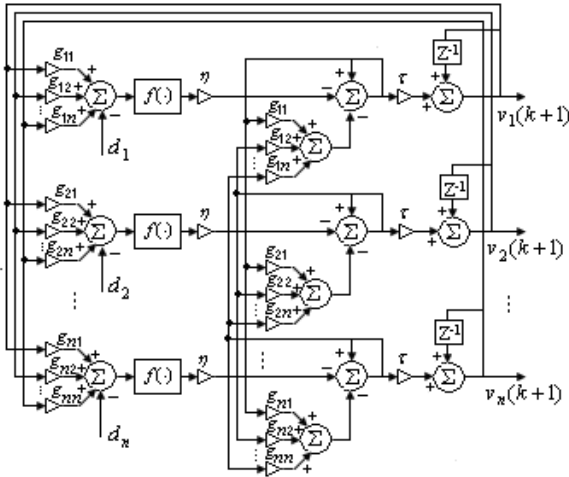


Figure 4. Architecture du réseau de neurones de Zhang discret.

## V. RESULTATS DES SIMULATIONS

La méthode proposée a été évaluée sur deux séquences d'images extraites de la récente base de données de Middlebury dont le flot optique réel est connu [14] : la séquence synthétique 'Yosemite' et la séquence réelle 'Hydrangea'.

L'erreur entre le flot optique des séquences artificielles et le mouvement estimé est mesurée par l'erreur angulaire moyenne EAM [12] :

$$EAM = \frac{1}{N.M} \sum_{i=1}^N \sum_{j=1}^M \arccos \left( \frac{u_c u_e + v_c v_e + 1}{\sqrt{(u_c^2 + v_c^2 + 1)(u_e^2 + v_e^2 + 1)}} \right) \quad (28)$$

où  $(u_c, v_c)$  et  $(u_e, v_e)$  désignent le flot optique vrai et le champ de déplacement estimé respectivement. Elle permet une analyse quantitative sur l'erreur d'estimation. La méthode d'estimation nécessite trois paramètres à ajuster : le nombre de niveaux  $K$ , le nombre d'itérations dans chaque niveau et la taille de la fenêtre d'analyse. Ces paramètres sont choisis de façon à minimiser l'erreur angulaire moyenne EAM. On fixe le nombre de niveaux à 3, le taille de la fenêtre à  $15 \times 15$  et le nombre d'itération à 3.

La figure (5) présente les courbes de convergence du réseau de Zhang pour la séquence 'Yosemite'. On remarque

que le réseau converge dans une trentaine d'itérations pour les trois niveaux et les trois itérations de la méthode de LK.

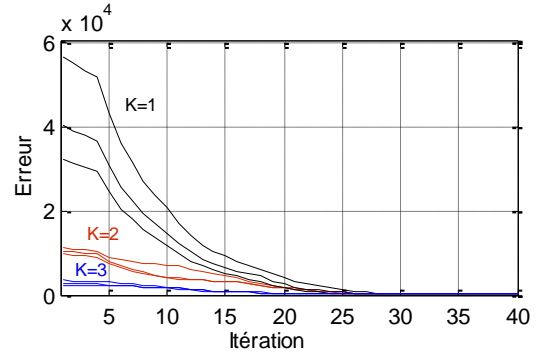


Figure 5. Courbes de convergence du réseau de Zhang pour  $\eta = 0.5$  et  $\tau = 1$ .

La figure (6) montre la performance de la méthode proposée pour la séquence 'Yosemite'. Dans le premier test, on a pris la séquence telle quelle est (fig.6.a), on estime son flot optique sans débruitage puis on lui applique un pré-filtrage par la transformée en ondelettes et par la transformée en curvelettes. Les figures 6.d,e,f montrent le flot estimé de la séquence sans filtrage, avec filtrage par TC et avec filtrage par TO.

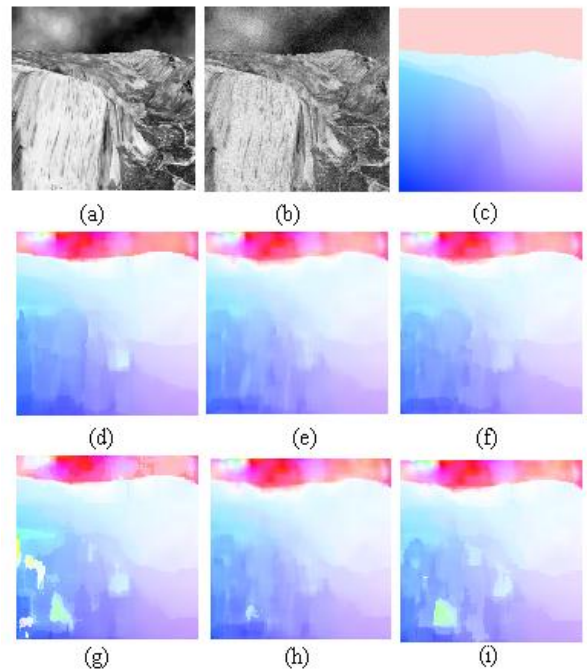


Figure 6. Flots optiques pour la séquence Yosemite. (a) frame 11, (b) frame bruitée (1%), (c) flot optique réel, (d,e,f), flots estimés sans débruitage, avec débruitage par la TC et la TO respectivement, (g,h,i) flots estimés sans débruitage, avec débruitage par la TC et la TO respectivement de la séquence bruitée (1%).

Dans le deuxième test, on répète la même procédure mais avec la séquence bruitée (ajout d'un bruit de 1%). Les figures 6.g, h, i montrent le flot estimé de la séquence sans filtrage, avec filtrage par curvelettes et avec filtrage par ondelettes.

Dans les deux tests on remarque que le flot estimé après un prétraitement par la transformée en curvelettes est meilleur que le flot estimé sans prétraitement aussi bien qu'après un prétraitement par la transformée en ondelettes. Les figures 7 et 8 présentent l'erreur angulaire moyenne pour différents bruits pour les séquences Yosemite et Hydrangea. Elles prouvent l'efficacité du prétraitement par la transformée en curvelettes pour l'estimation du mouvement.

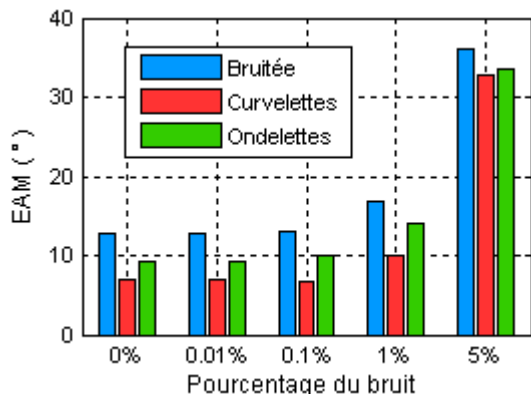


Figure 7. Erreur angulaire de la séquence 'Yosemite'

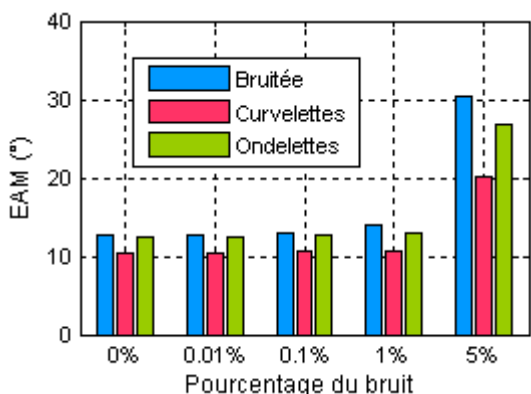


Figure 8. Erreur angulaire de la séquence Hydrangea

## VI. CONCLUSION

Dans cet article on a proposé deux outils pour l'estimation de mouvement multi-échelles de Lucas et Kanade dans une séquence d'images ; un réseau de neurones pour le calcul du flot optique et la transformée en curvelettes pour débruiter les images. La méthode proposée offre deux avantages: la rapidité d'exécution et la bonne précision des résultats numériques. Les simulations faites sur des séquences artificielles et naturelles montrent que le réseau de Zhang et la transformée en curvelettes contribuent à une nette amélioration des résultats. Suite à ces bonnes performances, l'approche proposée est très sollicitée pour les applications en temps réel citant entre autre la reconnaissance audio-visuelle, la médecine et la robotique.

## REFERENCES

[1] J.L Barron, D.J.Fleet, S.S. Beauchemin, T.A Burkitt, "Performance of Optical Flow Techniques", *Department of Computer Science, University of Western Ontario*, July 1993.

[2] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *In Proceedings of the 7th International Joint Conference on Artificial Intelligence*, pp. 674-679, 1981.

[3] J.Y. Bouguet, "Pyramidal implementation of the Lucas Kanade feature tracker," *Open CV documentation, Intel Corporation, Microprocessor Research Labs*, 1999.

[4] S.N.Tamgade, V. R.Bora, "Motion vector estimation of video image by Pyramidal implementation of Lucas Kanade Optical flow," *Second International Conference on Emerging Trends in Engineering and Technology, ICETET-09*, pp. 914-917, 2009.

[5] J.L. Starck, E.J. Candès, D.L. Donoho, "The curvelet transform for image denoising," *IEEE trans. on image processing*, vol 11, pp. 670-684, 2002.

[6] Hilton M. L., Jawerth B. D, et Sengupta A., "Compressing still and moving images with wavelets", *Multimedia systems*, vol. 2, pp.218-227, 1994.

[7] Marcellin M. W., Gormish M. J., Biling. A et Boliek M. P., "An overview of JPEG 2000", *Data Compression Conference*, pp. 523-544, 2000.

[8] C. Yi, Y. Zhang, "Analogue recurrent neural network for linear algebraic equation solving," *Electr. Lett.*, 2008, Vol 44 (18), pp. 1078-1079.

[9] E. J. Candès, L. Demanet, D. L. Donoho, L. Ying, "Fast discrete curvelet transforms," *Multiscale Model. Simu.*, vol 5 (3), pp.861-899, 2006.

[10] L. Tessens, A. Pižurica, A. Alecu, A. Munteanu, and W. Philips, "Context adaptive image denoising through modelling of curvelet domain statistics," *Journal of Electronic Imaging*, vol 17, pp.1-17, 2008.

[11] A. Benchabane, A. Bennia, F. Charif, A. Taleb-Ahmed, (2013). Multi-Dimensional Capon Spectral Estimation Using Discrete Zhang Neural Networks. *Multidim Syst Sign Process* vol 24, 583-598.

[12] S. Li, S. Chen, B. Liu, "Accelerating a recurrent neural network to finite-time convergence for solving time-varying Sylvester equation by using a sign-bi-power activation function," *Neural Process. Lett.* Vol 37 (2), (2013), pp. 189-205.

[13] S. Li, Y. Li, Z. Wang, "A class of finite-time dual neural networks for solving quadratic programming problems and its k-winners-take-all application," *Neural Netw.* vol 39, 2013, pp. 27-39.

[14] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical Flow," *Microsoft Research, Technical report MSR-TR-2009-179*, 2009. <http://vision.middlebury.edu/flow/>.