

Méthode d'optimisation pour l'appariement de pixels d'images stéréoscopiques basée sur une métrique conjointe entropie-distorsion

A. Kadaikar, A. Mokraoui et G. Dauphin

L2TI, Institut Galilée, Université Paris 13 Sorbonne Paris Cité
99, avenue Jean-Baptiste Clément 93430 Villetaneuse, France

Résumé

Cet article s'intéresse au problème de la mise en correspondance de pixels d'images stéréoscopiques pour estimer la meilleure carte de disparité au sens du critère entropie-distorsion. Dans la majorité des cas, les correspondants sont choisis au sens de la minimisation de l'erreur quadratique moyenne retenue comme critère de distorsion. Cependant pour l'appariement d'un même pixel, il est possible que plusieurs disparités soient candidates puisqu'elles répondent uniquement au critère de distorsion minimale. En revanche le choix adopté pourrait ne pas être en adéquation avec la réduction du coût de codage. Pour y remédier, cet article propose une approche d'optimisation où la métrique habituelle est remplacée par une métrique entropie-distorsion de façon à ce que les disparités sélectionnées réduisent non seulement la distorsion de l'image reconstruite mais également l'entropie associée à la carte de disparité. L'estimation de la carte de disparité s'appuie sur la construction séquentielle d'un arbre afin d'éviter une recherche exhaustive tout en assurant de bonnes performances en termes d'entropie-distorsion. A une profondeur donnée dans l'arbre, les M meilleurs chemins retenus selon le critère entropie-distorsion sont ensuite prolongés pour construire de nouveaux chemins. Ces chemins sont triés selon la métrique entropie-distorsion pour n'en retenir que les M meilleurs. Le processus est itéré jusqu'à la lecture du dernier pixel à apparier. Les résultats de simulation montrent que notre algorithme fournit de meilleurs résultats en termes d'entropie-distorsion comparé à la programmation dynamique.

This paper focuses on the matching problem of pixels in stereoscopic images to estimate the best disparity map under entropy-distortion criterion. In most cases, matches are selected according to the minimum mean square error used as a distortion criterion. However for matching the same pixel, it is possible that several disparities are candidates since they meet only the minimum distortion criterion. In contrast, the choice may not be adopted in adequation with the coding cost reduction. To address this, this paper proposes an optimization approach where the usual metric is replaced by a entropy-distortion metric so that the selected disparities not only reduce the distortion of the reconstructed image, but also the entropy associated with the disparity map. The estimate of the disparity map is based on the sequential construction of a tree to avoid an exhaustive search while ensuring good performance in terms of entropy-distortion. At a given depth in the tree, the M-best paths chosen by entropy-distortion criterion are then extended to build new paths. These paths are sorted according to the entropy-distortion metric to retain only the M-best. The process is iterated until reading the last pixel to be matched. Simulation results show that our algorithm provides better results in terms of entropy-distortion compared to dynamic programming.

Mots clé : Image stéréoscopique, image 3D, mise en correspondance, algorithme à M-chemins, optimisation, entropie, distorsion.

1. Introduction

Une image stéréoscopique (ou image 3D) permet de recréer une impression de profondeur dans la scène observée. Elle est composée de deux vues à savoir la vue droite et la

vue gauche. Ces vues correspondent à la même scène et sont capturées à partir de deux points de vue légèrement différents. De ce fait, les images stéréoscopiques requièrent deux fois plus d'informations qu'une image 2D. Il est donc important de considérer la question du codage des images stéréoscopiques dans le cadre applicatif de stockage ou de transmission notamment pour des flux vidéos stéréoscopiques. Dans l'objectif de réduire la redondance inter-vue, certains

travaux proposent de reconstruire l'une des deux vues en exploitant la deuxième vue (considérée comme vue de référence) combinée à la carte de disparité associée à la vue à reconstruire. La carte de disparité s'avère en effet moins coûteuse en termes binaire qu'une des deux vues. C'est le processus d'appariement de pixels homologues dans les deux vues qui permet d'estimer la carte de disparité associée à l'une des deux vues. La qualité de la vue reconstruite en dépend fortement. Les méthodes développées doivent en effet faire face aux éventuelles changements de luminosité, aux zones texturées et également aux occultations.

Un état de l'art sur l'appariement d'images stéréoscopiques montre que plusieurs travaux ont déjà considéré la question de l'estimation des cartes de disparité. Le problème de l'appariement stéréoscopique est généralement formulé par le problème de minimisation d'une fonction d'énergie (coût global pour une approche globale) ou plusieurs fonctions d'énergie (coûts locaux pour une approche locale) [SS02, BBH03]. Les méthodes développées se différencient de manière générale par : (i) les primitives adoptées (par exemple pixels, points d'intérêts, segments, blocs, régions, bords) et leurs attributs (par exemple niveau de gris, contraste, composantes de couleur, position du segment, orientation du segment) ; (ii) le coût global de l'appariement (incluant également les coûts locaux d'appariement qui mesurent le degré de dissimilarité entre les deux primitives correspondantes) et le coût des contraintes (par exemple l'unicité, l'ordonnement, le lissage) ; (iii) la taille de la fenêtre d'appariement ; (iv) la zone d'agrégation (ensemble des pixels pour le calcul du coût de mise en correspondance) ; et (v) la méthode d'optimisation. L'objectif principal des méthodes d'optimisation est de minimiser le coût global ou local pour garantir le meilleur appariement des primitives. La recherche exhaustive (greedy search) du meilleur correspondant n'est pas intéressante en raison de sa charge de calcul trop élevée. La programmation dynamique a été l'une des premières méthodes d'optimisation à être exploitée en stéréo. Différentes versions ont été proposées. Dans [OK85], des contraintes de régularité ont été introduites pour optimiser les appariements notamment lorsque la lecture est réalisée selon un balayage en ligne. Dans [Vek05], Veksler a imposé des contraintes de lissage selon les deux directions horizontale et verticale dans le but de récupérer la vraie carte de disparité. D'autres méthodes d'optimisation ont également été exploitées, telles que la relaxation [Nas92], les graph cut [BVZ01, BG05], et la propagation de croyance [Sun03, TCC07].

Les travaux présentés dans cet article visent à développer un algorithme d'appariement pixel à pixel. La métrique de mise en correspondance s'appuie généralement soit sur l'erreur quadratique moyenne (EQM), soit sur l'erreur en valeur absolue moyenne (EAM). Notons que, pour un pixel donné, il est parfois possible d'obtenir un ensemble de candidats qui satisfait au critère imposé. Néanmoins, certains de ces candidats risquent de coûter plus chers que d'autres en termes de débit binaire. Pour résoudre cette question, nous proposons de remplacer la métrique traditionnelle (EQM, EAM ...) par une métrique conjointe entropie-distorsion. De ce fait, les disparités sélectionnées améliorent non seulement la qualité de l'image reconstruite mais également réduisent l'en-

trie de la carte de disparité. Ce problème est formalisé par une minimisation Lagrangienne où la fonction coût est choisie comme nouvelle métrique de mise en correspondance. Pour éviter une trop grande charge de calcul (recherche exhaustive), l'estimation de notre carte de disparité repose sur la construction séquentielle d'un arbre à M -chemins (c.a.d a breadth-first search algorithm). A chaque profondeur de l'arbre, l'algorithme prolonge les M meilleurs chemins jusqu'à l'appariement du dernier pixel.

Notre article est organisé comme suit. La section 2, après avoir introduit quelques notations, formalise notre problème d'optimisation. Notre algorithme d'appariement basé sur la métrique conjointe entropie-distorsion est ensuite développé. Les résultats de simulation sont ensuite discutés en section 3. La section 4 conclut notre article.

2. Algorithme proposé de mise en correspondance d'image stéréoscopique

L'objectif de notre algorithme est d'estimer la carte de disparité relative à la vue droite à partir de la vue gauche au sens du critère entropie-distorsion. La vue gauche est considérée ici comme vue de référence.

2.1. Formulation du problème d'appariement au sens du critère entropie-distorsion

Introduisons tout d'abord quelques notations avant de décrire notre algorithme. Supposons que les images sont rectifiées. I_g représente l'image de la vue gauche et I_d celle de la vue droite, toutes deux de taille $K \times L$. $I_d(i, j)$ (respectivement $I_g(i, j)$) correspond à l'intensité du pixel situé à la position (i, j) dans I_d (respectivement I_g). Notons $d(i, j)$ la disparité du pixel $I_d(i, j)$; et $\mathbf{d} = \{d(i, j) \text{ avec } i = 0, \dots, K-1; j = 0, \dots, L-1\}$ la carte de disparité qui minimise le coût global choisit comme étant l'erreur quadratique moyenne entre l'image originale associée à la vue droite et sa version reconstruite donnée comme suit :

$$E_{global}(\mathbf{d}) = \sum_{i=0}^{K-1} \sum_{j=0}^{L-1} (\hat{I}_d(i, j) - I_d(i, j))^2$$

avec $\hat{I}_d(i, j) = I_g(i, j + d(i, j))$, (1)

sous la contrainte de minimiser aussi l'entropie $H(\mathbf{d})$ de la carte de disparité. Ce problème est régi par le formalisme de Lagrange donné par l'équation suivante :

$$\hat{\mathbf{d}} = \operatorname{argmin}_{\mathbf{d}} J(\lambda, \mathbf{d}) = \operatorname{argmin}_{\mathbf{d}} (E_{global}(\mathbf{d}) + \lambda H(\mathbf{d})), \quad (2)$$

où λ représente le multiplicateur de Lagrange.

2.2. Algorithme d'optimisation sous-optimale

Pour résoudre l'équation (2), avec le souci non seulement de réduire la complexité de calcul mais également d'assurer une bonne estimation de la carte de disparité, nous nous sommes appuyés sur le principe de l'algorithme de décodage séquentiel à M -chemins exploité dans le domaine des communications pour estimer le flux transmis à travers un canal bruité selon un critère de vraisemblance [Jel69].

Pour limiter l'espace de recherche, sans trop pénaliser les

performances de notre algorithme, nous avons introduit une fenêtre d'appariement notée $W = [w_{min}, \dots, w_{max}]$ de taille N . Elle est positionnée sur chaque pixel à appairier. L'image de référence est lue en parcourant ses lignes de gauche à droite.

L'algorithme d'optimisation proposé s'appuie sur la construction séquentiel d'un arbre. A chaque profondeur t de l'arbre, un pixel à appairier y est associé. La profondeur t dépend de la ligne courante i , de la colonne courante j et du nombre de colonne L de l'image I_d :

$$t = i \times L + j \text{ avec } i = 0, \dots, K-1 \text{ et } j = 0, \dots, L-1. \quad (3)$$

Supposons qu'à la profondeur $t-1$, les M meilleurs chemins aient été retenus. A chaque chemin retenu est associé un coût $J_{t-1}^k(\lambda, d)$ tel que :

$$J_{t-1}^k(\lambda, d) = E_{t-1}^k + \lambda H_{t-1}^k \text{ avec } k = 1, \dots, M, \quad (4)$$

où E_{t-1}^k est la distorsion cumulée sur le chemin k jusqu'à la profondeur $t-1$; et H_{t-1}^k est une estimation de l'entropie relative aux choix des disparités effectués sur le chemin k jusqu'à la profondeur $t-1$. Notons qu'aux M meilleurs chemins sont associés M cartes de disparité notées S^k :

$$S^k = \{d_1^k, d_2^k, \dots, d_{t-1}^k\} \text{ avec } k = 1, \dots, M. \quad (5)$$

A la profondeur suivante (c.a.d. t), chacun des M -chemins retenus est prolongé par N nouvelles branches auxquelles sont associées une disparité w et une distorsion locale $E_{b_t}^w$ donnée par :

$$E_{b_t}^w = ((I_d(i, j) - I_g(i, j + w))^2 \text{ avec } w = w_{min}, \dots, w_{max}. \quad (6)$$

Les distorsions cumulées des $M \times N$ chemins prolongés sont alors mises à jour comme suit :

$$E_t^m = E_{t-1}^k + E_{b_t}^w \text{ avec } m = 1, \dots, M \times N \\ k = 1, \dots, M \text{ et } w = w_{min}, \dots, w_{max}. \quad (7)$$

Notons que nous ne disposons pas de la distribution des probabilités des disparités, information nécessaire pour le calcul de l'entropie H_t^k à la profondeur t

$$H_t^k = - \sum_{w=w_{min}}^{w_{max}} \hat{p}_t^k(d=w) \log_2(\hat{p}_t^k(d=w)) \\ \text{avec } k = 1, \dots, M \times N. \quad (8)$$

Pour y remédier, nous proposons une estimation des probabilités par une loi de mélange donnée par l'équation suivante :

$$\hat{p}_t^k(d = w | d_1^k, d_2^k, \dots, d_t^k) = C_a \times p_a(d = w) + \\ C_{exp} \times \hat{p}_{exp}^k(d = w | d_1^k, d_2^k, \dots, d_{t-1}^k) + \\ C_c \times p_c(d = w | d = d_t^k), \quad (9)$$

où les coefficients C_a , C_{exp} et C_c vérifient la relation suivante :

$$C_a + C_{exp} + C_c = 1. \quad (10)$$

$p_a(d = w)$ représente la probabilité *a priori* associée à la disparité w choisie parmi les disparités possibles définies dans la fenêtre de recherche à la profondeur t . Nous la représentons par une loi discrète uniforme :

$$p_a(d = w) = \frac{1}{N} \text{ avec } w = w_{min}, \dots, w_{max}. \quad (11)$$

$p_{exp}^k(d = w | d_1^k, d_2^k, \dots, d_{t-1}^k)$ est déduite des disparités retenues jusqu'à la profondeur $(t-1)$ sur le chemin k (c.a.d. $d_1^k, d_2^k, \dots, d_{t-1}^k$). Enfin $p_c(d = w | d = d_t^k)$ est la probabilité relative à la disparité choisie à la profondeur t . Elle est donnée par l'expression suivante :

$$p_c(d = w | d_t^k = w_c) = \begin{cases} 1 & \text{if } w = w_c \\ 0 & \text{if } w \neq w_c \end{cases} \quad (12)$$

En ce qui concerne les coefficients de l'équation (10), nous proposons de les paramétrer comme suit :

$$C_a = \frac{\beta a}{\beta a + b + c}; C_{exp} = \frac{b}{\beta a + b + c}; C_c = \frac{c}{\beta a + b + c}; \\ \text{avec}$$

$$a = K \times L - t; b = t \text{ et } c = 1. \quad (13)$$

β est un paramètre inférieur à 1 que nous fixons de façon à pouvoir contrôler l'importance attribuée aux probabilités *a priori* lors de l'estimation de notre carte de disparité. Notons que les paramètres " a " et " b " évoluent au fur et à mesure que notre algorithme progresse dans son traitement. " a " est choisi de façon à décroître alors que " b " augmente durant le traitement. Ceci se justifie par le fait qu'au démarrage l'algorithme s'appuie davantage sur p_a par manque d'informations. De ce fait l'influence de p_a s'estompe au détriment de la probabilité p_{exp}^k qui prend le relais.

Les coûts J_t^k (voir équation (4)) sont calculés puis classés selon un ordre croissant. L'algorithme retient alors parmi les $M \times N$ nouveaux chemins les M meilleurs. Les cartes de disparité respectives (c.a.d. S^k) sont mises à jour. L'algorithme itère ce processus jusqu'à l'appariement du dernier pixel de l'image. Le premier chemin correspondra alors à la meilleure carte de disparité en termes de compromis entropie-distorsion.

La Figure 1 donne un aperçu rapide de notre algorithme d'appariement avec $M = 2$ et $N = 5$. Pour plus de détails, les différentes étapes sont résumées ci-dessous :

Algorithme à M -chemins basé sur un critère conjoint d'entropie-distorsion pour l'appariement stéréoscopique.

Input : Image gauche I_g et image droite I_d de taille $K \times L$

Output : Carte de disparité dense estimée, relative à I_d ;

1. Initialisation des valeurs : λ ; M ; w_{min} ; w_{max} ;
- β ; $i = -1$ et $j = -1$;
2. Incrémenter de 1 le numéro de la ligne i ;
3. Incrémenter l'index j ;
4. Positionner la fenêtre sur le pixel $I_g(i, j)$;
5. Prolonger les M meilleurs chemins courants jusqu'à la profondeur t ;
6. Calculer les distorsions des $M \times N$ branches ;
7. Mettre à jour les distorsions des chemins prolongés ;
8. Estimer les probabilités de choisir les disparités pour chacun des chemins prolongés ;
9. Déduire l'entropie associée aux disparités de chaque chemin ;
10. Calculer le coût J_t^k pour chacun des chemins ;
11. Classer les chemins dans l'ordre croissant en fonction de leurs coûts respectifs ;
12. Sélectionner les M meilleurs chemins parmi les $M \times N$ chemins ;
13. Mettre à jour les M cartes de disparités des chemins retenus ;

14. Recommencer depuis l'étape 3 si $j < L$ sinon continuer ;
15. Recommencer depuis l'étape 2 si $i < K$ sinon continuer ;
16. Choisir la meilleure carte de disparité associée à I_d .

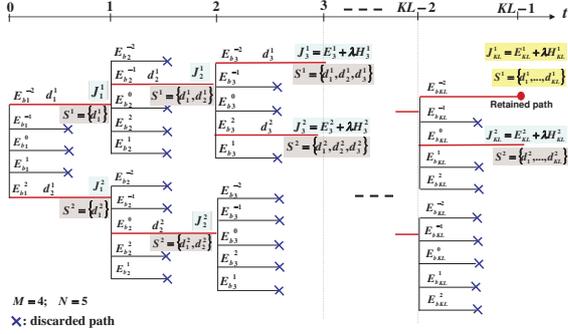


Figure 1: Principe de notre algorithme d'appariement stéréoscopique avec $M = 2$ et $N = 5$.

3. Analyse et discussion des résultats de simulation

Cette section présente les résultats de simulation fournis par notre algorithme d'optimisation pour la mise en correspondance d'images stéréoscopiques. Nos résultats sont comparés à ceux obtenus par l'algorithme de programmation dynamique de la toolbox "computer vision system" de Matlab [ref].

3.1. Image stéréoscopique de test : 'Tsukuba'

Les tests de simulation ont été réalisés sur les vues gauche et droite de l'image stéréoscopique "Tsukuba" de la base Middlebury [Mid]. La résolution spatiale de chaque image est de 288×384 pixels. La taille de la fenêtre d'appariement est fixée à 30 (c.a.d $N = 30$ avec $w_{min} = -15$ et $w_{max} = 14$) pour les deux algorithmes de mise en correspondance de pixels. La vue gauche est choisie comme vue de référence. La reconstruction de l'image droite repose sur la carte de disparité estimée soit par notre algorithme soit par l'algorithme de programmation dynamique. La Figure 2 montre l'image originale de la vue droite que nous cherchons à reconstruire à travers sa composante de luminance.

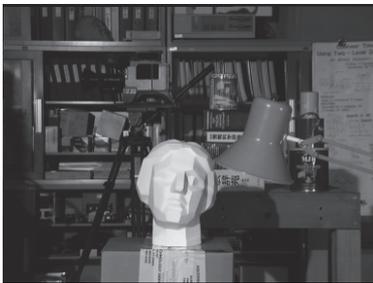


Figure 2: Image originale de la vue droite.

La Figure 3 est l'image reconstruite avec la carte de disparité estimée par la programmation dynamique basée sur des blocs de taille 1×1 . Cette taille de bloc a été retenue parce que l'image reconstruite présente un meilleur PSNR à savoir 38.77 dB comparé aux

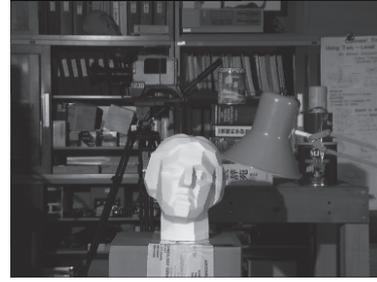


Figure 3: Image reconstruite : carte de disparité estimée par l'algorithme de programmation dynamique (PSNR=38.77 dB ; $H=3.65$ bpd).

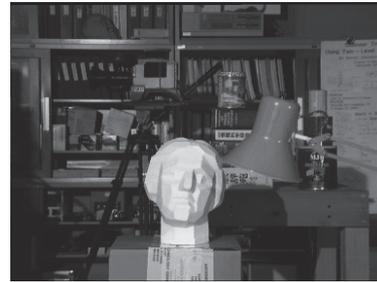


Figure 4: Image reconstruite : carte de disparité estimée par l'algorithme à M -chemins (PSNR=38.12 dB ; $H=1.92$ bpd).

autres tailles de blocs. La carte de disparité estimée (voir Figure 5) a une entropie égale à 3.65 bpd (bits par disparités).

La Figure 4 présente l'image reconstruite avec la carte de disparité estimée par notre algorithme à M -chemins avec les paramètres suivants : $M = 4$, $\lambda = 1100000$ et $\beta = 0.2$. La qualité de la reconstruction évaluée en termes de PSNR est de 38,12 dB. Ce PSNR est équivalent à celui de la programmation dynamique. En revanche notre carte de disparité a une entropie égale à 1.92 bpd (voir Figure 6). Pour une qualité équivalente, notre algorithme réduit le débit de 48%. Ceci est confirmée par l'analyse des histogrammes relatifs aux deux cartes de disparité fournis par les Figures 7 et 8.

Les courbes de la Figure 9 illustrent les performances débit-distorsion des deux algorithmes d'appariement stéréoscopique. La courbe marquée par '+', a été obtenue en jouant sur le paramètre λ . La courbe marquée par '*' a été réalisée en agissant sur la taille des blocs (de 1×1 jusqu'à 17×17). Nous observons clairement l'avantage de notre algorithme par rapport à l'algorithme de programmation dynamique. En effet, pour un débit par exemple de l'ordre de 2.32 bpd nous obtenons un gain de 19.8 dB en faveur de notre méthode d'optimisation.

3.2. Image stéréoscopique de test : 'Journal'

D'autres tests ont été réalisés sur l'image stéréoscopique 'Journal' de la base Middlebury [Mid]. La résolution spatiale des images des vues droite et gauche est de 383×435 pixels. Une fenêtre centrée de taille $N = 30$ ($w_{min} = -15$, $w_{max} = 14$) est choisie pour les deux algorithmes d'appariement. L'image originale de la vue droite à reconstruire est donnée par la Figure 10.

La Figure 11 présente la meilleure reconstruction possible

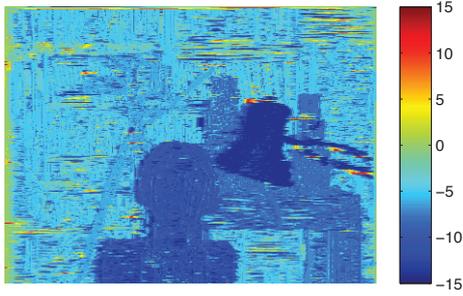


Figure 5: Carte de disparité estimée par la programmation dynamique ($PSNR=38.77$ dB ; $H=3.65$ bpd).

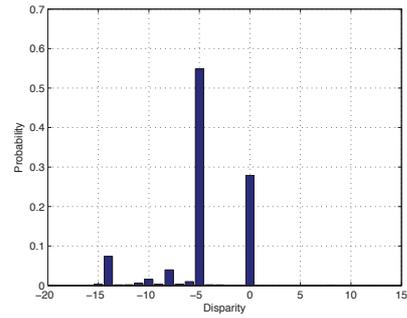


Figure 8: Histogramme des disparités estimées par l'algorithme à M -chemins ($PSNR=38.12$ dB ; $H=1.92$ bpd).

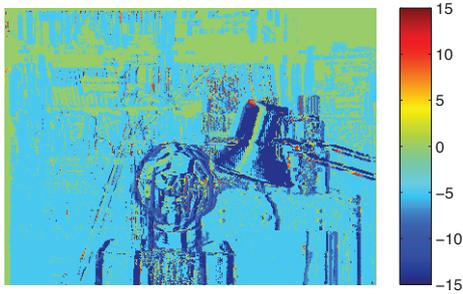


Figure 6: Carte de disparité estimée par l'algorithme à M -chemins ($PSNR=38.12$ dB ; $H=1.92$ bpd).

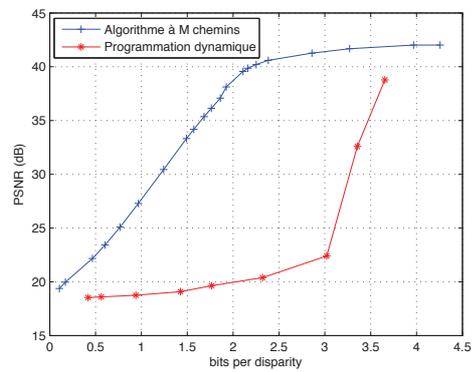


Figure 9: Courbes débit-distorsion.

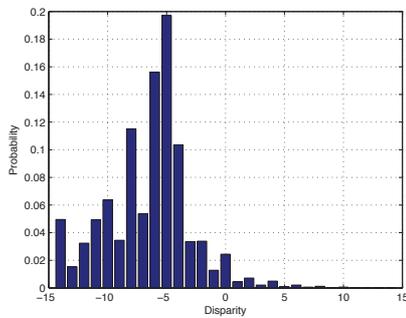


Figure 7: Histogramme des disparités estimées par la programmation dynamique ($PSNR=38.77$ dB ; $H=3.65$ bpd).

lorsque l'algorithme de programmation dynamique (avec des blocs de taille 1×1) est exploité pour estimer la carte de disparité. Le $PSNR$ relatif à cette reconstruction est de 27.06 dB avec une carte de disparité estimée ayant une entropie égale à 4.46 bpd (voir Figure 13).

La Figure 12 correspond à l'image droite reconstruite avec la carte de disparité estimée (voir Figure 14) par notre algorithme à M -chemins (avec $M=2$, $\lambda=10000000$ et $\beta=0.2$). Le $PSNR$ relatif à cette reconstruction est de 27.39 dB. L'entropie de notre carte de disparité est de 1.85 bpd. Pour un $PSNR$ équivalent, nous obtenons un gain en débit de 58.5% par rapport à la programmation dynamique. Les his-

rogrammes des cartes de disparité estimées confirment les avantages de notre méthode d'optimisation (voir Figure 15 et Figure 16).

Les courbes de la Figure 17 comparent les performances en termes de débit-distorsion. Les paramètres de notre algorithme ont été fixés comme suit : $M = 2$ et $\beta = 0.2$. En ce qui concerne la programmation dynamique, la taille des blocs varie de 1×1 à 17×17 . Nous constatons, pour un débit équivalent de l'ordre de 2.54 bpd, qu'un gain de 12 dB est atteint en faveur de notre méthode d'optimisation.

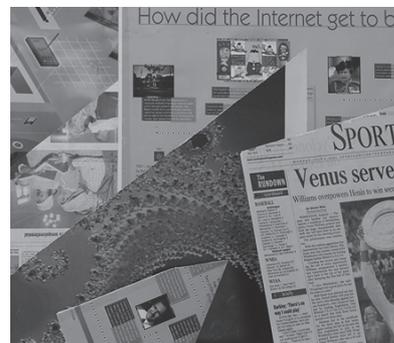


Figure 10: Image originale de la vue droite.



Figure 11: Image reconstruite : carte de disparité estimée par l'algorithme de programmation dynamique (PSNR=27.06 dB ; H=4.46 bpd).

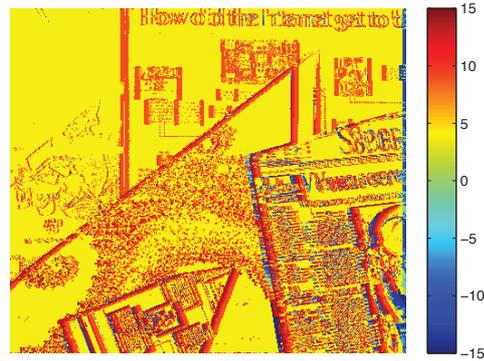


Figure 14: Carte de disparité estimée par notre algorithme à M -chemins (PSNR=27.39 dB ; H=1.85 bpd).

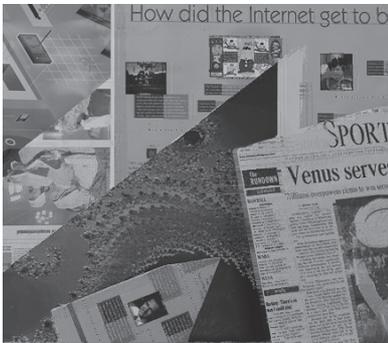


Figure 12: Image reconstruite : carte de disparité estimée par l'algorithme à M -chemins (PSNR=27.39 dB ; H=1.85 bpd).

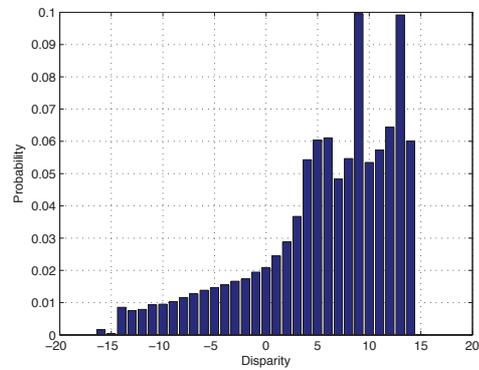


Figure 15: Histogramme des disparités estimées par la programmation dynamique (PSNR=27.06 dB ; H=4.46 bpd).

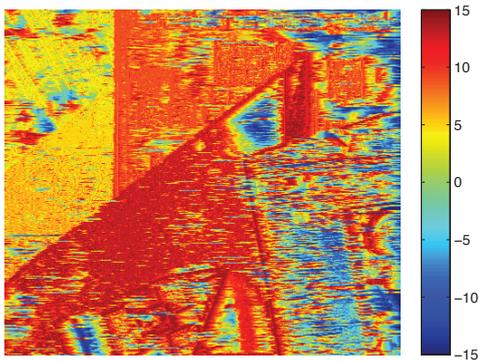


Figure 13: Carte de disparité estimée par la programmation dynamique (PSNR=27.06 dB ; H=4.46 bpd).

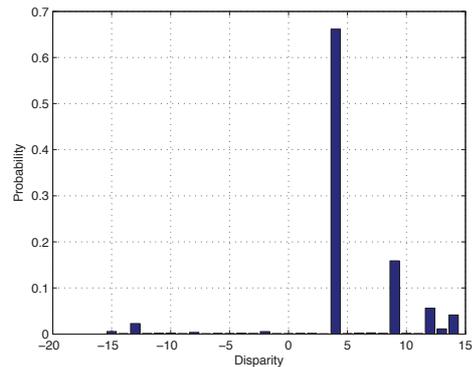


Figure 16: Histogramme des disparités estimées par l'algorithme à M -chemins (PSNR=27.39 dB ; H=1.85 bpd).

4. Conclusion

Nous avons développé un nouvel algorithme de mise en correspondance d'image stéréoscopique. Notre algorithme repose sur une métrique conjointe tenant compte à la fois de la qualité de l'image reconstruite (distorsion) mais également de l'entropie de la

carte de disparité. Ce problème s'appuie sur le formalisme de Lagrange qui est résolu par la construction séquentielle d'un arbre à M -chemins. Cette stratégie permet non seulement de réduire la complexité de calcul mais également d'offrir de bons résultats en termes d'entropie-distorsion comparé aux résultats fournis par l'algorithme

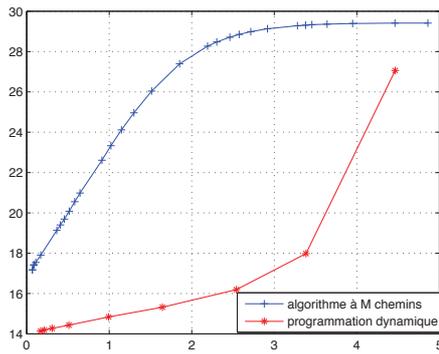


Figure 17: Courbes débit-distorsion.

classique de programmation dynamique. Des investigations sont en cours afin d'adapter notre algorithme à une mise en correspondance par bloc de façon à réduire significativement le débit tout en cherchant à préserver une bonne qualité de reconstruction.

Références

- [BBH03] BROWN M. Z., BURSCHKA D., HAGER G. D. : Advances in computational stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 25, Num. 8 (2003).
- [BG05] BLEYER M., GELAUTZ M. : Graph-based surface reconstruction from stereo pairs using image segmentation. *Videometrics VIII*. Vol. SPIE-5665 (janvier 2005).
- [BVZ01] BOYKOV Y., VEKSLER O., ZABIH R. : Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*. Vol. 23, Num. 11 (novembre 2001).
- [Jel69] JELINEK F. : Fast sequential decoding algorithm using a stack. *IBM Journal of Research and Development*. Vol. 13, Num. 6 (novembre 1969).
- [Mid] : <http://vision.middlebury.edu/stereo/data/>.
- [Nas92] NASRABADI N. M. : A stereo vision technique using curve-segments and relaxation matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*. Vol. 14, Num. 5 (mai 1992).
- [OK85] OHTA Y.-M., KANADE T. : Stereo by intra- and inter-scanline search using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 7, Num. 2 (mars 1985).
- [ref] : *Matlab toolbox* : <http://www.mathworks.fr/products/computer-vision/code-examples.html?file=%2Fproducts%2Fdemos%2Fshipping%2Fvision%2Fvideostereo.html>.
- [SS02] SCHARSTEIN D., SZELISKI R. : A taxonomy and evaluation of dense two-frame stereo correspondance algorithms. *International Journal of Computer Vision*. Vol. 47, Num. 1 (2002).
- [Sun03] SUN J. : Stereo matching using belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*. Vol. 25, Num. 7 (décembre 2003).
- [TCC07] TSENG Y.-C., CHANG N., CHANG T.-S. : Low memory cost block-based belief propagation for stereo correspondance. *IEEE International Conference on Multimedia and Expo* (2007).
- [Vek05] VEKSLER O. : Stereo correspondance by dynamic programming on a tree. *IEEE Conference Proceedings of Computer Vision and Pattern Recognition*. Vol. 2 (juin 2005).