

Traitement et Codage de Séquences d'Images Stéréoscopiques

Nicolas DE SNOECK



Université du Québec

Institut national de la recherche scientifique

INRS-Télécommunications

16, place du Commerce, Verdun

Québec, Canada, H3E 1H6

Octobre 1995

Rapport technique de l'INRS-Télécommunications no. 95-26

Table des matières

1	Introduction	1
2	Stéréoscopie et Séquences vidéo 3D	3
3	Etude théorique des diverses techniques d'estimation de disparité	6
3.1	Appariement de blocs	6
3.2	Estimation MAP (Maximum A Posteriori) et Champ dense	6
3.2.1	disparité seule	7
3.2.2	estimation conjointe disparité/mouvement	8
3.3	Optimisation avec contrainte	10
3.4	Approche multi-hiérarchique ou pyramidale	13
4	Implantation de 2 algorithmes d'estimation	14
4.1	Appariement de blocs	14
4.2	Méthode du champ dense	15
4.2.1	Base de la méthode	15
4.2.2	Raffinement par approche pyramidale	16
4.3	Résultats	17
5	Application au codage des séquences stéréoscopiques	18
5.1	Rappel sur le standard MPEG2	18
5.2	Scalabilité temporelle et séquences stéréo	19
5.3	Résultats	22
6	Conclusion	24
	Annexe	27
A	Champs de disparité	27

Table des figures

2.1	Géométrie épipolaire	4
2.2	Système de caméras parallèles	4
3.1	Contrainte de boucle	10
3.2	Modèle de projection sur une caméra	11
5.1	Prédiction et Types d'images dans MPEG2	20
5.2	Prédiction en mode de scalabilité temporelle	20
5.3	Prédiction pour les séquences stéréo	21
A.1	Champ dense pour "manege" (facteur d'échelle 1 pour λ et γ entre les niveaux).	27
A.2	Champ dense pour "train" (facteur d'échelle 1 pour λ et γ entre les niveaux)	28
A.3	Champ dense pour "manege" (facteur d'échelle 2 pour λ et γ entre les niveaux).	29
A.4	Champ dense pour "train" (facteur d'échelle 2 pour λ et γ entre les niveaux)	30

Chapitre 1

Introduction

Le but général du stage fut d'effectuer des travaux dans le domaine du traitement d'imagerie stéréoscopique, au sein du département *Communications Visuelles* de l'INRS-Télécommunications, sous la direction du professeur Janusz Konrad.

Récemment, le département a fait l'acquisition d'un système de visualisation des images fixes, ainsi que des séquences vidéo en stéréo—à savoir en 3 dimensions. Ce système requiert l'utilisation de lunettes "Stereographics" qui sont dotées de deux "shutters" électroniques (LCD) synchronisés avec le moniteur par émission infrarouge. Le système d'affichage "Viewgraphics" est capable de présenter sur l'écran des images selon la séquence: D-G-D-G-D-G, où D représente une image prise par la caméra de droite du système d'acquisition stéréoscopique et G une image prise par la caméra de gauche (l'acquisition de séquences d'images stéréoscopiques se fait en effet à l'aide de 2 caméras filmant la même scène, comme décrit dans le Chapitre 2). Les lunettes permettent ensuite d'attribuer à chaque oeil la vue correspondante: image D pour l'oeil droit, image G pour l'oeil gauche.

La recherche en stéréovision—TV en relief et multimédia 3D—a commencé très récemment au sein du département. Pour le moment, l'intérêt majeur repose sur le traitement des séquences d'images stéréoscopiques, en vue de leur compression et du codage. C'est dans ce cadre que s'est déroulé le stage.

Les deux principaux objectifs fixés par J. Konrad pour ce stage étaient les suivants :

1. création d'un ou plusieurs programmes permettant l'estimation du champ de disparité entre deux images gauche et droite, ou même l'estimation conjointe de la disparité et du mouvement sur une séquence d'images stéréoscopiques D-G-D-G. Cette estimation pourra être effectuée selon différentes méthodes :
 - appariement de blocs,
 - champ dense,
 - estimation basée région, nécessitant une segmentation préalable des images.

Un certain nombre de ces techniques d'estimation ont donné lieu à une étude bibliographique préliminaire dont les éléments sont fournis dans le Chapitre 3, avant l'implantation de deux de ces techniques qui fait l'objet du Chapitre 4.

2. compression et codage des séquences d'images stéréoscopiques, utilisant l'estimation des champs de disparité et de mouvement pour effectuer une prédiction entre les vues. La technique de codage utilisée s'appuie sur le standard MPEG2, en permettant une certaine compatibilité, comme le montre le Chapitre 5.

Chapitre 2

Stéréoscopie et Séquences vidéo 3D

Avant d'évoquer les quelques techniques de traitement d'images stéréoscopiques qui ont fait l'objet de ce stage, je me permets de préciser certains points sur la stéréovision proprement dite.

Lorsque nous percevons le monde tridimensionnel qui nous entoure, chacun de nos deux yeux effectue en fait une projection de la scène 3D, pour former deux images bidimensionnelles différentes. C'est ensuite le cerveau qui fait la fusion entre ces deux images pour nous donner l'impression de profondeur. Celle-ci provient de la différence entre les coordonnées d'un même point de l'espace 3D sur chacune des deux images, gauche et droite : c'est cette différence qui constitue la disparité.

Un système d'acquisition stéréoscopique consiste à substituer aux deux yeux, deux caméras dont les axes optiques sont plus ou moins convergents et dont les plans de projection sont liés par des paramètres de calibrage (rotation et translation) qui doivent être proprement définis, comme le montre la figure 2.1. Cette même figure nous permet également de préciser une notion importante : celle des lignes épipolaires. En effet, on peut noter que tous les points situés sur une même droite issue d'un centre optique S ont la même projection sur le plan optique correspondant, alors que leurs projections sur l'autre plan optique constituent une droite que l'on appelle ligne épipolaire. Il apparaît alors que, pour un point donné sur l'image gauche, le point correspondant sur l'image droite doit satisfaire une contrainte : se situer sur la ligne épipolaire correspondante. De plus, dans le cas général, comme le montre bien la figure 2.1, toutes les lignes épipolaires se coupent en un point, qui est l'intersection entre le plan optique et la droite joignant les deux centres optiques.

Cependant, dans la plupart des cas (et en particulier dans la majeure partie de ce rapport), on considère que les axes optiques des deux caméras sont parallèles et perpendiculaires à cette droite joignant leurs centres optiques respectifs. Dans ces conditions, en utilisant des considérations géométriques simples développées dans la Section 3.3, on constate que la disparité est purement horizontale et inversement proportionnelle à la profondeur. En effet, comme la droite joignant les centres optiques est parallèle aux plans optiques, les lignes épipolaires des deux images gauche et droite deviennent parallèles et horizontales, comme l'indique la figure 2.2.

Heureusement, même si les conditions d'acquisition ne sont pas toujours parfaites,

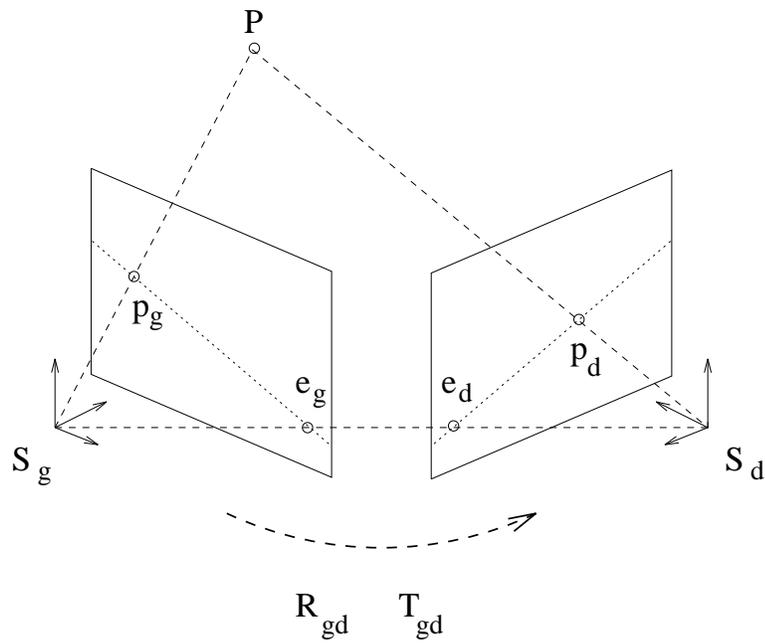


FIG. 2.1 - Géométrie épipolaire

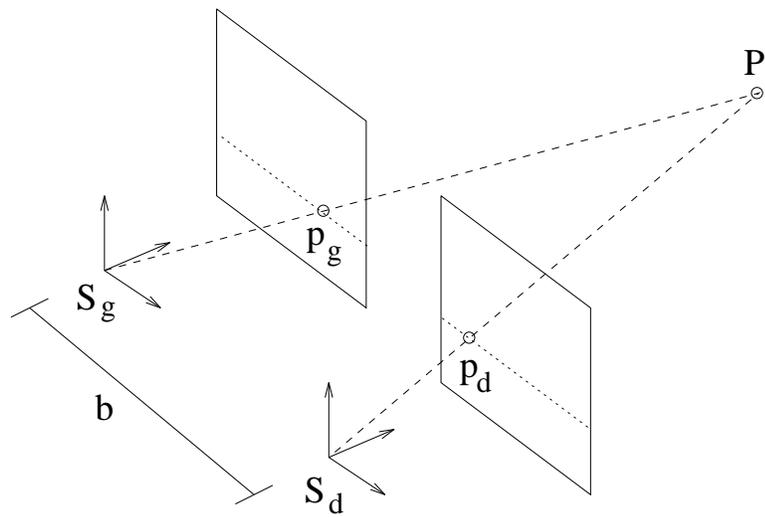


FIG. 2.2 - Système de caméras parallèles

cette hypothèse des caméras parallèles n'est pas réductrice, car il est toujours possible d'effectuer ce qu'on appelle une "rectification" préalable sur les images pour se ramener dans ce cas de figure. Mais ceci fait l'objet d'une autre étude un peu plus complexe que je ne développerai pas.

Sous toutes ces conditions, on voit donc que l'estimation d'un champ de disparité entre les images gauche et droite d'une séquence stéréoscopique peut permettre de reconstruire une image à partir de l'image correspondante, et fournir ainsi un bon moyen de compression de ces séquences.

Tous les tests évoqués dans le cadre de ce rapport ont été effectués essentiellement sur deux séquences : **train** et **manège**. Celles-ci sont constituées d'images de taille 720×576 , entrelacées, de fréquence temporelle 50 Hz, et contiennent chacune plus de 32 images (soit 64 trames). Elles ont été filmées par le CCETT, dans le cadre du projet européen RACE 2045 "DISTIMA" (Digital Stereoscopic Imaging and Applications).

Chapitre 3

Etude théorique des diverses techniques d'estimation de disparité

3.1 Appariement de blocs

La première méthode envisagée pour l'estimation de la disparité est l'appariement de blocs, ou *block-matching*. Cette technique, qui servira de référence pour comparer les résultats des différentes autres méthodes, est directement issue de ce qui existe dans le standard de compression MPEG pour effectuer l'estimation de mouvement.

Son principe est relativement simple. En effet, elle consiste dans un premier temps à découper l'image de référence (par exemple, l'image gauche) en blocs de taille fixe (habituellement 16x16). Puis, pour chaque bloc de l'image gauche, on détermine, au sein d'une fenêtre de recherche autour du bloc correspondant dans l'autre image, le bloc "le plus proche" dans l'image droite, au sens d'une distance plus ou moins complexe reposant sur les différences d'intensité des pixels du bloc.

L'utilisation un peu plus précise de cette technique fera l'objet de la Section 4.1.

3.2 Estimation MAP (Maximum A Posteriori) et Champ dense

Les champs de disparité estimés devant s'avérer physiquement plausibles, il convient d'avoir une certaine connaissance *a priori* de propriétés de régularité de ces champs. La solution proposée ici repose sur une approche bayésienne, faisant intervenir des densités de probabilité *a priori* et *a posteriori*. Cette approche conduit à la minimisation d'une fonction "Energie" et à un problème variationnel.

3.2.1 disparité seule

Le but est de trouver le champ de disparité $V = \{v(x, y)\}$ et les discontinuités Γ maximisant la probabilité *a priori* $p(V, \Gamma|I)$, où I représente les images gauche et droite, à savoir les intensités $(I_1(x, y), I_2(x, y))$. D'après Bayes, cela équivaut à maximiser la probabilité $p(I|V, \Gamma) \cdot p(V|\Gamma) \cdot p(\Gamma)$, à savoir une *Maximisation A Posteriori (MAP)*. Avec le modèle décrit dans [1], cela revient à minimiser une énergie $E(V, \Gamma)$ où interviennent trois facteurs :

- la différence d'intensités $d(x, y, v(x, y)) = I_1(x, y) - I_2(x + v(x, y), y)$ entre les points mis en correspondance par le champ de disparité, sous la forme :

$$\iint_F d^2(x, y, v(x, y)) dx dy + \int_{\Gamma} d^2(s, v) ds \quad (3.1)$$

- le caractère *lisse* du champ de disparité, par l'intermédiaire de son gradient, sous la forme :

$$\lambda \iint_F \|\nabla v(x, y)\|^2 dx dy \quad (3.2)$$

- le caractère *lisse* des discontinuités, reposant sur la longueur de celles-ci, sous la forme :

$$\alpha \int_{\Gamma} ds \quad (3.3)$$

L'énergie $E(V, \Gamma)$ à minimiser s'écrit alors comme la somme de ces trois termes. La résolution de ce problème variationnel donne lieu à deux équations : une du type Euler-Lagrange sur la zone de continuité F du champ de disparité,

$$d(x, y, v(x, y)) \cdot \frac{\partial I_2(x + v(x, y), y)}{\partial v(x, y)} + \lambda \Delta v(x, y) = 0 \quad (3.4)$$

et une condition aux limites marquant l'emplacement des discontinuités, valable le long des courbes Γ

$$\lambda \frac{\partial v}{\partial s_{\perp}} - d(x, y, v(x, y)) \cdot \frac{\partial I_2(x + v(x, y), y)}{\partial v(x, y)} = 0 \quad (3.5)$$

où la dérivée normale correspond à :

$$\frac{\partial v}{\partial s_{\perp}} = \frac{\partial v}{\partial x} \cdot \frac{dy}{ds} - \frac{\partial v}{\partial y} \cdot \frac{dx}{ds}$$

Une approximation aux différences finies sur le réseau de pixels (k, l) des images permet alors de résoudre ces deux équations de façon itérative par une méthode de relaxation déterministe. Pour obtenir les versions discrètes des équations (3.4) et (3.5), on remplace le laplacien $\Delta v(x, y)$ par $4 \cdot (\bar{v}_{kl} - v_{kl})$, où on note :

$$\bar{v}_{kl} = \frac{1}{4} \cdot (v_{k-1,l} + v_{k+1,l} + v_{k,l-1} + v_{k,l+1})$$

et la dérivée normale est approximée par :

$$\frac{\partial v}{\partial s_{\perp}} \approx v_{ij} - v_{cd}$$

où les deux valeurs de la disparité dépendent du type de dérivée approximée (à droite ou à gauche), comme cela est décrit dans [1]. Pour cela, on représente les discontinuités Γ par un ensemble d'éléments $\{\gamma_{kl}^h, \gamma_{kl}^v\}$ situés entre des pixels adjacents horizontalement (h) ou verticalement (v). Chacun de ces éléments vaut 0 s'il existe une discontinuité entre les 2 pixels, et 1 sinon. Ainsi, par exemple, une dérivée normale à droite d'un élément horizontal γ_{kl}^h est approximée par $v_{kl} - v_{k+1,l}$. Les deux équations itératives issues de (3.4) et (3.5) sont respectivement :

$$v_{kl}^{(n+1)} = \bar{v}_{kl}^{(n)} + \frac{d(k, l, v_{kl}^{(n)})}{4\lambda} \cdot \frac{\partial I_2(k + v_{kl}^{(n)}, l)}{\partial v_{kl}^{(n)}} \quad (3.6)$$

et

$$v_{ij}^{(n+1)} = \bar{v}_{cd}^{(n)} + \frac{d(i, j, v_{ij}^{(n)})}{\lambda} \cdot \frac{\partial I_2(i + v_{ij}^{(n)}, j)}{\partial v_{ij}^{(n)}} \quad (3.7)$$

Lors de la procédure d'estimation itérative, les éléments horizontaux et verticaux marquant les discontinuités sont réactualisés au cours de 2 étapes distinctes, afin de pouvoir utiliser les mêmes équations pour des pixels situés dans des "coins".

Cette technique fait apparaître deux problèmes majeurs :

1. elle suppose que les discontinuités sont connues, ce qui n'est pas le cas. Au cours du processus itératif, il faut donc appliquer les deux équations (3.6) et (3.7) à chaque point et garder la valeur du champ qui maximise l'énergie discrète. Si l'équation (3.7) donne le meilleur résultat, on met l'élément γ_{kl} correspondant à 0.
2. comment traiter les occlusions entre les images gauche et droite? Pour ce faire, on introduit une valeur NIL de la disparité qui peut être attribuée au cours de l'itération, ainsi qu'une valeur d_{NIL} de la différence d'intensités. Il faut également modifier en conséquence le terme $\sum d^2$ qui apparaît dans la version discrète de l'énergie, en introduisant le nombre N_{NIL} de pixels dont la disparité est NIL , comme le montre [1].

3.2.2 estimation conjointe disparité/mouvement

Le principe de cette estimation repose sur la même approche bayésienne, mais le calcul est nettement plus complexe car il fait intervenir 4 champs (2 pour les mouvements droit et gauche notés m_d et m_g , et 2 pour les disparités aux instants t et $t + 1$ notés d_1 et d_2), et 2 types de discontinuités (mouvement l_m et disparité l_d). De plus, il est nécessaire de connaître les paramètres de calibrage des caméras.

L'objectif est donc de maximiser une probabilité *a priori* qui peut s'écrire de la façon suivante : $P(m_d, m_g, d_1, d_2, l_m, l_d | I_{g1}, I_{g2}, I_{d1}, I_{d2})$. Comme dans le raisonnement précédent, cette probabilité peut être réécrite, en utilisant la règle de Bayes, sous la forme d'une probabilité *a posteriori*. Le problème se ramène ensuite, en fonction des modèles utilisés, à la minimisation d'une fonction énergétique faisant intervenir divers facteurs.

Dans la modélisation utilisée dans [7], les images sont considérées comme des réseaux discrets de pixels. L'une de ces images est prise comme référence et découpée en blocs carrés disjoints couvrant toute l'image. On désignera par $I(\mathbf{x})$ le vecteur dont les composantes sont les intensités des pixels constituant le bloc centré en \mathbf{x} . On utilisera de plus la notation $v(\mathbf{x})$ pour désigner l'un quelconque des champs de déplacement (mouvement ou disparité), les cliques à deux éléments utilisées étant issues d'un système de voisinage en 8-connexité (ce qui donne 8 cliques).

Vue la complexité des outils utilisés (champs de Gibbs-Markov, processus linéaires ...), je me contenterai de citer les différents types de fonctions "énergie" correspondant aux différentes contraintes imposées aux différents champs (les détails du raisonnement sont donnés dans [7] et certains modèles sont décrits précisément dans [4]) :

- différences d'intensités entre les blocs correspondants droit et gauche, ou t et $t + 1$. On supposera que celles-ci sont indépendantes et identiquement distribuées suivant la densité de probabilité gaussienne $\mathcal{N}(0, N)$. On obtient alors une somme de 4 énergies correspondant aux 4 champs de déplacement, sous la forme :

$$E_i = \sum_{image} \|I_a(\mathbf{x}) - I_b(\mathbf{x} + v(\mathbf{x}))\|_{N-1}$$

où la norme utilisée correspond à $\|X\|_{N-1} = X^T N^{-1} X$.

- contrainte de champs de déplacement lisses, sous la forme :

$$E_v = \sum_{image} \sum_{cliques(\mathbf{x}, \mathbf{y})} \|v(\mathbf{x}) - v(\mathbf{y})\| (1 - l(\mathbf{x}, \mathbf{y}))$$

où $l(\mathbf{x}, \mathbf{y}) = 1$ s'il y a discontinuité, et 0 sinon.

- fonction de coût pour l'introduction de discontinuités, sous la forme :

$$E_l = \beta_l \sum_{lignes} \sum_{cliques} V_l(l, clique, I)$$

où la fonction V_l est explicitée dans [4].

- contrainte épipolaire, sous la forme d'une fonction de coût :

$$E_{ép}(\mathbf{x}, \mathbf{y}) = |\mathbf{T} \cdot (\mathbf{R}\mathbf{x} \times \mathbf{y})|$$

où \mathbf{R} et \mathbf{T} représentent les paramètres de calibrage (rotation et translation) entre les deux caméras, comme indiqués sur la figure 2.1. En théorie, grâce aux

projections sur les plans des caméras, on doit avoir $\mathbf{T} \cdot (\mathbf{R}\mathbf{x} \times (\mathbf{x} + \mathbf{d}(\mathbf{x}))) = 0$. Cette fonction est donc appliquée entre \mathbf{x} et $\mathbf{x} + \mathbf{d}(\mathbf{x})$.

- contrainte de *boucle*, sous la forme d’une fonction de coût proportionnelle à la norme $\|(d_2 + m_l) - (d_1 + m_r)\|$, comme on peut le voir (avec d’autres notations) sur la figure 3.1.

La complexité des équations obtenues nécessite en outre une résolution à l’aide du “recuit simulé”, en faisant intervenir un paramètre de température T que l’on décrémente itérativement :

$$T_{k+1} = \alpha T_k$$

avec $0 < \alpha < 1$ et α proche de 1. Il semble donc que cette méthode soit trop complexe pour pouvoir être raisonnablement implémentée en vue d’obtenir une estimation efficace de la disparité.

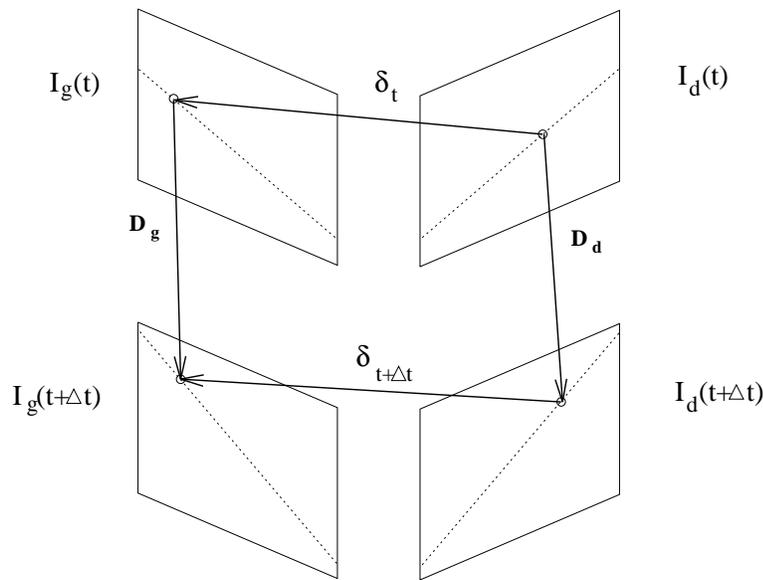


FIG. 3.1 - *Contrainte de boucle*

3.3 Optimisation avec contrainte

Cette méthode fait intervenir conjointement dans l’estimation les champs de disparité et de mouvement, entre les images gauche et droite, aux instants t et $t + 1$. On suppose dans ce qui suit que les paramètres de calibrage des deux caméras sont connus : matrice de rotation \mathbf{R} et vecteur de translation \mathbf{T} pour passer d’un système de coordonnées à l’autre, comme on peut le voir sur la figure 2.1. On peut ainsi écrire,

pour un point $P(x, y, z)$ de l'espace :

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_d = \mathbf{R} \begin{bmatrix} x \\ y \\ z \end{bmatrix}_g + \mathbf{T} \quad (3.8)$$

Lors de l'estimation du champ de disparité, le principe même de projection tridimensionnelle impose que le point homologue recherché dans la deuxième image se trouve sur la *ligne épipolaire*. Si on note (X, Y) les coordonnées de la projection du point P sur le plan d'une caméra, et f la distance focale de cette caméra (voir figure 3.2), on a :

$$X = f \frac{x}{z} \quad , \quad Y = f \frac{y}{z}. \quad (3.9)$$

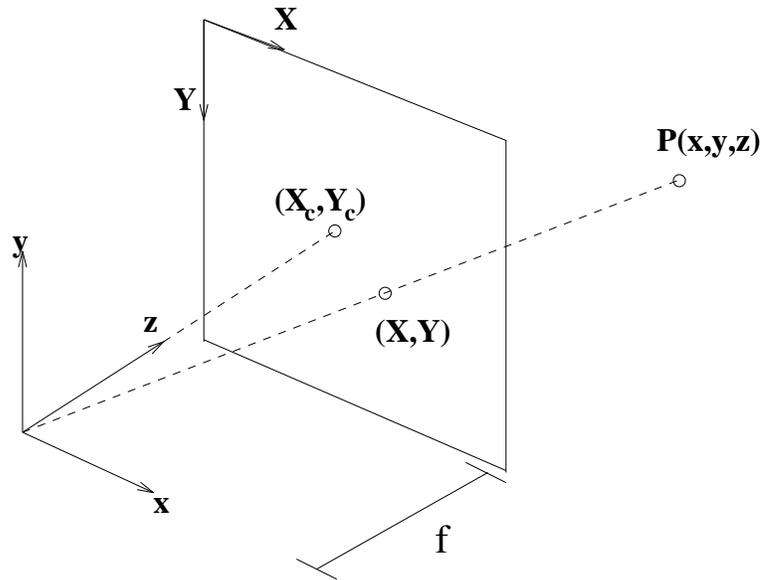


FIG. 3.2 - *Modèle de projection sur une caméra*

De plus, une contrainte géométrique issue du calibrage (facteurs d'échelle et coordonnées des centres optiques projetés) permet de limiter à un intervalle la recherche de ce point homologue par optimisation d'une fonction de corrélation (pour les équations précises, se référer à [8]).

Lorsque l'on cherche à estimer dans le même temps les champs de mouvement sur chaque image, il est alors nécessaire de tenir compte d'une autre contrainte, dite de *boucle*. Ainsi, si l'on note $\vec{\delta}(t)$ le champ de disparité entre l'image gauche et l'image droite à l'instant t , et \vec{D} le champ de déplacement sur une image entre les instants t et $t + 1$, on peut écrire que :

$$\vec{\delta}(t) + \vec{D}_d = \vec{\delta}(t + 1) + \vec{D}_g \quad (3.10)$$

comme le montre la figure 3.1.

Sous l'hypothèse des caméras parallèles (voir figure 2.2) et avec la connaissance des paramètres de calibrage, il est possible d'exprimer l'équation (3.10) sous une autre forme, en faisant apparaître les composantes u et v du vecteur de déplacement (voir [8] et [5]). En effet, d'après l'équation (3.8), on peut écrire :

$$\begin{cases} x_d = r_{11}x_g + r_{12}y_g + r_{13}z_g + t_1 \\ y_d = r_{21}x_g + r_{22}y_g + r_{23}z_g + t_2 \\ z_d = r_{31}x_g + r_{32}y_g + r_{33}z_g + t_3 \end{cases}$$

avec, comme on sait par hypothèse que $z_d = z_g$, $r_{31} = r_{32} = r_{13} = r_{23} = t_3 = 0$ et $r_{33} = 1$.

En utilisant alors l'équation (3.9), on en déduit une version *continue* de l'équation (3.10), sous la forme :

$$\begin{aligned} X_d &= r_{11}X_g + r_{12}Y_g + t_1 f \frac{1}{z_g} \\ Y_d &= r_{21}X_g + r_{22}Y_g + t_2 f \frac{1}{z_g} \end{aligned} \quad (3.11)$$

La correspondance entre les équations (3.10) et (3.11) provient du fait que, sous les hypothèses énoncées ci-dessus, la disparité est inversement proportionnelle à la profondeur z .

En combinant maintenant les deux parties de l'équation (3.11) et en dérivant par rapport au temps t , on peut obtenir la relation attendue :

$$(r_{21}t_1 - r_{11}t_2)u_g + (r_{22}t_1 - r_{12}t_2)v_g + t_2u_d - t_1v_d = 0 \quad (3.12)$$

Le problème de l'estimation conjointe des deux types de champs se réduit alors à trouver le vecteur à 4 composantes $X^* = (u_g, v_g, u_d, v_d)^T$ qui minimise une fonction de coût $\mathcal{F}(X)$ (par exemple, la somme des différences d'intensité de blocs correspondants entre les instants t et $t + 1$, sur les deux images gauche et droite), sous la contrainte exprimée par l'équation (3.12). Cette minimisation peut être effectuée par une méthode classique du gradient, comme cela est décrit dans [8]. La résolution itérative s'écrit alors :

$$X_{k+1} = X_k - \varepsilon [I - C_i^T(C_i C_i^T)^{-1}C_i] \nabla f(X)$$

où C_i est le vecteur-ligne tel que la contrainte (3.12) s'écrive $C_i X = 0$, à savoir :

$$C_i = (r_{21}t_1 - r_{11}t_2, r_{22}t_1 - r_{12}t_2, t_2, -t_1)$$

Comme on peut le constater, cette méthode d'estimation repose sur une approche sensiblement différente de l'approche bayésienne vue précédemment. Toutefois, il semble tout à fait possible d'utiliser certains potentiels résultant d'une ou plusieurs contraintes imposées aux champs dans le calcul probabiliste, en vue de modifier la fonction $\mathcal{F}(X)$ à minimiser pour obtenir une fonction du type de celle introduite dans [5, p.22]. De plus, on pourra noter que, sous les hypothèses énoncées au départ, la contrainte (3.12) résulte directement de ce que nous avons appelé la *contrainte épipolaire* dans l'approche précédente (voir [7]) : il suffit pour cela de développer l'équation $\mathbf{T} \cdot (\mathbf{R}\mathbf{x}_g \times \mathbf{x}_d) = 0$, et de la dériver par rapport à t .

3.4 Approche multi-hiérarchique ou pyramidale

Cette approche, décrite en particulier dans [3], ne constitue pas à proprement parler une technique d'estimation de disparité. En réalité, il s'agit d'un processus hiérarchique intégrant des techniques comme celles énoncées précédemment, et permettant d'améliorer la qualité et la rapidité de l'estimation.

Pour ce faire, on crée à partir de chaque image une *pyramide* par des opérations successives de filtrage et de sous-échantillonnage. Ainsi, à partir d'images de taille 256×256 (niveau le plus fin), on crée des images de tailles respectives 128×128 , 64×64 et 32×32 (niveau le plus grossier). L'estimation (quelle que soit sa forme) est alors effectuée à chaque niveau, en utilisant pour initialisation du processus itératif au niveau k , une prédiction du vecteur-déplacement reposant sur 3 critères :

prédiction hiérarchique le meilleur de 4 vecteurs voisins choisis au niveau $k - 1$

prédiction temporelle vecteur choisi au niveau k de l'ensemble d'images précédent

prédiction spatiale le meilleur de 3 vecteurs sur la ligne précédente du niveau k

Selon le type de machine utilisée pour effectuer le calcul, il peut être envisageable de réaliser en parallèle 3 réestimations différentes à partir de 3 valeurs possibles de la prédiction, et d'effectuer le choix sur le résultat de ces estimations. En pratique, il semble plus concevable de choisir, dès ce niveau, une unique prédiction qui servira d'initialisation à un seul processus de réévaluation du vecteur-déplacement à l'échelon hiérarchique k . Le choix du vecteur que l'on va conserver se fait comme toujours en cherchant celui qui minimise un coût d'estimation basé sur une différence d'intensités entre les deux blocs correspondants.

Il est également envisagé de faire intervenir une mesure de confiance sur la prédiction effectuée, afin de pallier l'apparition de discontinuités ou le phénomène d'occlusion. Celle-ci est décrite avec précision dans [2], et est basée sur l'analyse d'une surface de corrélation entre les deux images liées par le champ de déplacement que l'on cherche à estimer.

Une fois la prédiction du vecteur-déplacement effectuée, le processus de réévaluation de ce vecteur décrit dans [3] repose sur la minimisation d'une somme de deux énergies pondérées par la mesure de confiance effectuée auparavant, et conduit à une résolution itérative de la forme :

$$v_{k+1} = (1 - C)\bar{v}_k + Cv_0$$

où les notations sont similaires à celles introduites dans la Section 3.2.2.

Toutefois, il semble possible d'adapter ce processus de réévaluation dans le cadre d'une estimation conjointe des champs de mouvement et de disparité, en utilisant, à chaque niveau hiérarchique d'un ensemble de 4 pyramides formées à partir de 4 images (gauche et droite, à t et $t + 1$), la technique d'optimisation avec contrainte décrite dans la Section 3.3. La résolution itérative résultant de la méthode des gradients serait alors initialisée avec les solutions du niveau hiérarchique précédent.

Chapitre 4

Implantation de 2 algorithmes d'estimation

En dépit de la multiplicité des techniques évoquées dans le Chapitre 3, la complexité de la plupart d'entre elles a rendu impossible l'implantation des algorithmes correspondants. Néanmoins, deux de ces méthodes ont été implémentées, ce qui nous a permis d'obtenir quelques résultats comparatifs intéressants.

4.1 Appariement de blocs

Comme je l'indiquais dans la Section 3.1, la méthode d'estimation qui sert de référence est l'appariement de blocs. Le gros intérêt de cette méthode est qu'elle est simple à mettre en oeuvre, et surtout qu'elle est à la base du standard MPEG de compression des séquences vidéo traditionnelles, ce qui assure une certaine compatibilité avec un éventuel schéma de codage, comme on le verra dans le Chapitre 5.

Ainsi, après avoir récupéré sous une forme adaptée les séquences gauche et droite, le programme recherche, dans une fenêtre définie par l'utilisateur, pour chaque bloc dont l'utilisateur définit la taille (habituellement 16x16) de l'image de référence, le bloc "le plus proche" dans l'image correspondante, au sens d'une distance qui est la somme pondérée de deux termes :

- somme des carrés des différences d'intensité pour chaque pixel correspondant sur le bloc (on parle de DFD),
- carré de la norme de la différence entre le vecteur de disparité et celui du bloc précédent, assurant le caractère lisse du champ.

Cette recherche peut être effectuée avec une précision d'un pixel ou d'un demi-pixel. Dans ce deuxième cas, il est nécessaire d'effectuer une interpolation sur les valeurs de la luminance entre deux pixels. On pourra utiliser une interpolation bilinéaire (moyenne des pixels voisins) ou bicubique (convolution permettant d'obtenir une intensité continue).

Une fois le champ de disparité estimé pour tous les blocs qui constituent l'image, celui-ci peut être stocké sous deux formes différentes, selon l'utilisation que l'on veut en faire par la suite :

- un vecteur par bloc,
- un vecteur par pixel.

A partir de ce champ, on peut alors construire l'image prédite à partir de l'image de référence, ainsi que l'erreur de prédiction.

4.2 Méthode du champ dense

La seconde méthode utilisée pour estimer la disparité entre les images gauche et droite d'une séquence stéréoscopique, permet d'obtenir directement un champ dense, à savoir un vecteur de disparité par pixel. Le principe consiste à résoudre de façon itérative un problème variationnel de minimisation.

4.2.1 Base de la méthode

Le champ de disparité doit faire correspondre entre eux des pixels qui correspondent au même point physique tridimensionnel. En plus de la DFD, comme pour le block-matching, on tient compte également du caractère lisse du champ que l'on veut obtenir dans la fonction de coût que l'on cherche à minimiser. Celle-ci est donc de la forme :

$$\mathcal{P}(u) + \lambda \mathcal{S}(u)$$

où λ contrôle le poids accordé au caractère lisse du champ, et $u(x, y)$ représente le champ de disparité supposé horizontal en première approximation (hypothèse des caméras parallèles évoquée dans le Chapitre 2).

La première fonction représente l'erreur sur l'intensité :

$$\mathcal{P}(u) = \iint [L(x, y) - R(x + u(x, y), y)]^2 dx dy \quad (4.1)$$

La seconde représente la régularité du champ par l'intermédiaire des dérivées partielles :

$$\mathcal{S}(u) = \iint (u_x^2 + u_y^2) dx dy \quad (4.2)$$

Le problème de minimisation donne lieu à la résolution d'une équation différentielle de type Euler-Lagrange :

$$\lambda \Delta u + [L(x, y) - R(x + u(x, y), y)] R_x(x + u(x, y), y) = 0 \quad (4.3)$$

Comme les images sont en fait constituées de pixels, il est alors nécessaire de résoudre l'équation (4.3) de façon discrète, en utilisant l'approximation aux différences

finies $\Delta u = 4(u^*(i, j) - u(i, j))$, où $u^*(i, j)$ représente la moyenne de u sur les voisins du pixel (i, j) en 4-connexité.

En réarrangeant un peu l'équation discrète obtenue, on obtient une équation itérative qui permet d'obtenir théoriquement la valeur du champ en chaque pixel :

$$u^{n+1}(i, j) = u^{*n}(i, j) + \frac{1}{4\lambda} [L(i, j) - R(i + u^{*n}(i, j), j)] R_x(i + u^{*n}(i, j), j) \quad (4.4)$$

où n indique l'étape de l'itération.

Dans la pratique, il s'avère que cette méthode simplifiée ne donne pas de résultats satisfaisants, dans la mesure où les images ne satisfont pas parfaitement la contrainte des caméras parallèles. Par conséquent, il est nécessaire de conserver un champ de disparité bidimensionnel (u, v) , ce qui permet d'obtenir deux équations itératives du type (4.4). Néanmoins, pour que cette méthode n'introduise pas une disparité verticale trop importante, il convient d'ajouter une autre contrainte sous la forme d'une troisième fonction de coût $\gamma \mathcal{D}(v)$, avec :

$$\mathcal{D}(v) = \iint v^2(x, y) dx dy \quad (4.5)$$

Ce nouveau terme dans la fonction de coût modifie légèrement l'équation itérative (4.4), en ce qui concerne la disparité verticale v . Cependant, quels que soient les paramètres λ et γ , ce processus ne parvient jamais à converger vers le champ voulu, mais la solution reste proche du champ nul qui servait d'initialisation. Une amélioration est donc nécessaire, et elle passe par une approche pyramidale.

4.2.2 Raffinement par approche pyramidale

Comme on vient de le voir, la méthode itérative qui doit conduire à un champ dense pose un problème de convergence, inhérent au fait que la solution initiale est le champ nul.

Pour pallier ce problème, l'approche pyramidale décrite dans la Section 3.4 est très utile. En effet, celle-ci consiste à créer une pyramide d'images dont les tailles sont divisées par 2 dans les deux coordonnées à chaque étage. Ensuite, on peut appliquer la méthode présentée ci-dessus à chaque étage, la solution obtenue à l'étage k servant d'initialisation pour l'étage $k + 1$. On peut alors constater que la réduction de taille des images permet de s'éloigner plus rapidement du champ nul, et ainsi de converger aux étages supérieurs vers une solution cohérente.

Toutefois, un problème différent propre à l'approche pyramidale se pose à présent : comment doit-on faire évoluer les paramètres λ et γ lors des changements d'étages ? Compte tenu des changements d'échelle, il semble logique de les multiplier par 2 en passant à un étage plus fin, afin de conserver les mêmes coûts par pixel. Cependant, dans la pratique, il apparaît que le résultat final dépend surtout des valeurs de λ et de γ à l'étage final.

4.3 Résultats

Comme je l'ai indiqué dans le Chapitre 2, les tests effectués l'ont été essentiellement sur deux séquences : **train** et **manège**. Les résultats présentés ont été obtenus sur l'image initiale 720×576 .

Les différents champs de disparité pour les premières trames de **train** et **manège** obtenus sont présentés dans l'Annexe. Pour chacune des deux séquences, les champs présentés sont les suivants :

1. champ dense pyramidal avec $\lambda = 1000$ constants à chaque étage,
2. champ dense pyramidal avec $\lambda = 1000$ divisés à chaque étage.

Au vu de ces champs, on constate que la méthode du champ dense avec approche pyramidale permet d'obtenir des résultats intéressants. Néanmoins, l'observation des images prédites à l'écran nous montre que des erreurs majeures subsistent, surtout sur le **manège**. En effet, la méthode utilisée ne tient pas compte des discontinuités dans le champ qui apparaissent aux contours des objets : au contraire, elle tend à lisser ce champ sur toute l'image. Or le **manège** présente de nombreuses discontinuités.

Malgré tout, ces résultats sont très encourageants, dans la mesure où toute méthode de codage fait intervenir une erreur de prédiction. Ces deux méthodes sont donc utilisables pour créer un schéma de codage des séquences stéréoscopiques. En l'occurrence, la technique d'appariement de blocs va pouvoir s'inscrire dans l'adaptation du standard MPEG2 pour ce codage.

Chapitre 5

Application au codage des séquences stéréoscopiques

5.1 Rappel sur le standard MPEG2

Avant de se lancer dans l'application aux séquences 3D, il convient tout d'abord de rappeler certains points essentiels du standard MPEG2 de compression des images animées (MPEG = Moving Pictures Experts Group). Le cadre d'application de ce standard est très large, puisqu'il doit permettre de coder des séquences dont la qualité peut aller de la TV classique à la TV Haute Définition, avec des débits entre 5 et 40 Mbit/s (voire moins). De plus, des contraintes d'utilisation s'imposent dans MPEG2, telles que l'accès aléatoire ou encore l'avance et le retour rapides.

Une description détaillée du fonctionnement de l'algorithme de codage pourra être trouvée dans [6], mais je me permets de préciser ici certains points majeurs. Tout d'abord, les principales étapes de cet algorithme sont les suivantes :

1. prédiction temporelle par compensation de mouvement, à partir de champs de déplacement calculés par blocs de 16×16 pixels qui définissent des **Macro-blocs (MB)**.
2. transformation des erreurs de prédiction par DCT (Transformation en Cosinus Discret).
3. quantification des coefficients ainsi transformés.
4. codage entropique des niveaux de quantification.

En outre, pour satisfaire aux contraintes indiquées ci-dessus, il est nécessaire de distinguer 3 types d'images :

Images I (intra) images de référence pour les accès aléatoires et le codage des autres images (sauf I), autonomes et n'utilisant aucune compensation de mouvement.

Images P (prédites) images de référence pour le codage des autres images (sauf I), utilisant la compensation de mouvement avant, dite **forward**, par rapport à l'image de référence précédente.

Images B (bidirectionnelles) images prédites par compensation de mouvement avant, dite **forward**, ou arrière, dite **backward**, ou les deux avec interpolation.

La norme reste très souple quant à la répartition des différents types d'images, mais l'utilisation la plus courante est représentée par la Figure 5.1 qui illustre bien les relations entre les types d'images.

De nombreux autres éléments entrent en jeu dans la norme MPEG2, comme les différents modes de codage des Macro-blocs (fixe, inter ou intra) ou encore la distinction entre les prédictions “field” et “frame” pour les séquences entrelacées. Mais tout ceci nous ferait entrer dans des détails techniques qui ne font pas l'objet de notre sujet, et qui n'ont pas de lien étroit avec l'application aux images stéréoscopiques. En revanche, l'un des atouts fondamentaux de cette norme est l'introduction d'outils de “scalabilité” standardisés :

- partition de données
- scalabilité spatiale
- scalabilité temporelle

Ces deux derniers outils ont été développés pour permettre à la norme MPEG2 d'être utilisables par l'intermédiaire de différents récepteurs qui disposeraient de résolutions spatiales (TV classique / TVHD) ou de débits (ou fréquences) différents. Ceci se traduit par un codage à deux niveaux : *base layer* et *enhancement layer*.

5.2 Scalabilité temporelle et séquences stéréo

Comme on vient de le préciser, la norme MPEG2 dispose de plusieurs options, parmi lesquelles la *scalabilité temporelle*. Celle-ci doit permettre de coder des images à haute résolution temporelle (pour la TVHD par exemple), de façon à ce que tous les récepteurs puissent obtenir la meilleure qualité possible. Ainsi, le *base layer* correspond à la séquence dont la fréquence est celle de la TV classique, alors que le *enhancement layer* contient les informations additionnelles pour atteindre le débit de la TVHD. Dans la pratique, le codage s'effectue donc en deux temps :

1. codage MPEG2 “normal” du *base layer*, comme décrit dans la Section 5.1.
2. codage du *enhancement layer* par prédiction temporelle à partir des deux niveaux.

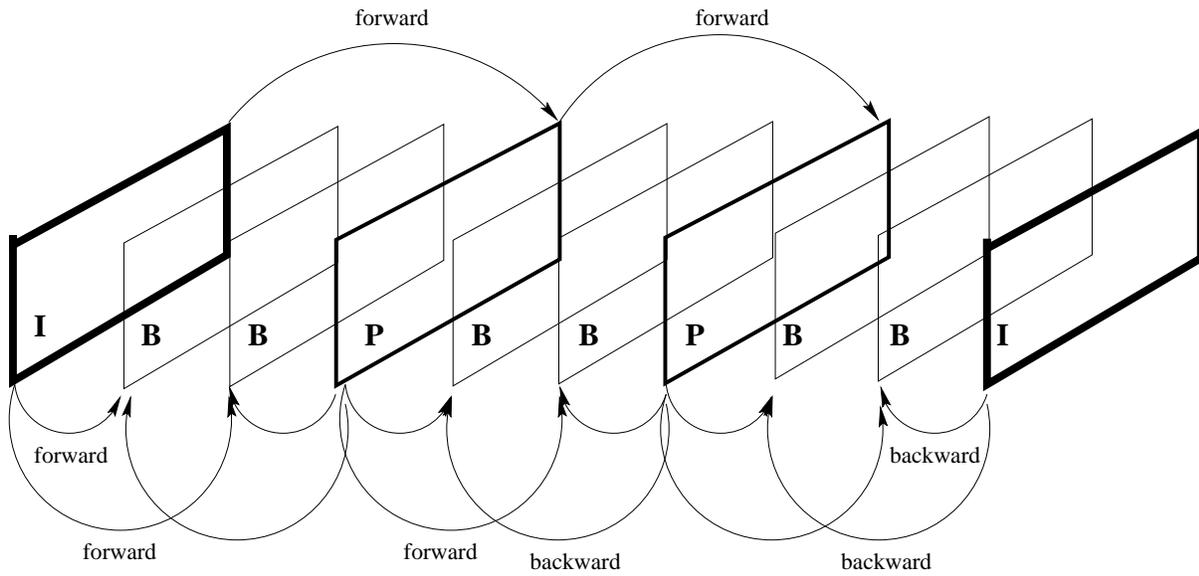


FIG. 5.1 - *Prédiction et Types d'images dans MPEG2*

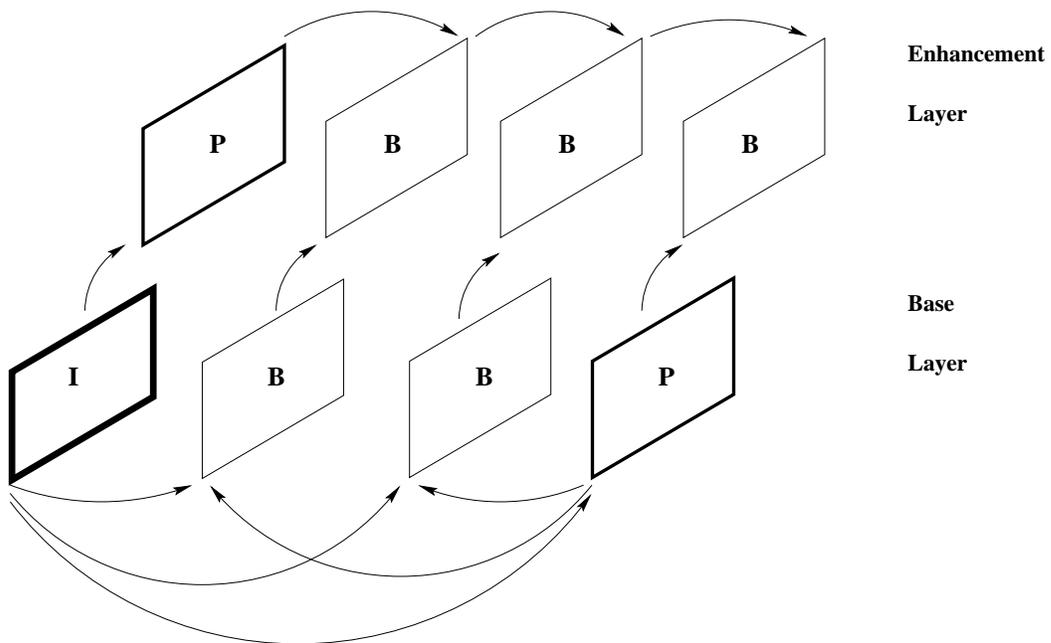


FIG. 5.2 - *Prédiction en mode de scalabilité temporelle*

C'est ce que montre de manière simplifiée la figure 5.2.

L'idée qui vient alors à l'esprit lorsqu'on travaille sur des séquences stéréoscopiques est de remplacer les deux niveaux de codage par les deux séquences gauche et droite. Mais cette fois, le lien qui existe entre les deux niveaux ne sera plus simplement temporel (compensation de mouvement), mais lié à la disparité. Toutefois, le schéma de prédiction peut s'obtenir de façon tout à fait naturelle, comme le montre la figure 5.3, en substituant le champ de disparité (trait pointillé) au champ de mouvement que l'on conserve pour les prédictions au sein d'un même niveau.

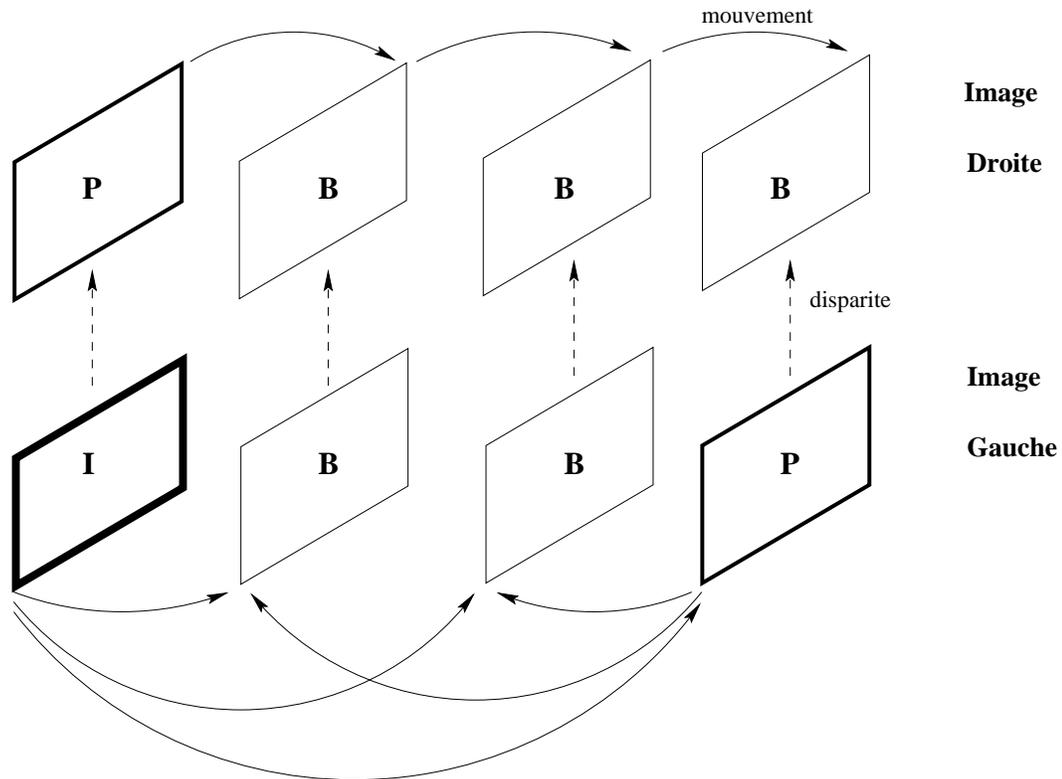


FIG. 5.3 - Prédiction pour les séquences stéréo

Malgré l'apparente simplicité de l'analogie, plusieurs problèmes se posent, et il suffit d'observer attentivement la figure 5.3 pour s'en convaincre.

Tout d'abord, comme les images I ont pour rôle essentiel de permettre un accès aléatoire à la séquence et doivent donc être codées sans aucune compensation, ce type d'images ne doit pas apparaître dans le codage de la séquence droite: en effet, comme l'accès aléatoire peut se faire uniquement sur l'une des deux séquences, il est intéressant du point de vue de la compression d'autoriser la compensation de disparité sur toutes les images de la séquence droite. En revanche, toujours pour ce problème d'accès aléatoire, il est nécessaire qu'à une image I de la séquence gauche, corresponde une image de référence, et donc une image P, pour la séquence droite. Toutes les autres images sont considérées comme des images B, car elles peuvent être codées à partir de deux vecteurs de déplacement.

Le second problème, et non des moindres du point de vue de l'utilisation d'un algorithme standard, concerne le lien entre les images et les champs de déplacement. En effet, comme on l'a vu dans la Section 5.1, les images P sont codées à partir de l'image de référence (I ou P) qui précède, donc grâce à un champ *forward*. Or, dans notre cas de figure, il est naturel de conserver comme champ *forward*, le champ de mouvement entre deux images successives de l'image droite : mais alors, une image P sera codée grâce au champ de disparité qui est un champ *backward*. Loin d'être purement syntaxique comme on pourrait le penser, cette anomalie entraîne des adaptations importantes dans l'algorithme d'un codeur MPEG2 qui se voudrait entièrement compatible, ce qui était notre but.

5.3 Résultats

Comme pour les expériences relatées dans le Chapitre 4, les tests de ce codeur MPEG2 "stéréo" ont été effectués sur les deux séquences stéréoscopiques **train** et **manège**, dont les caractéristiques sont précisées dans le Chapitre 2, sur respectivement 32 et 24 images. Le choix des débits de codage a été guidé par la nécessité d'obtenir une bonne qualité pour la séquence gauche, tout en conservant un nombre de bits suffisant pour la séquence droite. Toutefois, il est envisageable que la qualité intrinsèque de la séquence droite soit subjectivement insuffisante, alors que la qualité de la séquence stéréo demeure acceptable. Tous ces critères nous ont poussés à choisir un débit pour la séquence droite, moitié de celui attribué à la séquence gauche.

Ainsi, pour chaque séquence, 3 séries de codage ont été effectuées :

- gauche à 2 Mbit/s et droite à 1 Mbit/s
- gauche à 4 Mbit/s et droite à 2 Mbit/s
- gauche à 8 Mbit/s et droite à 4 Mbit/s

On peut alors parler de codages 2 + 1, 4 + 2 et 8 + 4.

Pour donner une idée quantitative des résultats de ces expériences, des rapports signal sur bruit (PSNR) ont été calculés à partir des images gauches et droites décodées localement au cours du codage. Les PSNR moyens pour la luminance de chaque séquence sont regroupés dans les deux tableaux suivants, le premier pour les séquences gauches, le second pour les séquences droites.

Toutefois, au-delà des chiffres, c'est la qualité subjective des séquences décodées qui est importante. A ce niveau, plusieurs observations intéressantes peuvent être faites.

Dans un premier temps, l'effet de blocs provenant de la technique d'estimation du codage MPEG2 apparaît sur chaque séquence lorsque le débit diminue, ce qui est tout à fait normal dans le cadre du standard de compression. Néanmoins, on constate que

Séquence	2 + 1 Mb/s	4 + 2 Mb/s	8 + 4 Mb/s
Train	31.68	35.48	37.4
Manège	27.36	29.94	33.12

TAB. 5.1 - *PSNR sur la luminance des séquences gauches*

Séquence	2 + 1 Mb/s	4 + 2 Mb/s	8 + 4 Mb/s
Train	31.15	34.1	36.54
Manège	27.71	29.38	32.15

TAB. 5.2 - *PSNR sur la luminance des séquences droites*

cet effet est fortement atténué lorsque l'on visionne la séquence en stéréo, en raison d'un phénomène de masquage inhérent à la vision stéréoscopique.

De plus, on note que, dès le débit 4+2, la qualité visuelle de la séquence stéréoscopique est très bonne, surtout pour le train, alors qu'une image fixe d'une des deux séquences reste médiocre. En effet, cela résulte de la superposition de deux effets : le mouvement et le masquage stéréoscopique.

Tout cela peut permettre d'envisager techniquement le codage de séquences stéréoscopiques à des débits raisonnables, surtout dans la mesure où l'on peut envisager l'utilisation de techniques d'estimation plus efficaces, comme par exemple certaines évoquées au Chapitre 3 utilisant conjointement la disparité et le mouvement.

Chapitre 6

Conclusion

Dans le cadre de ce stage, ont été analysées un certain nombre de techniques d'estimation de disparité, et deux d'entre elles ont été implantées afin de les utiliser dans un schéma de codage des séquences d'images stéréoscopiques s'appuyant sur le standard de compression MPEG2.

Les résultats des premières expériences de codage ont donné des résultats très intéressants, puisqu'ils permettent d'obtenir une bonne qualité stéréo pour des débits relativement réduits. Toutefois, de nombreuses expériences doivent encore être faites, en modifiant par exemple la différence de débit entre les deux séquences gauche et droite, ou encore en cherchant la répartition qui offrirait la meilleure qualité pour un débit total fixé. De plus, comme on l'a vu, il existe de nombreuses méthodes d'estimation qui n'ont pas encore été implantées en raison de leur grande complexité de calcul, mais qui pourraient donner de meilleures estimations des champs de déplacement (en particulier, des estimations conjointes mouvement/disparité).

En outre, le progrès des techniques de segmentation d'images permet d'envisager un nouveau type d'estimation, basée sur les régions déterminées lors de cette segmentation. Cependant, ce genre de méthode pose toute sorte de nouveaux problèmes en terme de codage, dans la mesure où les vecteurs de déplacement sont sensiblement réduits, alors que l'information nécessaire pour définir la segmentation de l'image reste très importante.

On constate donc que l'avenir de la télévision en 3D reste très ouvert, avec de nombreuses perspectives, mais aussi de nombreux problèmes. Par exemple, quels sont les modes d'affichage à retenir pour le confort de l'utilisateur : lunettes ou écrans lenticulaires ? Ou encore, quels sont les problèmes psychophysiques qui peuvent se poser ? Cette dernière question semble être l'élément majeur qui va guider le futur de cette technologie : en effet, dans le codage que nous avons utilisé, l'image perçue par l'oeil droit est de moins bonne qualité que celle perçue par l'oeil gauche, ce qui pourrait provoquer à long terme des perturbations au niveau du cerveau. De nombreux tests sont mis en place à ce sujet dans différents laboratoires, mais la recherche se poursuit.

Remerciements

Je tiens tout d'abord à remercier mon directeur de stage, M. Janusz KONRAD, pour ses encouragements et son aide précieuse, technique et morale, qui m'ont permis de mener à bien mon travail dans d'excellentes conditions.

Je dois également saluer la contribution majeure de Nadia BAAZIZ, dans tout ce qui concerne les techniques de compression MPEG2, ainsi que le travail de Stéphane COULOMBE et Albert GOLEMBIOWSKI pour les outils d'acquisition des séquences d'images.

Enfin, je voudrais remercier l'ensemble du groupe *Communications Visuelles* de l'INRS-Télécommunications, Aiaze MITHA et Juan Manuel RAMOS pour leur amitié et leur grande disponibilité tout au long du stage.

Bibliographie

- [1] T. Aach and A. Kaup, "MAP-estimation of dense disparity-fields for stereoscopic images," in *Singapore International Conference on Image Processing*, (Singapore), pp. 113–117, Septembre 1992.
- [2] P. Anandan, "A computational framework and an algorithm for the measurement of visual motion," *Intern. J. Comput. Vis.*, vol. 2, pp. 283–310, 1989.
- [3] B. Chupeau, "A Multiscale Approach to the Joint Computation of Motion and Disparity. Application to the Synthesis of Intermediate Views," in *4th European Workshop on Three-Dimensional Television*, (Rome, Italy), pp. 223–230, Octobre 1993.
- [4] J. Konrad and E. Dubois, "Bayesian estimation of motion vector fields," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-14, pp. 910–927, Septembre 1992.
- [5] R. Laganière and F. Labonté, "Stereokineopsis: a survey," tech. rep., INRS-Télécommunications, Mars 1995.
- [6] D. LeGall, "MPEG: A video compression standard for multimedia applications," *Communications ACM*, vol. 34, pp. 46–58, Avril 1991.
- [7] S. Malassiotis and M. G. Strintzis, "Joint motion/disparity estimation for stereo image sequences," in *Proc. SPIE Visual Communications and Image Process.*, (Chicago), pp. 614–625, Septembre 1994.
- [8] A. Tamtaoui and C. Labit, "Constrained disparity and motion estimators for 3DTV image sequence coding," *Signal Process., Image Commun.*, vol. 4, pp. 45–54, 1991.

Annexe A

Champs de disparité



FIG. A.1 - Champ dense pour "manège" (facteur d'échelle 1 pour λ et γ entre les niveaux).

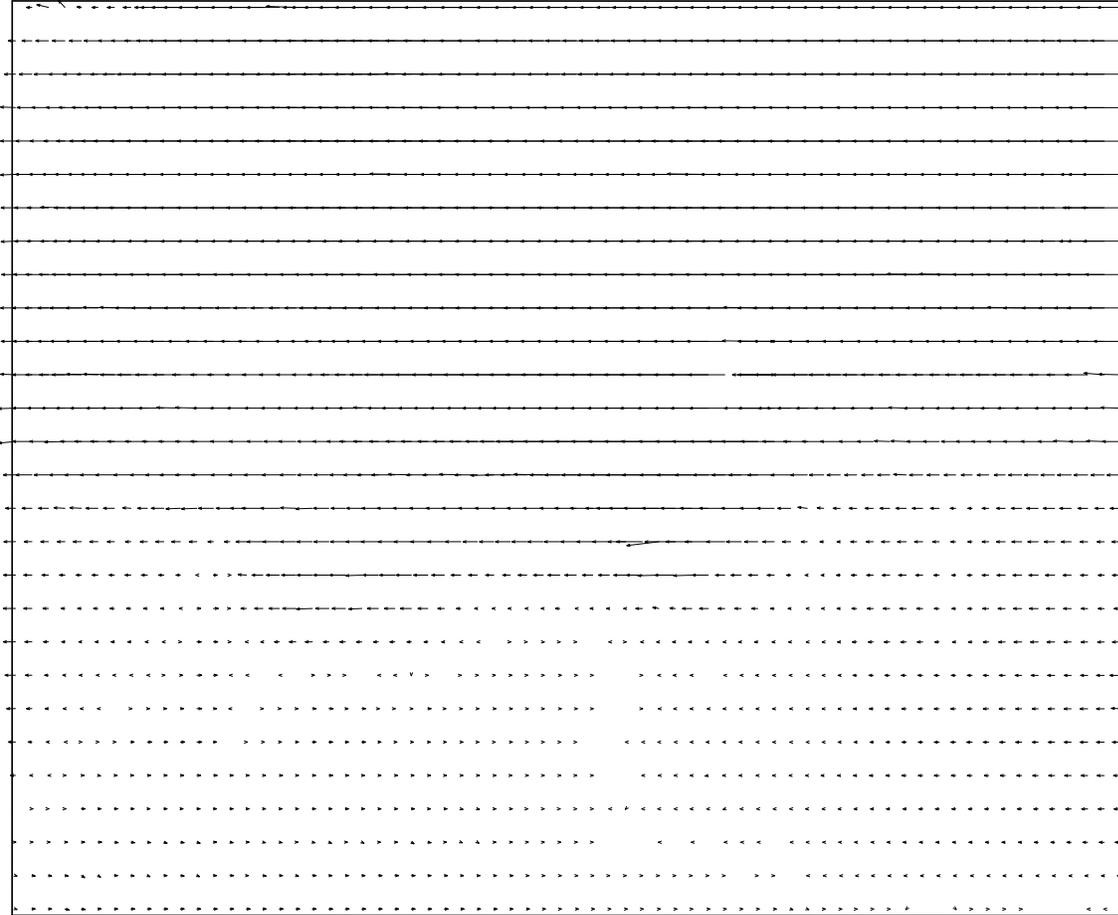


FIG. A.2 - *Champ dense pour "train" (facteur d'échelle 1 pour λ et γ entre les niveaux)*

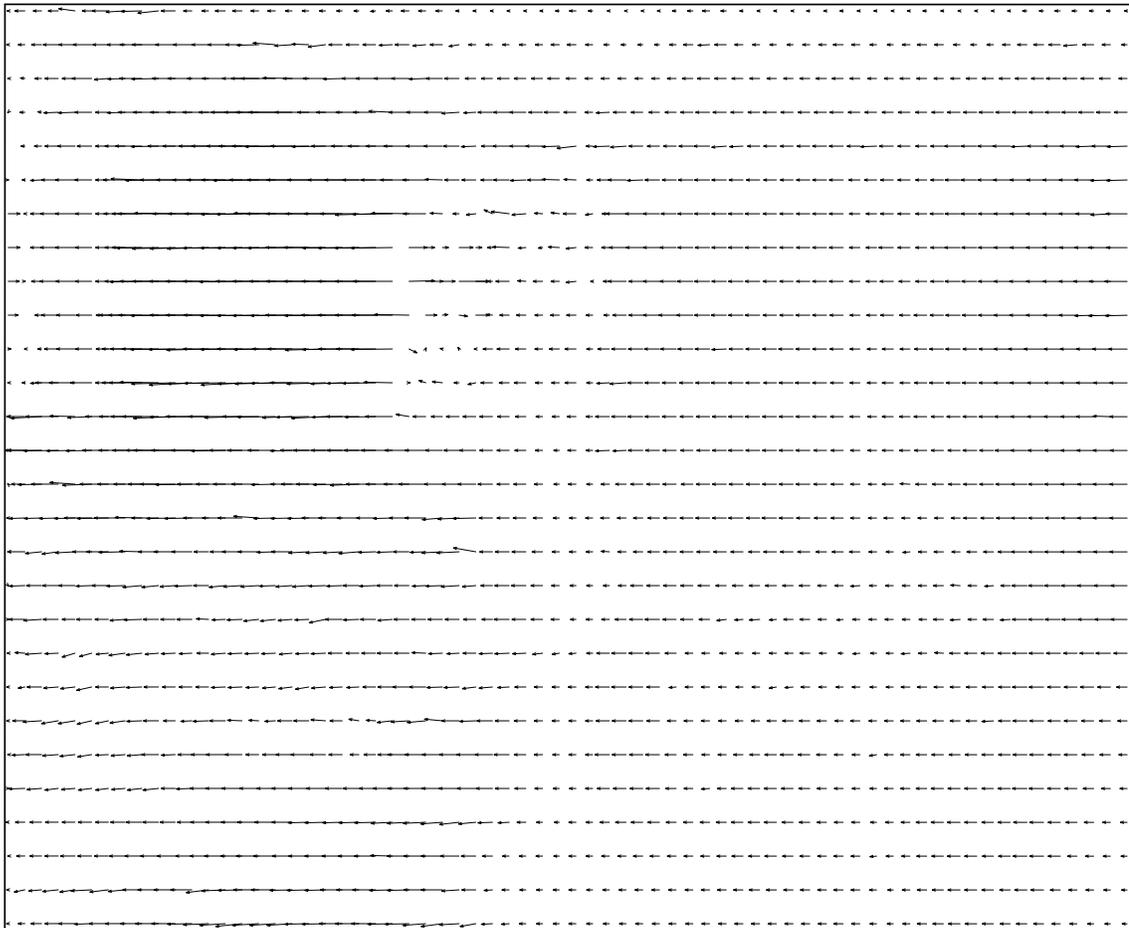


FIG. A.3 - *Champ dense pour "manege" (facteur d'échelle 2 pour λ et γ entre les niveaux).*

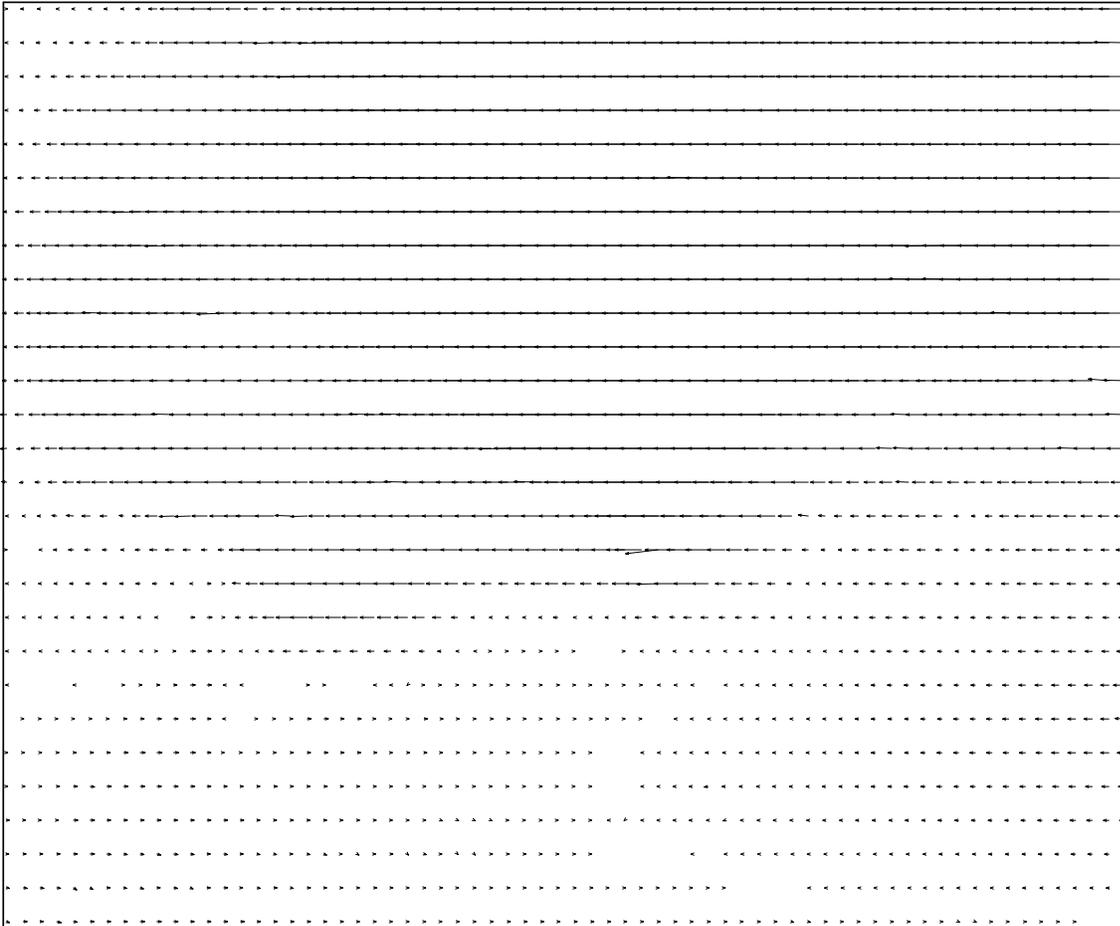


FIG. A.4 - *Champ dense pour "train" (facteur d'échelle 2 pour λ et γ entre les niveaux)*