

Calcul de l'Estimation du Mouvement Intégré dans un Capteur d'Images CMOS

David Navarro, Guy Cathébras

► **To cite this version:**

David Navarro, Guy Cathébras. Calcul de l'Estimation du Mouvement Intégré dans un Capteur d'Images CMOS. JNRDM: Journées Nationales du Réseau Doctoral de Microélectronique, 2003, Toulouse, France. pp.489-491. lirmm-00191960

HAL Id: lirmm-00191960

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00191960>

Submitted on 26 Nov 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Calcul de l'estimation du mouvement intégré dans un capteur d'images CMOS

David Navarro
LIRMM
161, rue Ada
34392 Montpellier

Guy Cathébras
LIRMM
161, rue Ada
34392 Montpellier

E-mail : . navarro@lirmm.fr

Résumé

Ce papier présente l'intégration d'un traitement d'image, l'estimation du mouvement, dans un capteur d'images CMOS. Cette estimation a pour but de générer un champ de vecteurs représentant le mouvement de tous les pixels entre deux images acquises successivement. Ce traitement d'image a déjà été mis en œuvre, informatiquement [3], ou à l'aide de FPGAs [2], en utilisant des caméras standards. Nous présentons ici une architecture intégrant ce calcul au sein même du capteur d'images.

1. Introduction

L'émergence des technologies CMOS fortement sub-microniques a permis aux capteurs d'images CMOS de concurrencer les produits en technologie CCD. En effet, ces nouveaux capteurs atteignent des résolutions suffisantes pour certaines applications, ont des coûts de fabrication bien moins élevés, et permettent de placer de l'électronique dans le capteur lui-même. L'intégration de transistors au sein du capteur a donné naissance aux rétines CMOS, par exemple [4], qui sont des capteurs d'images 'intelligents' traitant l'image dès l'acquisition. Une rétine permettant d'estimer un mouvement entre deux images est présentée ici.

2. Capteur d'images

2.1 Capteurs d'images CCD

Dans un capteur d'images (ou imageur) CCD, les charges électriques créées par les photons dans les zones photosensibles (des diodes) sont acheminées vers un amplificateur de lecture par un registre à décalage analogique. C'est ce dernier qui donne son nom à la technologie : un registre CCD (Charge Coupled Devices) est une succession de capacités MOS (Métal - Oxyde - Semi-conducteur) commandées par trois signaux d'horloge [1] [2] [3].

Ces capteurs ont été les précurseurs dans le domaine de l'acquisition d'images à base de circuit intégré. Les premiers ont été développés à partir de 1969 et leur commercialisation, par Sony, date de 1981. Leur fort développement a alors permis d'atteindre une grande

qualité et une grande maturité de fabrication. Ce type de capteur est maintenant considéré comme la référence en imagerie numérique. A ce jour, les capteurs CCD ont un excellent rendement lors de la transduction photons-électrons (beaucoup de caméras CCD peuvent fonctionner avec des éclairagements inférieurs au lux), ainsi qu'un très faible taux de pertes lors des décalages des charges analogiques.

Cependant, ces décalages très nombreux induisent une consommation importante. De plus, le schéma de lecture de l'image est figé et on est obligé de la lire en totalité avant de lire l'image suivante, ce qui interdit des cadences d'acquisition élevées, sauf au prix d'artifices [1]. De plus, les caractéristiques d'une technologie CCD sont contradictoires avec ceux d'une technologie CMOS : il n'est donc pas possible de produire des imageurs CCD et des circuits intégrés classiques sur les mêmes chaînes de fabrication. Ceci explique le coût encore élevé de ces imageurs.

2.2 Capteurs d'images CMOS

La technologie CMOS, bien que très ancienne (1963), n'a réussi à trouver sa place sur le marché de l'imagerie que vers la fin des années 90. Ceci s'explique principalement par leur 'légendaire' faible résolution causée par d'imposants pixels.

La forte densité d'intégration, due à l'arrivée des technologies CMOS sub-microniques, a permis d'atteindre des pixels de quelques micromètres de côté, de taille équivalente aux pixels des capteurs CCD, et donc des résolutions, enfin intéressantes, de plusieurs millions de pixels. Des produits tels que des web-cams ou des appareils photographiques existent déjà avec de tels composants. Les avantages de cette technologie sont la rapidité, la faible puissance consommée, et la possibilité, pour des fabricants de semi-conducteurs ne souhaitant pas développer des lignes de fabrication CCD, de pénétrer le marché des imageurs.

En plus de l'accès aléatoire aux pixels qui permet d'acquérir des « zones d'intérêt » à des cadences plus élevées que l'image entière, le fait d'utiliser une technologie standard permet aussi d'ajouter au sein même du capteur des fonctionnalités de traitement d'image. Ces traitements d'image sont dits de bas niveau car ils ont pour échelle le pixel et sont le plus souvent destinés à alléger les traitements ultérieurs. L'information en sortie du capteur

peut être le résultat de la corrélation entre deux images [7], les contours ou une information sur les mouvements à l'intérieur de la scène observée. Les composants intégrant un traitement au même niveau que le capteur sont appelés rétines, par analogie avec les « capteurs d'images » biologiques qui effectuent, eux-aussi, un traitement de bas niveau avant de transmettre les informations au cortex...

Nous nous intéressons aux applications d'estimation de mouvement. Actuellement, les systèmes utilisés nécessitent une architecture composée d'un capteur d'image standard et d'un périphérique de traitement d'image : processeur ou composant programmable. Les quelques circuits dont les architectures sont intégrées au niveau du capteur sont généralement adaptés à des scènes fortement contrastées ou faiblement texturées. Notre méthode offre une approche nouvelle puisqu'elle permet de traiter des images complexes, comme les scènes réelles, en utilisant un algorithme performant déjà éprouvé [6]. Nous allons détailler cet algorithme puis notre architecture.

3. Algorithme d'estimation du mouvement

Ce traitement d'image [5] est composé de deux phases : une première phase de codage des pixels à partir de la structure locale de l'image, et une phase d'appariement de ces codes sur deux images successives.

La phase de codage sert à obtenir, pour chaque pixel, un code binaire sur 8 bits qui caractérise sa luminosité par rapport à celle de ses voisins. Pour cela, chaque pixel est considéré successivement comme pixel central, puis sa luminosité, c'est à dire la valeur du niveau de gris, est comparée à son voisinage le plus proche. Si le niveau de gris d'un voisin est inférieur à celui du pixel central, ce voisin est noté '1', sinon '0'. Cette comparaison est faite sur les 8 pixels voisins, ce qui code le pixel central sur 8 bits, comme le montre la figure 1. Ainsi, chaque pixel est remplacé par un codage de la texture du voisinage dont il est le centre.

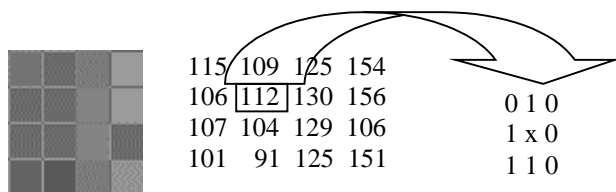


Figure 1 : Image, niveaux de gris et calcul du code du pixel '112'

Par exemple, la valeur du niveau de gris du premier voisin, 115, est comparée à celle du pixel central, 112, ce qui donne '0' car 112 est inférieur à 115; de même pour les autres valeurs. Le pixel 112, en position centrale, prend donc la valeur '01010110' en binaire ou 56 en hexadécimal. Tous les pixels subissent cette transformation. Nous devinons ici un des avantages de cette transformée : elle est peu sensible aux changements de luminosité. Cette étape est répétée sur la deuxième image, de sorte que tous les pixels des deux images soient codés sur 8 bits. Le résultat du codage - en hexadécimal - des deux images est représenté figure 2.

Une fois cette phase de codage des pixels terminée, chaque code d'un pixel de la première image est recherché

dans un voisinage dans la deuxième image. Une distance de Hamming (nombre de bits différents entre deux mots

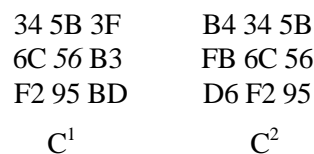


Figure 2 : Codes des pixels dans deux images successives

binaires) est calculée entre les codes et la distance la plus petite, normalement 0, détermine le déplacement réel du pixel d'une image à l'autre. Cette distance nulle correspond à un code de la première image qui a été retrouvé dans la deuxième image. Dans l'exemple de la figure 2, le pixel codé 56 dans les codes C¹ de la première image se retrouve dans C², en traduisant un mouvement d'un pixel vers la droite. Cette méthode, fortement itérative, donne un champ de vecteurs dense, dont la représentation est celle du mouvement (figure 3).

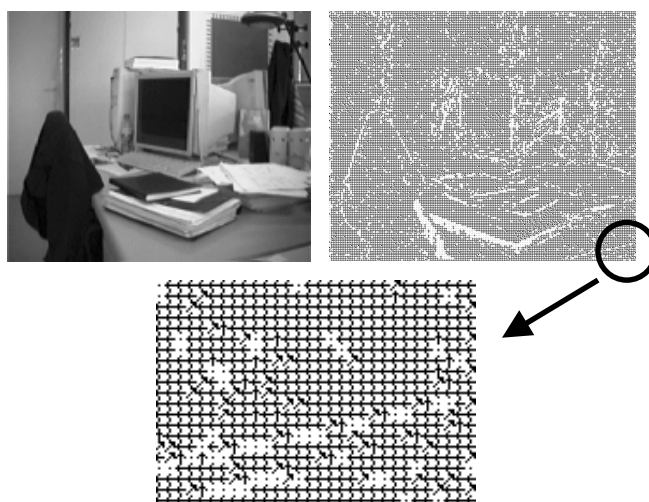


Figure 3 : Image de la séquence et champ de vecteurs denses

L'information résultante est donc, pour chaque pixel, un vecteur indiquant le déplacement de celui-ci entre l'image présentée figure 3 et l'image suivante dans la séquence.

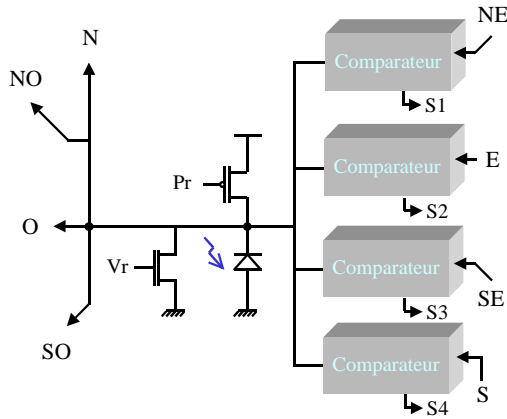
Alors que les rétines CMOS actuelles s'appuient sur des méthodes peu adaptées aux scènes réelles, nous présentons ici l'implantation de cet algorithme de façon électronique.

4. Intégration dans un capteur

Le photorécepteur le plus classiquement utilisé, auquel nous nous intéressons, est composé d'une photodiode et d'un transistor de commande. La tension de sortie, prise sur le fil reliant la photodiode au transistor, est une droite dont la pente est proportionnelle à la puissance lumineuse reçue. Plus la photodiode est éclairée, plus la pente est importante. Chaque pixel d'une image peut avoir une luminosité différente, ce qui entraîne un photocourant et donc une

décharge propre à chaque pixel. C'est de ce montage que découle notre architecture : les tensions électriques sont comparées directement pour effectuer le codage des pixels.

Ce bloc est situé au sein même du pixel, et sert à comparer deux valeurs de luminosité. Son schéma bloc est montré dans la figure 4. Chaque pixel est relié à ses huit voisins (représentés par les directions cardinales). Pour des raisons de symétries évidentes (A-B est l'opposé de B-A), le nombre de comparateurs est réduit à quatre, sans perte d'information.



Ces comparaisons se font de manière quasi-instantanée, le circuit mémorise alors les codes des images.

Ces codes, présents sur les sorties S1 à S4, sont ensuite traités par une unité de calcul spécifique qui génère l'information du mouvement. Un tel bloc a été développé (en VHDL) afin d'effectuer la recherche des codes à travers deux images. Ce bloc, placé en bas de la matrice de pixels, effectue un calcul de manière parallèle, il traite ainsi tous les pixels d'une ligne en même temps. La figure 5 montre ces éléments de calculs parallèles.

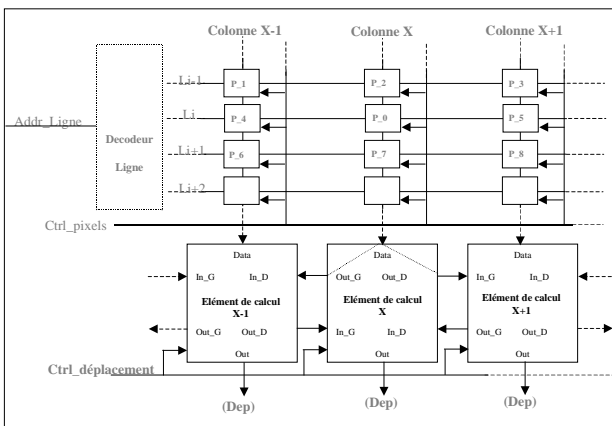


Figure 5 : Eléments de calcul traitant les informations issues de la matrice de codes

Ce calcul consiste en la recherche du code minimal de chaque pixel de la première image dans une fenêtre de pixels de la deuxième image. Alors que cette étape est très itérative pour un élément de calcul classique, la parallélisation des données permet ici un calcul en quelques microsecondes. La recherche de ces codes donne les caractéristiques des vecteurs.

5. Conclusion

Un algorithme d'estimation du mouvement, basé sur une méthode de mise en correspondance de blocs permettant le traitement d'images fortement texturées, a été implanté dans un capteurs d'images spécifique en technologie AMS 0,35 μm . La simulation a permis de valider les différentes structures, et les mesures sont actuellement menées pour confirmer la fonctionnalité des différentes architectures de calcul électronique.

Références

- [1] W.F. Kosonocky, et al. *360 × 360 element three-phase very high frame rate burst image sensor: design, operation and performance*. IEEE Transactions on Electron Devices, vol. 44, pp. 1617 –1624, October 1997.
- [2] K. Itakura, et al. *A 2/3-in 2.0 M-pixel CCD imager with an advanced M-FIT architecture capable of progressive scan*. IEEE Transactions on Electron Devices, vol. 44, pp. 1625 - 1632, October 1997.
- [3] T. Yamada, et al. *A progressive scan CCD image sensor for DSC applications*. IEEE Journal of Solid-State Circuits, vol. 35, pp. 2044 - 2054, December 2000.
- [4] D. Scheffer, B. Dierickx, G. Meynants. *Random addressable 2048 × 2048 active pixel image sensor*. IEEE Transactions on Electron Devices, vol. 44, pp. 1716 -1720, October 1997.
- [5] R. Zabih, J. Woodfill. *Non-Parametric Local Transforms For Computing Visual Correspondance*. Proceedings of the 3 rd European Conference on Computer Vision, pp 151-158, 1994.
- [6] J. Woodfill, B. Von Herzen. *Real-Time Stereo Vision on the PARTS Reconfigurable Computer*. IEEE Symposium on Field-Programmable Custom Computing Machines, pp 242-250, 1997.
- [7] G. Cathebras et al. *A Continuous Time Pattern Recognition Retina*. Proceedings of the 28 th European Solid-State Circuits Conference, pp. 719-722, September 2002.