

SEGMENTATION SPATIO-TEMPORELLE SUR LA MACHINE
PARALLÈLE DE VISION TRANSVISION

R.CANALS† J.P.DÉRUTIN† F.HEITZ††

†Laboratoire d'Electronique - URA 830 CNRS,
Université Blaise Pascal, 63177 Aubière Cedex, France
††IRISA - INRIA, Campus de Beaulieu, 35042 Rennes Cedex, France
RÉSUMÉ ABSTRACT

Cet article traite de la parallélisation d'un algorithme de segmentation spatio-temporelle en régions basé sur une modélisation statistique markovienne. L'algorithme mis en œuvre est un algorithme de relaxation globale associé à une procédure d'estimation des paramètres du modèle.

Ce type de processus étant intégré dans des chaînes de traitement plus complexes à contraintes temporelles sévères, une parallélisation de cet algorithme est proposée sur une machine de vision à architecture hétérogène reposant sur deux modèles d'exécution pipe-line et MIMD.

This paper deals with the parallelization of a spatio-temporal segmentation algorithm based on a Markov Random Field model. The algorithm under concern corresponds to a deterministic relaxation scheme associated with a learning procedure for the identification of the parameters of the model.

This kind of process being integrated in more complex processing chains with severe temporal constraints, a parallelization of this algorithm is proposed on a general purpose vision machine presenting a heterogeneous architecture based on both pipe-line and MIMD execution model.

I. Introduction

Cet article traite de la parallélisation d'un algorithme de segmentation spatio-temporelle en régions basé sur une modélisation markovienne. L'algorithme de segmentation considéré [BOU93] fournit, à chaque instant, une partition de l'image analysée en régions homogènes au sens du mouvement et assure un suivi temporel à court terme de ces régions. Il s'appuie sur un algorithme de relaxation déterministe, associé à une procédure d'apprentissage des paramètres du modèle markovien utilisé [GEM84].

Le portage de cet algorithme sur la machine parallèle de vision TRANSVISION développée au L.E. de Clermont-Ferrand, permet d'envisager des temps de traitement compatibles avec les contraintes d'applications en temps-réel.

La machine TRANSVISION présente une architecture hétérogène de type Pipeline et de type MIMD à base de transputers, particulièrement bien adaptée aux algorithmes de vision artificielle. Elle doit, par la suite, être embarquée dans un véhicule routier autonome pour fournir, en temps réel, une segmentation spatio-temporelle de la scène observée [BOU93].

II. Segmentation spatio-temporelle par champs markoviens

La carte de segmentation spatio-temporelle à estimer à chaque instant $e = \{e_s, s \in S\}$ (S correspondant à l'ensemble des sites s) est modélisée par un champ markovien. L'algorithme extrait les partitions spatio-temporelles e à partir d'un ensemble d'observations o correspondant aux gradients spatiaux et temporels préalable-

ment extraits de la séquence d'images. On montre dans [BOU93] que le critère du Maximum A Posteriori (MAP) conduit, dans ce contexte, à minimiser la fonction d'énergie globale suivante :

$$U = \mu \sum_{c \in C} (1 - \delta_{c_e, c_t}) + \sum_{s \in S} \left(\frac{1}{2\sigma_{e_s}^2} (\vec{\nabla} G(s) \cdot \vec{V}_{\theta_{e_s}}(s) + G_t(s))^2 \right)$$

où $\vec{\nabla} G(s)$ et $G_t(s)$ désignent les gradients spatiaux et temporels de la fonction intensité G et $\vec{V}_{\theta_{e_s}}(s)$ correspond au vecteur vitesse au site s représenté par un modèle linéaire de paramètres θ_{e_s} .

L'algorithme de segmentation peut alors être décrit par le synoptique de la figure 1.

L'algorithme est initialisé avec une partition correspondant à une seule région (qui couvre donc toute l'image). A l'itération n , les paramètres du modèle de mouvement dans chaque région de la partition sont estimés au sens du maximum de vraisemblance. Puis, lors d'une ou de plusieurs phases de relaxation, le numéro de région affecté à chaque site s est remis à jour par un algorithme déterministe. Une étiquette supplémentaire dite de "rejet" ρ est proposée et assignée si l'observation locale au site s ne s'accorde avec aucun des modèles de mouvement définis sur les régions déjà créées. L'affectation de l'étiquette ρ traduit la création d'une nouvelle région; à l'issue de la phase de relaxation tous les sites étiquetés ρ sont réétiquetés en composantes connexes.

Cette procédure est itérée à moins que le nombre de sites ayant changé d'étiquette entre deux passes soit inférieur à un seuil. Dans ce cas, on lance le traitement sur l'image



suivante de la séquence.

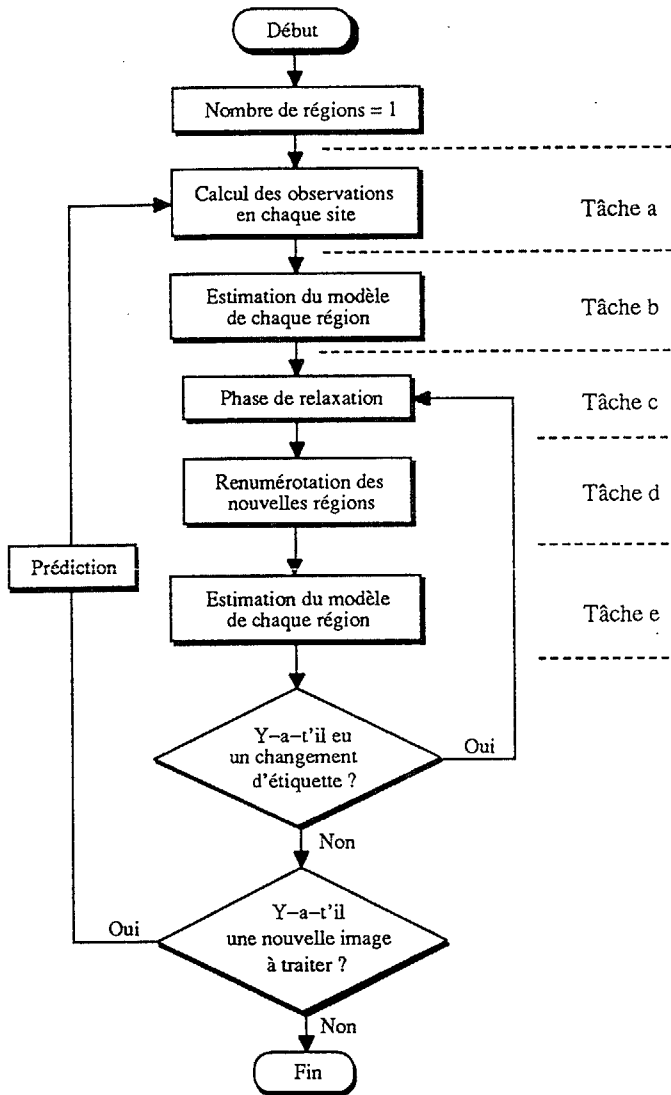


Fig 1 : Synoptique de l'algorithme.

La phase de prédiction, quant à elle, consiste simplement à déterminer la position des sites constituant chaque région dans la carte de segmentation de l'image suivante. Cette carte sert alors de segmentation initiale dans le nouveau processus de relaxation (fonctionnement normal).

III. La machine TRANSVISION

Il s'agit d'une machine hétérogène basée sur un monde pipe-line temps réel vidéo et un monde MIMD à mémoire distribuée [DER91]. Pour être complètement adaptée à la vision artificielle temps réel, cette machine met en œuvre un concept de nœud vidéo permettant la passerelle entre le monde pipe-line et le monde MIMD : le réseau MIMD peut être alimenté et resynchronisé sans goulot d'étranglement. Chaque nœud vidéo est basé sur un processeur élémentaire de type transputer, ainsi que le reste du réseau MIMD.

Le monde pipe-line est constitué d'un ensemble de modules de la firme DataCube. Cette machine permet de définir une configuration machine cible adaptée aux contraintes temporelles de l'application.

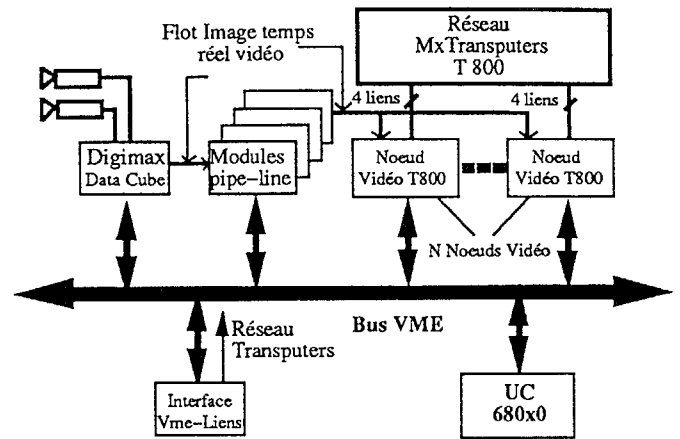


Fig 2 : TRANSVISION.

IV. Parallélisation sur la machine TRANSVISION

A. Implantation sur le modèle MIMD

Comme nous l'avons vu dans la section précédente, l'un des points forts des méthodes de relaxation est qu'elles n'exigent principalement que des calculs locaux. Ceci signifie donc que ces méthodes sont hautement parallélisables et se prêtent bien à une implantation sur la machine TRANSVISION.

Tâche a : Calcul des observations

Cette tâche consiste à déterminer les gradients spatiaux et temporels de deux images successives, les gradients spatiaux se faisant sur la deuxième image. Il est basé sur le masque 3×3 de SOBEL. Le gradient temporel est basé sur une différence d'images. Ces deux calculs sont indépendants des données et de type local. Un parallélisme par partage de données est utilisé. L'image est découpée en bandes par le premier étage de processeurs qui sont en fait des nœuds vidéo. Ces NV_i calculent ces gradients pendant que le reste du réseau effectue la segmentation sur les gradients correspondant au couple d'images précédent.

Le reste du réseau supporte l'ensemble des calculs locaux relatifs aux tâches b à e. Certaines de ces tâches (b et e) présentent des calculs de type local ou global. La tâche c n'est que locale et la tâche d globale.

Ces remarques entraînent la définition d'une structure hiérarchique en introduisant, en plus des PE_i , un processeur PS chargé de gérer les passages local \leftrightarrow global. Une méthode de diffusion de type 1 vers N et N vers 1 est implantée sur l'ensemble du réseau.

La topologie idéale pour ce profil de diffusion est l'étoile, impossible à réaliser avec les transputers.

Nous proposons la topologie présentée dans la figure suivante :

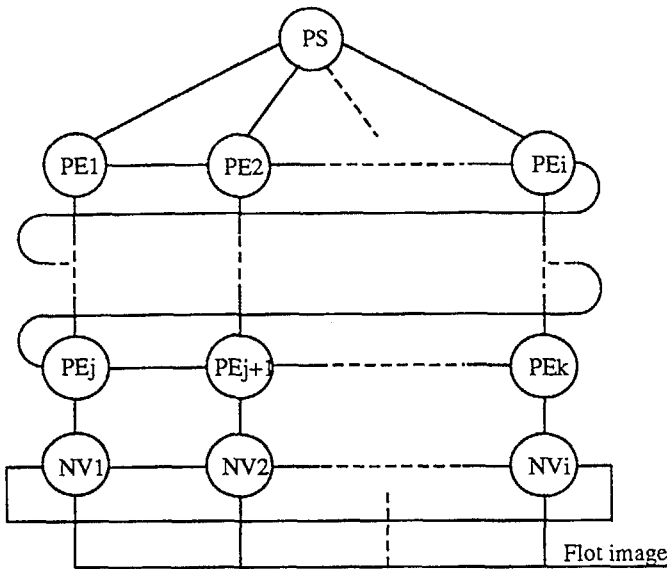


Fig 3 : Topologie MIMD adoptée.

Les liens verticaux entre PE_i permettent les diffusions de 1 vers N et N vers 1. Les liens horizontaux permettent les échanges entre processeurs 2 à 2 pour résoudre les problèmes de bord dus au partage de données décrits dans la tâche c. Les liens entre les nœuds vidéo permettent leur synchronisation pour qu'ils travaillent sur la même image.

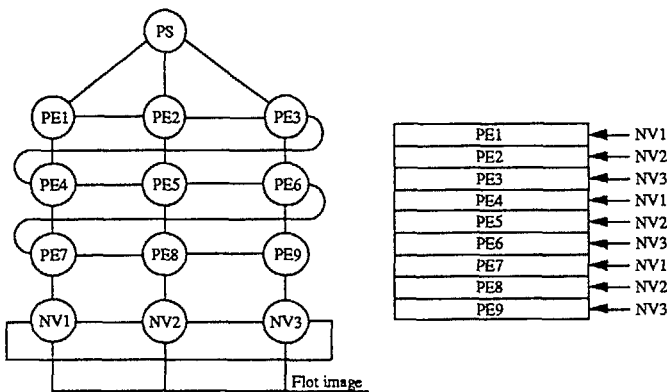


Fig 4 : Calcul des gradients et diffusion dans le réseau.

Tâche b : Calcul du modèle de mouvement de chaque région

Cette tâche est constituée des phases suivantes :

- Calcul des caractéristiques de chaque région.

Pour chaque région, 24 paramètres doivent être calculés. Chaque processeur effectue un balayage de sa bande de données, et en fonction de l'étiquette de région de chaque site, détermine les 24 paramètres de chaque région ou partie de région. Une diffusion de N vers 1 permet de grouper les résultats de manière globale pour chaque région.

- Calcul du modèle de chaque région.

Le modèle de mouvement de chaque région est obtenu par une méthode de maximum de vraisemblance. Ceci revient finalement à résoudre un système matriciel. Le PS possédant l'ensemble des paramètres de chaque région effectue ce calcul. Une diffusion de 1 vers N permet de répartir l'ensemble des modèles de mouvement sur le réseau de PE_i .

- Calcul de la variance sur chaque région.

L'erreur entre le modèle de mouvement et les observations est calculée pour chaque région ou partie de région contenue dans la bande de chaque PE_i .

Une phase de diffusion de N vers 1 permet au PS de calculer la variance sur chaque région qui est alors diffusée vers les PE_i .

Tâche c : Processus de relaxation

Cette tâche est typiquement locale. A partir de la variance, du modèle de mouvement de chaque région et des observations, chaque PE_i effectue le processus de relaxation déterministe sur sa bande. Il doit aussi gérer le tableau d'instabilité de sa bande de données. Un phénomène de bord doit être pris en compte, ce qui implique un échange des frontières des bandes entre les $PE_i \leftrightarrow PE_{i+1}$. Plusieurs balayages de chaque bande peuvent être effectués au cours de cette étape.

Tâche d : Renumerotation des nouvelles régions

Cette tâche est de type globale. Elle est implantée localement sur chaque PE_i et une phase d'homogénéisation est réalisée en fin de traitement.

Chaque PE_i réalise une renumérotation locale des sites "rejet" puis échange ses résultats frontières avec le PE_{i+1} . Cette phase d'échanges s'exécute séquentiellement $PE_1 \rightarrow PE_2, PE_2 \rightarrow PE_3 \dots$. Le dernier processeur émet les résultats vers le serveur. Celui-ci diffuse vers l'ensemble du réseau le nombre total de régions défini dans la phase de renumérotation.

Tâche e : Phase identique à la phase b

La modification du champ des étiquettes de région au cours de la tâche c entraîne la nécessité de cette étape avant d'itérer le processus.

Si le nombre total de sites ayant changé d'étiquette de région entre deux phases de relaxation est inférieur à un seuil d'arrêt prédéfini, le processus de segmentation est fini sur l'image actuelle et l'image suivante de la séquence peut être traitée. Le prochain champ des étiquettes n'est pas initialisé avec une région unique, mais avec un champ prédit à partir du champ résultant de la segmentation précédente.

Le schéma de parallélisation obtenu est de type *MSPMD* dans le sens où chaque processeur effectue un ensemble de traitements identiques. Ce schéma présente une bonne répartition des charges de travail sur le réseau à l'exception du deuxième point de la tâche b et de la tâche c (relaxation). Le calcul du modèle de chaque région est réalisé par PS. Les autres processeurs PE_i sont en attente des résultats. La tâche c est dépendante de la répartition des régions en mouvement dans la totalité de l'image. Ce déséquilibre très complexe à gérer est en cours d'étude.

B. Implantation sur les deux modèles d'exécution de la machine

Au vu des caractéristiques de l'algorithme, et pour mettre à profit les deux modèles d'exécution de la machine, le cal-



cul des différents gradients peut être réalisé par le module pipe-line. Ce module délivre directement aux nœuds vidéo un flot de données correspondant aux gradients spatiaux et temporels.

Cette approche supprime la tâche a, chaque NV_i obtient en temps réel vidéo la bande de données exploitée par la tâche b. Il n'y a plus de problème de diffusion des données vers les processeurs du réseau.

Cette configuration optimale est présentée dans la figure qui suit.

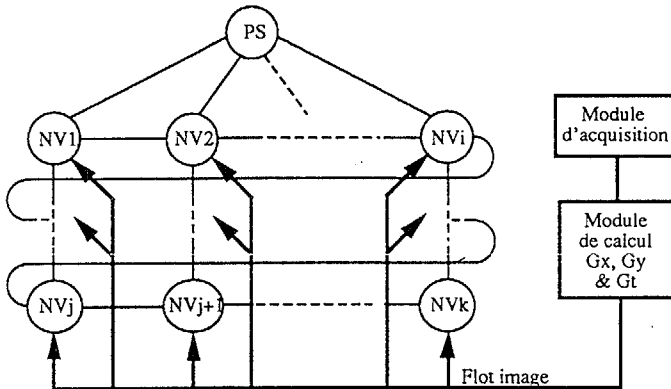


Fig 5 : Topologie pipe-line MIMD adoptée.

V. Résultats

L'implantation de cet algorithme de segmentation est effective sur la machine TRANSVISION. Le schéma de parallélisation fournit les résultats escomptés, résultats dépendant grandement des données. Seule la parallélisation sur le modèle d'exécution MIMD a été testée (configuration de la figure 4). Les temps de traitements obtenus sur deux séquences d'images sont donnés dans la table 1. Ces temps dépendent du nombre d'objets en mouvement dans la scène, du nombre de processeurs utilisés et du champ des étiquettes initial choisi (phase d'initialisation sans connaissance *a priori* du champ des étiquettes ou fonctionnement normal avec connaissance *a priori* issue de la segmentation précédente).

Nombre processeurs		2	4	6	8	10
Séquence carrefour	Initial.	4.89	2.58	1.83	1.41	1.22
	<i>A priori</i>	2.55	1.39	0.98	0.77	0.67
Séquence piétons	Initial.	6.63	3.59	2.56	2.01	1.67
	<i>A priori</i>	4.02	2.16	1.55	1.35	1.11

Tableau 1 : Résultats (en sec.).

Il ressort de ces résultats que pour une séquence donnée, le temps de traitement ne varie pas linéairement par rapport au nombre de processeurs. En effet, comme précisé auparavant, les charges de travail ne sont pas réparties équitablement sur les PE_i lors du calcul du modèle de chaque région et de la phase de relaxation. Cette répartition dépend directement des données contenues dans les images à traiter ; il faut donc opérer une répartition dynamique des charges de travail sur les processeurs. Pour ce faire, le passage à un modèle d'exécution de type *ferme de processeurs* est à envisager : le processeur serveur PS devient le *maître* et les processeurs de traitement PE_i les *esclaves*.

VI. Conclusion

L'étude menée ici a permis, dans un premier temps, de définir le schéma de parallélisation optimal d'un algorithme de segmentation du mouvement apparent sur le modèle d'exécution MIMD de la machine TRANSVISION; ce schéma a ensuite été complété par un module pipe-line.

Les résultats obtenus semblent concluants quant à ce schéma de parallélisation MIMD, mais ils montrent également que le passage à une structure algorithmique de type *ferme de processeurs* s'avère nécessaire pour une répartition des charges de travail équitable sur l'ensemble des processeurs PE_i .

Le passage à la version transputer T9000 est en projet et devrait permettre des gains en temps de calcul de l'ordre de 10.

L'étude présentée dans cet article peut être généralisée à tous les critères de segmentation (texture, ...).

Cette étude s'intègre dans le cadre du GRECO GDR134, GT6 et GT8, ainsi que du projet EUREKA-Prometheus-ProArt.

BIBLIOGRAPHIE

- [BOU93] P. Boutheymy & E.François, Motion segmentation and qualitative dynamic scene analysis from an image sequence, Int. Journ. Computer Vision, vol.10, n° 2, p.157-182, 1993.
- [DER91] J.P.Dérutin, B.Besserer, J.Gallice, A parallel vision machine : TRANSVISION, Proc. of Computer Architecture for Machine Perception CAMP91, Ed B.Zavidovique and P.L.Wendel, Paris, p.241-251, Dec. 91
- [GEM84] S.Geman & D.Geman, Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images, IEEE Trans. on Pattern Anal. Mach. Intelligence, vol.6, n° 6, p.721-741, Nov. 84.

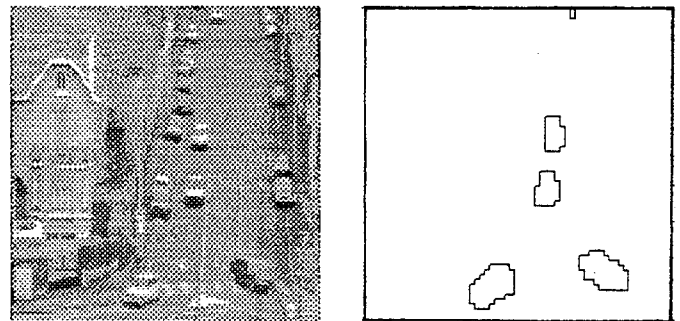


Fig 6 : Séquence carrefour et segmentation.

(Une seule image de luminance et le résultat à l'initialisation sur 10 PE_i).

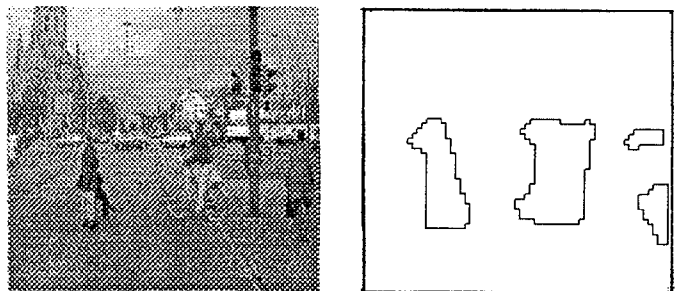


Fig 7 : Séquence piétons et segmentation.

(Idem).