

## CARTE PROCESSEUR PROGRAMMABLE DE VOISINAGE

 Jean Michel MATHE Christine LEMPEREUR  
 Stéphane MARQUERIE

ONERA CERT DERMES 2 Avenue Edouard Belin 31055 TOULOUSE CEDEX

## RÉSUMÉ

Cet article décrit la conception d'un dispositif d'imagerie temps réel qui permet d'effectuer des traitements programmables dans un environnement 3 x 3. L'intérêt d'un tel système pour les visualisations en Mécanique des Fluides y est développé.

## 1. INTRODUCTION

La visualisation qualitative et quantitative des écoulements est un moyen expérimental largement utilisé dans les études fondamentales en Mécanique des Fluides. Dans un premier temps, elle illustre la connaissance intuitive que l'on peut avoir des phénomènes physiques. L'analyse dynamique des paramètres densitométriques ou géométriques de l'image permet ensuite de produire des valeurs numériques servant de conditions aux limites ou de validation pour des modélisations mathématiques.

Le dispositif décrit dans cet article a été conçu pour répondre à ces besoins d'imagerie temps réel d'écoulements mais son utilisation peut être extrapolée à de nombreux domaines d'applications.

Les mesures à réaliser sur ce type d'images sont de deux natures:

densitométriques: dans ce cas, il s'agit de quantifier une densité locale en chacun des points du plan de mesure (écoulements essentiellement bidimensionnels). Pour ce faire, il est nécessaire de réaliser un étalonnage de la caméra, et de corriger chaque pixel pour restituer la mesure en grandeur physique. De manière générale, les traitements consistent à effectuer des opérations arithmétiques entre images (correction, sommation pour l'obtention de la moyenne, ...).

géométriques: les informations recherchées (aire, barycentre,...) nécessitent la détection d'entités caractérisant l'écoulement (traceurs particuliers ou continus), dont l'évolution au cours du temps permet de remonter à la dynamique du phénomène physique observé. Le premier traitement envisageable consiste à seuiller le signal pour travailler sur une image binaire. En fait, cette approche n'est possible que si l'expérimentation permet de créer un contraste de qualité, condition rarement réalisable.

## ABSTRACT

This article describes the design of a real time imaging system which allows programmable processings to be carried out in 3\*3 window. The interest of such a system for flow visualisation is discussed in the following summary.

Il faut alors développer des traitements plus élaborés pour détecter ces entités géométriques. L'extraction de contours est le moyen le plus utilisé: Il existe une importante bibliographie sur ce sujet qui permet d'adapter l'opérateur aux caractéristiques de l'image à traiter. Dans ces conditions, il est intéressant de créer un dispositif susceptible d'émuler les divers opérateurs de détection de contours qui travaillent, pour la plupart, dans un environnement 3\*3.

L'analyse de la dynamique des phénomènes nécessite de traiter un ensemble d'images; on peut alors enregistrer le signal sur un magnétoscope puis le relire et le traiter image par image. Cette solution n'est qu'un palliatif en raison des problèmes d'imprécision spatiale entraînée par l'arrêt sur image (jitter).

Il est donc préférable de traiter et d'acquérir l'information en **temps réel**, c'est-à-dire dès qu'elle se présente. Les systèmes vidéo fonctionnant à 50 Hz, il faut traiter 10 Millions de pixels par seconde (pour une résolution spatiale typique 512\*512 pixels), ce qui suppose la mise en oeuvre de moyens très spécifiques.

La sauvegarde des données, à des fins de traitements différés, s'effectue après la mise en oeuvre des primitives de bas niveau qui permettent de compacter l'information contenue dans l'image. Nous ne parlerons ici que de l'analyse géométrique: Ce même processus peut s'appliquer à l'analyse densitométrique, mais le compactage n'est plus réalisé par des primitives de bas niveau.

Ces spécifications nous ont conduits à réaliser un dispositif temps réel, intégré à un système de vision, capable d'effectuer des opérations de traitements d'images dans un voisinage 3\*3; nous verrons qu'il peut être étendu à une connectivité supérieure, et peut traiter aussi bien des données entières que flottantes avec une puissance de calcul importante.



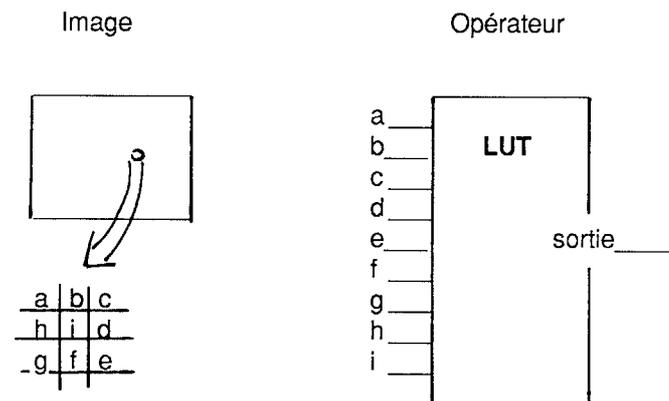
## 2. DISPOSITIF DE TRAITEMENT

### 2.1 Traitement dans un voisinage 3\*3

On peut considérer une image numérisée comme une matrice  $L(x,y)$  de valeurs représentatives de la luminance en chaque point. Tout naturellement, l'analyse géométrique des entités nécessite de traiter un pixel par rapport à son environnement immédiat. De fait, la bibliographie propose un ensemble d'algorithmes qui s'intéressent à un pixel et ses huit voisins (éventuellement six dans le cas d'un canevas hexagonal). Cette matrice, où sont sauvegardés les pixels, est rangée dans la mémoire centrale du système. Le processeur central peut réaliser tout type de traitements, mais les puissances de calcul actuelles ne permettent pas encore d'aborder le temps réel pour des coûts acceptables.

Il s'agit donc de développer un module temps réel, interfaçable avec un système de vision, capable de traiter un environnement minimum 3\*3, à la fréquence vidéo de 10 MHz, supportant les primitives de base les plus usitées (Sobel, Roberts, Morphologie Mathématique,...).

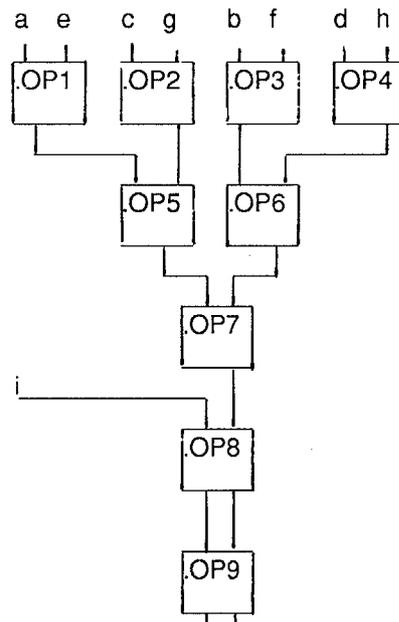
Ce module reçoit les entrées composées du pixel central et de ses voisins. L'opérateur de calcul qui gère ces neuf données peut être distribué sous la forme d'un ensemble de processeurs traitant chacun deux données. Plusieurs architectures peuvent être utilisées. Par exemple, dans le cas d'une structure arborescente, chaque noeud du système représente un processeur de calcul particulier. Un circuit spécialisé pourrait assurer les opérations arithmétiques et logiques élémentaires, mais la nature des opérateurs serait trop limitée pour pouvoir coder tous les algorithmes souhaités.



Dans le cas d'un environnement 3\*3 de pixels codés sur 8 bits, la capacité nécessaire pour coder toutes les configurations s'élève à  $2^{72}$  octets opérant avec des temps d'accès inférieurs à 100 ns!!!

Il est donc nécessaire de distribuer le traitement dans un ensemble de LUTs organisé en un canevas particulier. Tenant compte des algorithmes qui existent en imagerie, une structure arborescente qui privilégie les médianes et diagonales semble bien adaptée.

Le module est réalisé au moyen de circuits RAM de 32Ko à temps d'accès de 40ns. La capacité réduite de ces circuits permet de traiter des pixels codés sur 7 bits seulement. Le résultat est présenté sur 8 bits, dont sept seulement sont transmis sur l'étage suivant. En raison du temps d'accès des RAM et du 15<sup>ième</sup> bit disponible (adressant 16 Ko), nous avons pu loger deux traitements par cycle d'horloge pixel dont la période est de 100ns. Le dernier étage effectue une opération entre les deux résultats.



Nous pouvons résumer le fonctionnement du module: il se décompose en deux phases:

- 1/ *initialisation des contenus de chaque RAM.* En raison des opérations d'entiers réalisées, il est indispensable d'effectuer un calcul tenant compte d'un facteur d'échelle pour éviter des non-linéarités.

- 2/ *mise en oeuvre du calcul.* Un dispositif amont, constitué de lignes à retard, crée l'environnement 3x3 à appliquer au réseau de LUTs. Chaque étage introduit un retard d'un demi-cycle d'horloge. Le système génère un délai de trois périodes, auquel on doit ajouter une ligne et un pixel pour tenir compte du dispositif de création de la fenêtre 3\*3.

Les systèmes de traitement d'images temps réel utilisent souvent des **Look Up Tables (LUT)** pour modifier le signal numérisé. Ce procédé permet de programmer une fonction de transfert mono-dimensionnelle quelconque et de l'appliquer à un signal numérique. On peut aussi augmenter le nombre d'entrées et créer une fonction de transfert multi-dimensionnelle à une sortie. La capacité de la mémoire LUT est fonction du nombre d'entrées et de leur dynamique.

## 2.2 Environnement

Le dispositif réalisé s'intègre dans un système d'imagerie constitué d'un calculateur MOTOROLA SYS133 à 25 MHz, structuré autour d'un bus VME sous VERSADos. Des cartes de la famille DATA-CUBE effectuent l'acquisition, la mémorisation et la visualisation en temps réel des images par l'intermédiaire d'un bus vidéo spécifique: le Max-Bus. L'échange d'informations entre les divers modules constituant le système vidéo est indépendant du bus VME, et les débits sont essentiellement fonction du type de cartes utilisées et du nombre des liaisons inter-cartes. Typiquement, et à titre d'illustration, un système de six cartes DATA-CUBE peut véhiculer plus de 120 Mo/s.

Comme nous l'avons souligné, l'analyse dynamique de séquence d'images suppose un compactage d'informations aux moyens de primitives de bas niveau. Les opérateurs d'extraction de contours, de type SOBEL, LAPLACIEN ou Morphologie Mathématique nécessitent plusieurs dizaines, voire centaines de millions d'opérations par seconde. Par exemple, l'opérateur de SOBEL:

$$\begin{aligned} X &= (a + 2b + c) - (e + 2f + g) \\ Y &= (c + 2d + e) - (a + 2h + g) \\ S &= (X^2 + Y^2)^{1/2} \end{aligned}$$

nécessite 180 Mopérations/seconde.

Les opérations se répartissent suivant les différents étages:

OPnt:

n spécifie l'une des neuf LUTs,  
t indique le cycle dans la période

### premier étage

demi-cycle1                      demi-cycle 2

$$\begin{aligned} \text{OP11} &= a - e & \text{OP12} &= e - a \\ \text{OP21} &= c - g & \text{OP22} &= c - g \\ \text{OP31} &= b - f & \text{OP32} &= 0 \\ \text{OP41} &= 0 & \text{OP42} &= d - h \end{aligned}$$

### deuxième étage

$$\begin{aligned} \text{OP51} &= \text{op11} + \text{op21} & \text{OP52} &= \text{op12} + \text{op22} \\ \text{OP61} &= 2 * \text{op31} & \text{OP62} &= 2 * \text{op42} \end{aligned}$$

### troisième étage

$$\text{OP71} = \text{op51} + \text{op61} \qquad \text{OP72} = \text{op52} + \text{op62}$$

### quatrième étage

$$\text{OP81} = \text{op71} * \text{op71} \qquad \text{OP82} = \text{op72} * \text{op72}$$

### cinquième étage

$$\text{OP91} = (\text{op81} + \text{op82})^{1/2} \qquad \text{OP92} = ?$$

OP92 peut être programmé comme OP91 par exemple, mais intégrer aussi un seuillage.

## 3. CONCLUSIONS

Ce module, de par sa versatilité, permet de coder un grand nombre d'algorithmes utilisés en traitement d'images. Il est possible d'associer plusieurs cartes identiques pour accroître la résolution et traiter des fenêtres de dimension  $3n \times 3n$ .

Il est par ailleurs envisageable d'augmenter le nombre de traitements par cycle d'horloge pixel en mettant en oeuvre des mémoires plus rapides, par exemple trois opérations avec des temps d'accès de 20ns.

La troisième évolution a trait à la dynamique du signal. Pour la maquette développée, les données sont codées sur 7 bits: Une densité plus importante de circuits mémoire devrait permettre d'augmenter la précision du calcul, en ayant un codage intermédiaire sur une dizaine de bits.

L'outil actuel, intégré dans une chaîne de vision DATA-CUBE, est un auxiliaire précieux pour la compréhension des phénomènes étudiés en mécanique des fluides. Les performances "temps réel" peuvent s'avérer très intéressantes pour de nombreux autres domaines d'applications industrielles (robotique, contrôle qualité,...) ou médicales par exemple.