

XIV CONGRES DE LA SOCIETE INTERNATIONALE DE PHOTOGRAMMETRIE

G.III/2 — TRAITEMENT DE L'IMAGE

PROPAL II ET LE TRAITEMENT D'IMAGES

Serge BRACONNE
CIMSA, 10-12 Avenue de l'Europe - 78140 VELIZY
(FRANCE)

SOMMAIRE

- 1 - Introduction
- 2 - Système PROPAL II et Système de traitement et visualisation d'images :
conséquences architecturales
- 3 - Considérations méthodologiques et applications
 - 3.1. Gestion des Transferts
 - 3.2. Structuration des données dans la mémoire de stockage M2
 - 3.3. Utilisation de la mémoire M1 à des fins algorithmiques
- 4 - Conclusions
- 5 - Références

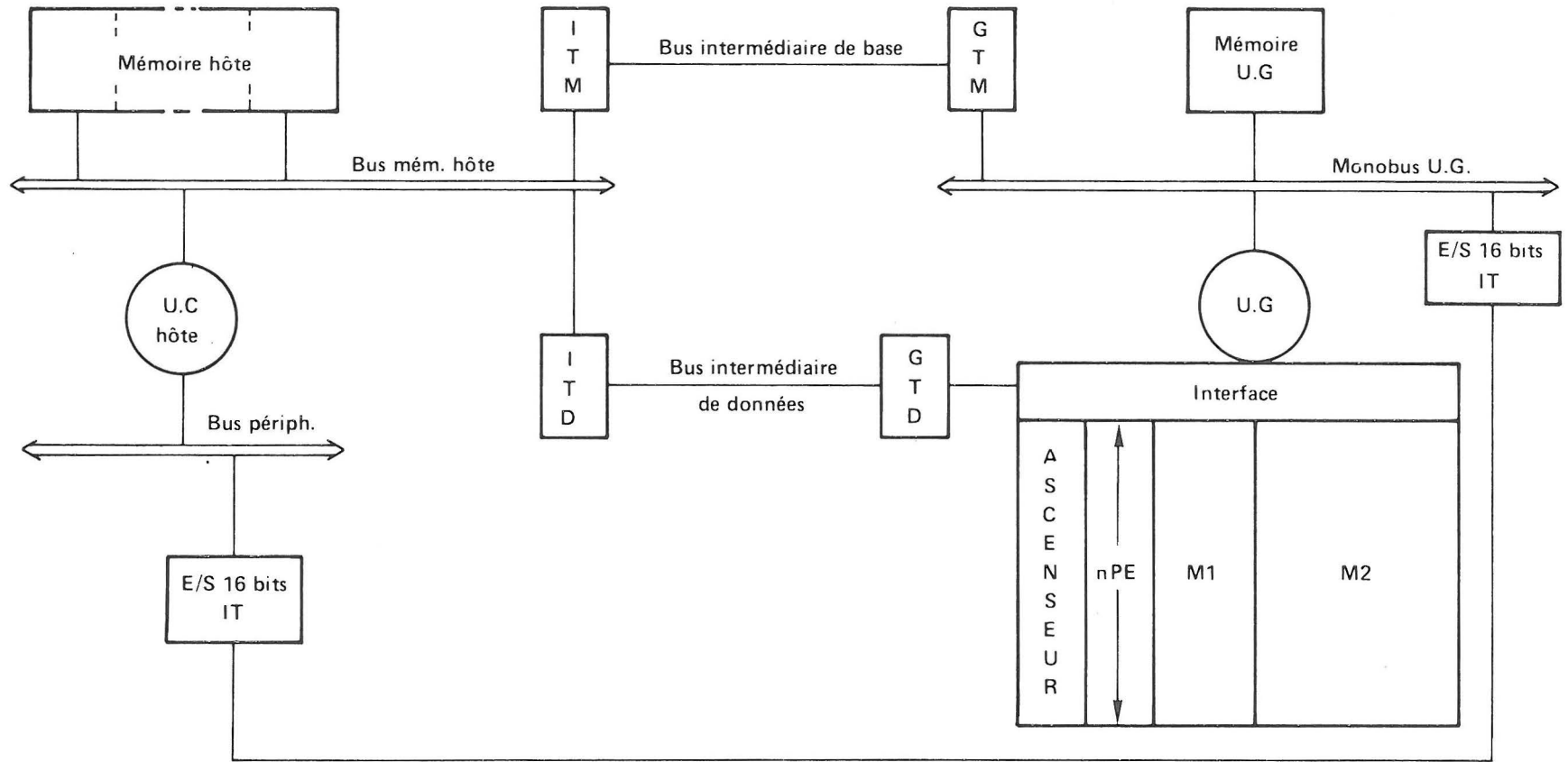
1 - INTRODUCTION

Très tôt (dès 1958 d'après [1]) l'inadaptation des ordinateurs conventionnels au traitement d'images est apparue, en effet, ce dernier est caractérisé souvent par un processus répétitif, combinaison d'opérations simples, au niveau du pixel ou du groupe de pixels (voisinages). D'où l'idée d'introduire la notion de parallélisme dans l'architecture des ordinateurs : divers prototypes ont été réalisés ou tentés de l'être (voir l'article de synthèse [1]) ; la phase de réalisation n'a pas toujours vu le jour compte tenu des difficultés théoriques et technologiques qui sont apparues après les premières études. La plupart de ces prototypes permettent d'accélérer l'exécution des traitements au niveau d'une fenêtre de taille réduite (32 x 32 par exemple).

Les premières applications développées sur PROPAL II ont concerné le traitement d'images (prétraitement d'images satellites [2]), images médicales, reconnaissance de caractères ou d'empreintes digitales, etc.) ; ces applications étaient purement calculatoires, et il est apparu rapidement que le parallélisme pouvait également être utilisé pour réaliser un système de traitement et visualisation véritablement temps réel, exploitant pleinement la puissance de calcul offerte par PROPAL II.

Nous commencerons par exposer les conséquences architecturales qu'implique un tel système avant de présenter quelques considérations méthodologiques (inspirées des applications précitées) concernant l'utilisation de PROPAL II en traitement d'images (structuration de données en mémoire, recouvrement des temps de calcul et temps de transfert).

FIGURE 1 – CONFIGURATION HOTE PROPAL II



Couplage
hôte propal II

PROPAL II
standard

ITM : interface transfert mémoire } partie spécifique à chaque hôte
ITD : interface transfert données }
GTM : gestion transfert mémoire
GTD : gestion transfert données

UG : unité de gestion de PROPAL II
PE : processeurs élémentaires (en nombre n)
M1 : mémoire rapide (256 bits par PE, 165 ns de cycle de base)
M2 : mémoire de stockage (16 Kbits par PE, 330 ns de cycle de base)
(l'ascenseur permet les communications entre PE)

2 - SYSTEME PROPAL II ET SYSTEME DE TRAITEMENT ET VISUALISATION D'IMAGES : CONSEQUENCES ARCHITECTURALES

PROPAL II a déjà été décrit plusieurs fois ,par exemple dans [3], pour plus de détails sur le système (matériel et logiciel) on se réfèrera aux manuels de présentation [4] exploitation [5] et référence [6] . Nous nous bornerons ici à fournir un schéma descriptif de la configuration hôte. PROPAL 2 (fig.1). L'hôte est a priori quelconque : les parties à développer pour chaque calculateur hôte sont l'Interface Transfert Mémoire (ITM) et l'Interface Transfert Données (ITD), le bus intermédiaire de base permet de transférer à la fois des programmes et des données (via l'unité de gestion), le bus intermédiaire de données ne peut transférer que des données (directement dans PROPAL II). Cette configuration est disponible actuellement sur MITRA 125 et le sera à la fin de l'année sur les gammes SOLAR et MITRA 225.

Le principe de la connexion à un système de visualisation et d'acquisition est donné à la figure 2. Cette connexion est rendue possible grâce au bus d'entrées-sorties directes de l'ascenseur dont le débit peut être adapté en utilisant une horloge externe.

A titre indicatif calculons le temps de transfert T de 16 K bits XNPE (taille de M2) avec une cadence d'horloge de 0,150 μ s (horloge interne de PROPAL), il se décompose en 1024 transferts parallèles (de M2 dans l'ascenseur) de 16 XNPE bits prenant :

$$T1 = 1024 \times 16 \times 0,330 \mu s = 5406 \mu s$$

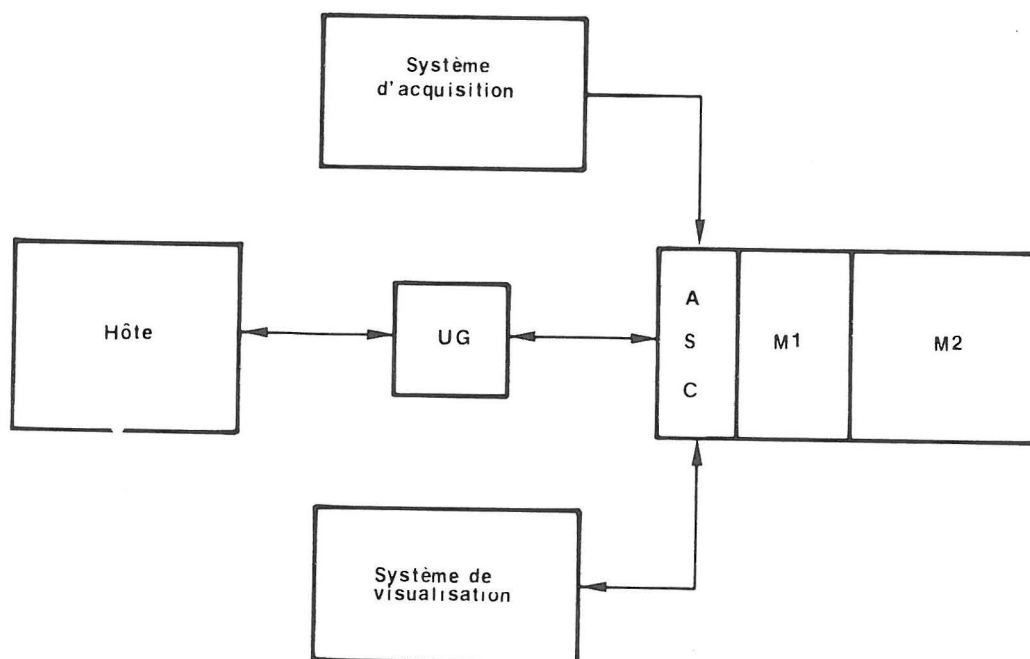
et en 1024 transferts constitués à chaque fois de NPE décalages d'ascenseur, soit

$$T2 = 1024 \times NPE \times 0,150 \mu s$$

On notera que T2, contrairement à T1 est dépendant du nombre de processeurs NPE. Si NPE = 512 on a T2 \approx 80 ms et T \approx 85 ms, à ces temps doivent s'ajouter ceux de l'initialisation des fonctions de transferts et de décalages, bref, on peut dire que le temps de transfert de la mémoire M2 de PROPAL dans la mémoire d'image MI du système de visualisation peut s'effectuer en environ 100 millisecondes.

L'utilisateur est entièrement maître de la structuration de M2 au niveau taille d'image et taille de pixel il est donc vraisemblable que les quantités d'information à transférer seront moins importantes, le nombre de P.E. doit néanmoins être suffisant : ainsi faire de l'imagerie $2n \times 2n$ paraît raisonnable.

Le système de visualisation est classique, c'est-à-dire que les signaux sortant de l'ascenseur vont dans une mémoire d'image M.I., puis dans le cas le plus simple, adressent des tables de correspondance couleurs (LUT) puis des convertisseurs analogiques digitaux (CAD) afin de pouvoir visualiser des images en fausses ou vraies couleurs, ce système étant approvisionné à partir de la mémoire M.I. il ne pénalise en rien la puissance de calcul de PROPAL II. Le système d'acquisition est également classique (par exemple caméras TV) et fournit un signal sur 16 bits en entrée directe sur l'ascenseur.



Le débit de l'ascenseur est de 100Mbits/s avec l'horloge interne et peut atteindre 300Mbits/s avec une horloge externe (avec dans les deux cas, un canal de 16 bits de large).

FIGURE 2 - SYSTEME DE VISUALISATION PROPAL II

3 - CONSIDERATIONS METHODOLOGIQUES ET APPLICATIONS

3.1. - GESTION DES TRANSFERTS

Lorsqu'on utilise un ordinateur rapide comme PROPAL et dans le cas où l'ordinateur hôte a l'initiative des transferts de données et PROPAL l'initiative des calculs, on doit éviter qu'un des deux calculateurs soit en attente de l'autre. Un tel problème a été étudié par S. TAHE [7] sur un exemple de filtrage 5 X 5. Il s'agit de recouvrir les temps d'entrée-sortie par des temps de calculs, bref de minimiser :

(temps de lecture de NBL lignes origines + temps d'écriture de NBL lignes filtrées) - (temps de filtrage de NBL lignes)

Schématiquement on a la représentation de la figure 3.

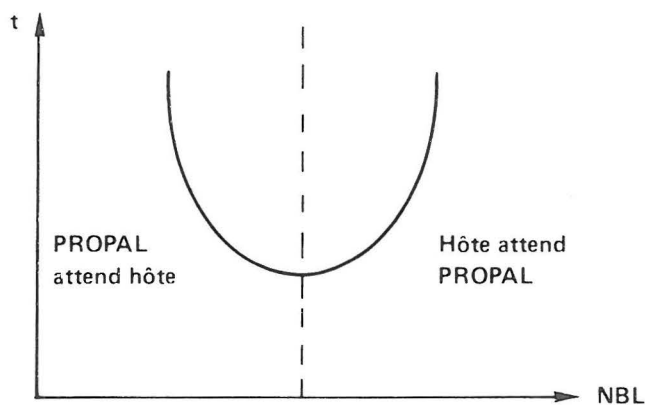


FIGURE 3

La solution est d'utiliser "deux tampons" dans la mémoire de l'unité de gestion, ainsi deux opérations se déroulent simultanément :

1 - Vidage et remplissage d'une mémoire tampon avec NBL lignes de l'image origine ou de l'image filtrée.

2 - Filtrage de NBL + 4 lignes de l'image origine contenues dans l'autre mémoire tampon, les NBL lignes de l'image filtrée construites étant rangées dans ce même tampon.

Le choix de NBL dépend des capacités des différentes mémoires de PROPAL ainsi que des caractéristiques du (ou des) disque (s) où sont stockées les images. Un tel filtrage sur une image 512 x 512 x 8 bits avec un PROPAL de 64 PE prend 50 s sans optimisation et 25 s avec, l'hôte seul (MITRA) prend 20 mn, un nombre de PE plus important améliorerait bien sûr considérablement ces performances.

3.2. - STRUCTURATION DES DONNEES DANS LA MEMOIRE DE STOCKAGE M2.

Plusieurs niveaux de structuration peuvent être distingués dans le traitement d'images : la structure la plus générale est la structure "bibliothèque images", elle est située sur une mémoire de masse (disques à grande capacité) et ne nous intéresse pas ici. De cette bibliothèque vont être extraites des images ou plutôt des sous-images (lorsque les volumes mis en jeu sont trop importants) qui vont être stockées dans un tampon de mémoire centrale dans le cas d'ordinateur classique, et dans la mémoire M2 dans celui de PROPAL.

On obtient ainsi un second niveau de structuration caractérisée par la taille maximale de l'image et les possibilités d'obtention rapide d'images (par échantillonnage de rétraction de fenêtre) ou de réorganisation (transposition par exemple). Le tableau de la figure 4 montre les possibilités de stockage d'images de taille standard (avec des pixels d'un octet) dans la mémoire M2 (16 Kbits) en fonction du nombre de processus élémentaires

Taille \ NPE	128 x 128 x 8 bits	256 x 256 x 8 bits	512 x 512 x 8 bits
128	16	4	1
256	32	8	2
512	64	16	4

FIGURE 4

Dans M2 une image est vue comme un ensemble de colonnes (Z_i) dont les adresses sont incrémentées de NB bits, NB étant la taille du pixel. Les possibilités d'extraction de sous-images par échantillonnage s'effectuent par positionnement des bascules D (échantillonnage ligne) et modification de la fréquence d'échantillonnage adresse (échantillonnage colonne). La réorganisation de données s'effectue par transfert $M2 \rightarrow M1 \rightarrow M2$, ou même par transfert $M2 \rightarrow M1$ hôte $M1 \rightarrow M2$, on obtient ainsi facilement une possibilité de transposition (figure 5).

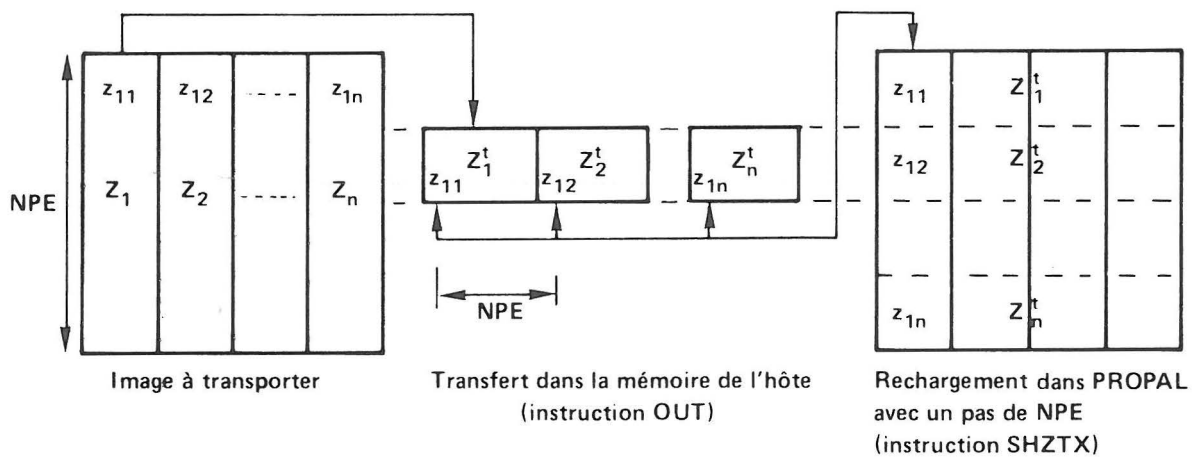


FIGURE 5

Un troisième niveau de structuration nous semble pouvoir être distingué au niveau du pixel, en effet, outre certains systèmes qui considèrent une image $512 \times 512 \times \text{bits}$ comme n plans $512 \times 512 \times 1$ bits, dans le cas d'image multi-dimensionnelle (par exemple LANDSAT) on a le choix en une représentation $512 \times 512 \times (4 \times 8)$ bits ou $4 \times (512 \times 512 \times 8)$ bits et il est important de pouvoir passer d'une structure à l'autre suivant le type de traitement utilisé, là encore on dispose de toute la souplesse nécessaire grâce à l'adressage bit.

De plus travailler avec des pixels d'un octet n'est plus nécessaire : il est bien connu qu'en général un codage sur un nombre plus faible de bits est suffisant (six constitue souvent un maximum, les bits supplémentaires étant du bruit, cf [8], Image quantization p. 154), le seul justificatif était la manière peu aisée de travailler au niveau du bit sur les ordinateurs classiques : ce n'est pas le cas dans PROPAL où l'on peut, de plus, effectuer les opérations, traditionnellement lourdes, de compactage et décompactage propres à la compression d'information en codage d'images.

3.3. - UTILISATION DE LA MEMOIRE M1 A DES FINS ALGORITHMIQUES

M1 est la mémoire rapide dans laquelle vont se dérouler les algorithmes, l'origine des données peut être la mémoire de stockage M2, celle de l'unité de gestion via l'ascenseur ou celle de l'hôte. Cette mémoire peut être considérée par l'utilisateur comme un registre vectoriel de 256 bits de large qu'il va lui appartenir de structurer en zones de largeur variable suivant l'utilisation désirée, par exemple pour effectuer une déconvolution ligne (interpolation sur 7 éléments à coefficients constants).

$$Z_{ij} = \sum_{r=1}^7 (\alpha_r Z_{i,j+k-4})$$

La structure de M1 peut être celle de la figure 6.

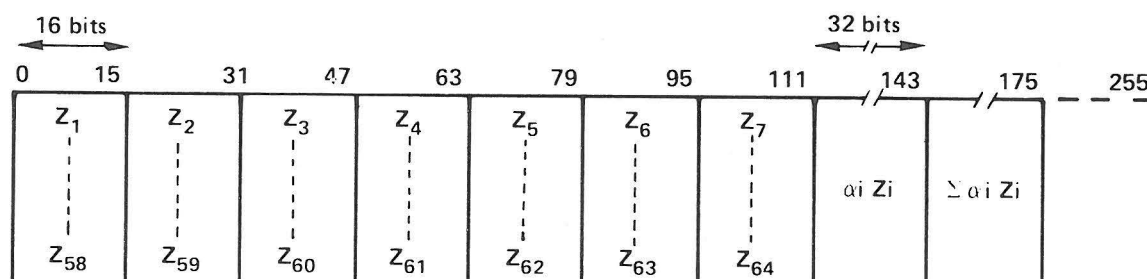


FIGURE 6

On notera que la déconvolution s'effectue par 7 duplications décalées à chaque fois d'une unité vers le haut. Ce processus est typique du traitement de voisinage, il a été utilisé dans une application relevant du pré-traitement des images de télédétection comportant des corrections radiométriques (correction de gain, déconvolution ligne, correction de calibration) et une correction radiométrique associée à une correction géométrique (déconvolution et rééchantillonnage colonne). Le temps de prétraitement d'une image 1024 x 1024 avec un PROPAL de 64PE prenait 3 mn alors qu'un MITRA 125 effectuait le même travail en 57 mn (avec dans les deux cas 1 mn consacrée aux entrées sorties).

En ce qui concerne le traitement des voisinages signalons également le cas des images binaires où le filtrage peut être effectué simplement par adressage de table (figure 7). La zone de 9 bits obtenue par translation et décalage est considérée par l'instruction SHZIT comme table d'adresses pour des valeurs situées dans la mémoire de l'unité de gestion, l'exécution de cette instruction permet de remplacer chaque adresse par ces valeurs.

Une autre approche des images binaires consiste à effectuer un codage en séquences (très rapide sur PROPAL) : chaque séquence de 1 est caractérisée par son origine et sa taille, un tel codage nous a permis de faire de la caractérisation de composantes connexes (figure 8), une fois l'image synthétique obtenue il est possible d'effectuer des traitements statistiques en parallèle (calculs de centre de gravité et d'axe d'inertie par exemple), ces traitements étant effectués lorsque le nombre de PE utilisés est optimal.

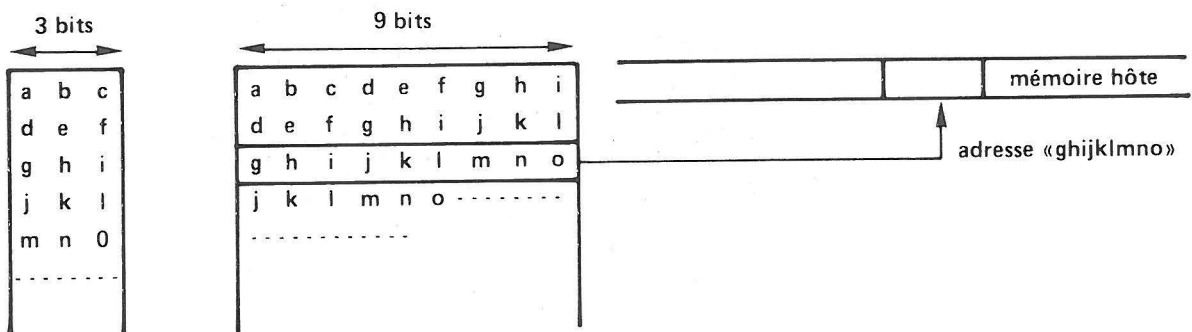


FIGURE 7

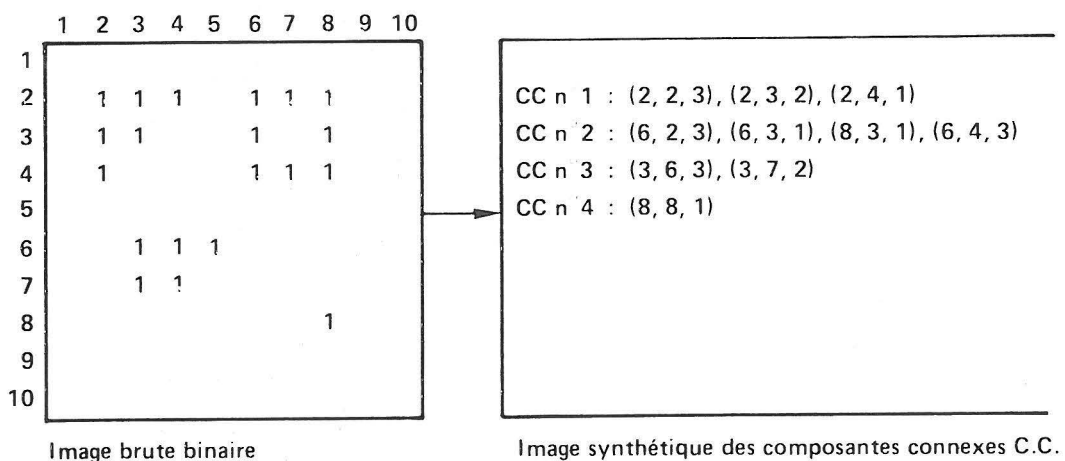


FIGURE 8

4 - CONCLUSION

Le système PROPAL II peut être utilisé en traitement d'images non seulement à des fins calculatoires, mais également à des fins de visualisation temps réel, les possibilités d'extensibilité et de modularité offrent une grande faculté d'adaptation au problème spécifique de l'utilisateur qui peut ainsi optimiser la configuration de son application.

5 - REFERENCES

- 1 - FU, K-S, "Special Computer Architectures for Pattern Recognition and Image Processing", Proc. 1978, National Computer Conference, pp/003-1013
- 2 - BOUDAREL R. et LEGAL Y. "Processeur Parallèle et Associatif PROPAL II Système de traitement des données du radar latéral à ouverture synthétique de SEASAT.A", les équipements de photogrammétrie analytique et de Télédétection, Symposium international de la SIP, Commission II, Paris 12.14. Septembre 1978.
- 3 - GANDOSI Alain, "Le calculateur parallèle et associatif PROPAL II, First European Conference on Parallel and Distributed Processing", Toulouse, France 14.16. February 1979, JC Syre Editor.
- 4 - CIMSA, "Processeur parallèle associatif PROPAL II. Manuel de présentation", ref 8014 IP3/FR, Décembre 1979.
- 5 - CIMSA, "PROPAL II, Manuel de référence", ref 8215/U/FR, Janvier 1980
- 6 - CIMSA, "PROPAL II, Manuel d'exploitation", ref 8216/U/FR. Janvier 1980.
- 7 - TAHE Serge, "Transfert des données dans un calculateur parallèle vectoriel", thèse de Docteur Ingénieur. Université de Rennes I 2 Juillet 1979.
- 8 - PRATT William K. , "Digital Image Processing" Wiley Interscience 1978.