



HAL
open science

Optimisation discrète et indices de stabilité appliqués à la stéréoscopie en contexte routier

Mathias Paget

► **To cite this version:**

Mathias Paget. Optimisation discrète et indices de stabilité appliqués à la stéréoscopie en contexte routier. Traitement du signal et de l'image [eess.SP]. Université Paris-Est, 2017. Français. NNT : 2017PESC1084 . tel-01800049

HAL Id: tel-01800049

<https://theses.hal.science/tel-01800049>

Submitted on 25 May 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



École doctorale MSTIC

Mathématiques et Sciences et Technologies de l'Information et de la Communication

Thèse de doctorat

**Optimisation discrète et indices de stabilité
appliqués à la stéréoscopie en contexte
routier**

Mathias Paget

Soutenu le 13 décembre 2017

Jury

Hugues TALBOT	Président	ESIEE
Pascal MONASSE	Directeur de thèse	École des Ponts
Anne VERROUST-BLONDET	Rapporteur	INRIA
Gabriele FACCILOLO	Rapporteur	ENS Cachan
Jean-Philippe TAREL	Examineur	IFSTTAR
Pierre CHARBONNIER	Examineur	Cerema



École doctorale MSTIC

Mathématiques et Sciences et Technologies de l'Information et de la Communication

Thèse de doctorat

**Optimisation discrète et indices de stabilité
appliqués à la stéréoscopie en contexte
routier**

Mathias Paget

Jury

Hugues TALBOT	Président	ESIEE
Pascal MONASSE	Directeur de thèse	École des Ponts
Anne VERROUST-BLONDET	Rapporteur	INRIA
Gabriele FACCILOLO	Rapporteur	ENS Cachan
Jean-Philippe TAREL	Examineur	IFSTTAR
Pierre CHARBONNIER	Examineur	Cerema

Remerciements

Je tiens à remercier mon directeur de thèse, Pascal Monasse, pour ces relectures qui m'ont permis d'être plus clair et plus précis. J'ai une pensée particulière pour Didier Aubert, qui a dirigé la première moitié de ma thèse. Je garde le souvenir d'un homme chaleureux et consciencieux.

Je tiens à remercier l'IFSTTAR et la direction scientifique de m'avoir permis de réaliser ma thèse dans les meilleures conditions, en trois ans et quelques mois.

Je remercie Anne Verroust-Blondet de l'intérêt porté à mon travail et d'avoir accepté d'être rapporteur. Son expertise dans le domaine de la modélisation en trois dimensions a apporté un autre point de vue sur mes travaux. Je remercie Gabriele Facciolo d'avoir accepté d'être le rapporteur de cette thèse. Ses connaissances en reconstruction et en optimisation ont donné lieu à de nombreuses remarques constructives. Et enfin, je remercie Hugues Talbot d'avoir accepté d'être dans mon jury.

Je remercie toute l'équipe du LEPSIS. J'ai pris plaisir à venir au laboratoire pendant ces trois années. En particulier, je remercie Eric Dumont, son directeur et Amélie Grichi, l'assistante de laboratoire, qui ont toujours fait leur possible pour me venir en aide. Ils m'ont permis de me concentrer sur mon travail en toute sérénité. Aussi, je remercie Roland Bremont et Sio-Song Ieng pour les réponses enrichissantes qu'ils ont pu me donner. Plus généralement, j'ai apprécié la diversité des activités de recherches du LEPSIS qui a contribué à me donner une plus grande ouverture sur le monde scientifique.

Je tiens surtout à remercier Jean-Philippe Tarel qui m'a encadré et suivi quotidiennement pendant ces trois années pour son investissement. Il a pris le temps de m'écouter, de me conseiller en faisant toujours preuve de pédagogie. Il m'a laissé m'approprier mon sujet et m'a encourager à approfondir la compréhension des méthodes utilisées. Je remercie aussi Pierre Charbonnier pour son encadrement durant la thèse.

J'en profite pour remercier les collègues que je n'ai pas cités précédemment : Celine, Fabrice, Stéphane, Léa, Erwan, Randy, Numa, Sami, Julien et enfin Geoffrey pour ses relectures.

Je remercie mes parents pour leur soutien inconditionnel dans tous les projets que j'ai pu entreprendre, pour les longues relectures et les paroles réconfortantes.

De dernier lieu, je remercie mon mari Ariel d'avoir été à mes côtés et de m'avoir donné la force et la confiance nécessaires pour réaliser ce travail.

Table des matières

Table des matières	i
Table des figures	vii
Liste des tableaux	xiii
Introduction	1
1 Reconstruction stéréoscopique	2
1.1 Problème direct et problème inverse	2
1.2 Spécificités des scènes routières	3
1.3 Formulation dans le cadre Bayésien	3
1.4 Approche par apprentissage	4
2 Méthode d'optimisation	5
2.1 Optimisation par coupure de graphe	6
2.2 Optimisation par programmation dynamique	7
3 Indices de confiance dans la solution	7
3.1 Incertitude des données	7
3.2 Caractériser la solution	8
4 Plan de la thèse	8
1 Optimisation par coupe de graphe	9
1 Représentation d'une fonction par un graphe	9
1.1 Coupure de graphe	9
1.2 Fonctions pseudo-booléennes	10
1.3 La sous-modularité et représentation en graphe	11
1.4 Contraction du graphe	12
2 Minimisation d'un problème par coupure de graphe	12
2.1 Recherche du flot maximal	13
2.1.1 Flot maximal/Coupe minimale	13
2.1.2 Poussage de flot	13
2.1.3 Algorithme Maxflow	14

2.2	Poussage de flot et FPB	14
2.2.1	Chemin augmentant et somme alternée	14
2.2.2	Problème de maximisation-minimisation	14
3	Extension de la formulation par coupure de graphe	15
3.1	Résolution exacte des problèmes multi-labels	16
3.1.1	Représentation du problème par un graphe	17
3.1.2	Contraintes sur la fonction de régularisation	19
3.1.3	Construction	19
3.1.4	Cas particuliers	20
3.2	Résolution approchée des problèmes multi-labels	22
3.2.1	Fusion binaire	22
3.2.2	Changement de variables	23
3.2.3	Cas de la fusion ternaire	23
3.3	Minimisation approchée des fonctions quadratiques (QPBO)	24
3.3.1	Roof duality	24
3.3.2	Symétrie	25
3.3.3	Autarcie et persistance	26
3.3.4	QPBO	27
3.3.5	Les heuristiques	28
3.4	Réduction d'ordre	28
3.4.1	Méthode par substitution et équation de contrainte	29
3.4.2	Méthode de substitution par une fonction équivalente	29
3.4.3	Autres méthodes	30
4	Schéma d'optimisation des problèmes multi-label	30
4.1	Garantie de décroissance de l'énergie	31
4.1.1	Cas Sous-modulaire	31
4.1.2	Fusions approchées	31
4.2	Schémas avec fusions binaires sous-modulaires	32
4.2.1	Fusions dans le cas convexe	32
4.2.2	α - β -Échange	32
4.2.3	α -Expansion	33
4.3	Schéma d'optimisation par fusion approchée	34
4.3.1	α -Expansion Quantifiée	34
4.3.2	Expansion-Saut alternés	34
4.3.3	Autres schémas	34
5	Conclusion	35

1	Formulation par coupure de graphe appliquée au dé-bruitage	37
1.1	Modélisation du problème	37
1.2	Comparaison des fonctions de régularisation	38
1.2.1	Cas où la fonction r est convexe	39
1.2.2	Sous-additivité et sur-additivité des fonctions de régularisation	40
1.2.3	Réglage du paramètre de régularisation	40
1.3	Comparaison entre les schémas d'optimisation approchés et la résolution exacte	43
1.3.1	Cas où r est la fonction valeur absolue	43
1.3.2	Cas où r est la fonction valeur absolue lissée	44
1.4	Comparaison des performances des différents schémas d'optimisation	45
1.4.1	Comparaison entre les schémas d'optimisation	46
1.4.2	Comparaison des temps de convergence des différents schémas	47
1.5	Conclusion	48
2	Reconstruction stéréoscopique par coupure de graphe	48
2.1	Géométrie rectifiée	48
2.2	Modèles d'attache aux données	49
2.2.1	Méthodes par pixel	50
2.2.2	Méthodes sur un voisinage	51
2.3	Modèles de régularisation	52
2.3.1	Régularisation du premier ordre	53
2.3.2	Régularisation du deuxième ordre	53
2.4	Modèle de reconstruction par GC	54
2.4.1	Optimisation d'un volume par GC	54
2.4.2	Optimisation d'une carte de disparité	55
2.4.3	Validation croisée entre les cartes de disparité	58
3	Expérimentations de reconstruction	58
3.1	Les jeux de données	59
3.1.1	Base de Middlebury	59
3.1.2	Kitti	59
3.2	Cas où la fonction r de la régularisation est la valeur absolue	60
3.2.1	Sur des scènes d'intérieur de la base Middlebury	61
3.2.2	Sur des images de la base KITTI	62
3.2.3	Cas de mauvaise reconstruction avec α -Expansion	64
3.3	Cas où la fonction r est une fonction de Potts lissée	65
3.3.1	Sur des scènes d'intérieur de la base Middlebury	65
3.3.2	Sur la base KITTI	66
4	Conclusion	66

3	Programmation dynamique appliquée à la reconstruction stéréoscopique	69
1	Optimiser un champ de Markov du premier ordre par DP	69
1.1	Optimisation d'un champ de Markov 1D du premier ordre par DP	69
1.1.1	Maximisation de la probabilité a posteriori	70
1.1.2	Minimisation d'une énergie	71
1.2	Application à des champs non-1D	71
1.2.1	Les champs acycliques	72
1.2.2	Optimisation d'un graphe acyclique par propagation de croyance (BP)	72
1.2.3	Décomposition en champs acycliques	72
2	Appariement semi-global, <i>Semi-Global Matching</i> (SGM)	73
2.1	La méthode SGM	73
2.2	Variantes de la méthode	76
2.2.1	Variante CAT	76
2.2.2	Variante MGM	77
2.2.3	Proposition de modification des variantes en exploitant l'anisotropie des méthodes	78
2.3	Raffinement de la solution par DP	78
3	Expérimentation	80
3.1	Effet du nombre de directions prises en compte dans SGM et du raffinement de la solution	80
3.1.1	Comparaison des méthodes SGM ₄ , SGM ₈ et SGM ₁₆	80
3.1.2	Effet des raffinements	85
3.2	Variantes des méthodes SGM	86
3.2.1	Méthode CAT et ses variantes	86
3.2.2	Méthode MGM et ses variantes	86
3.3	Comparaison entre CG et DP	89
3.3.1	Solution SGM comme approximation de la solution globale en 4-connexité	89
3.3.2	Différences entre les énergies utilisées dans GC et SGM	91
3.3.3	Temps de calcul	95
4	Conclusion	95
4	Stabilité de la solution par optimisation discrète et application à reconstruction stéréoscopique	97
1	Indices et méthodes d'évaluation de l'indice	97
1.1	Estimation locale de la variance	97
1.1.1	Dans le cas du dé-bruitage avec un bruit gaussien	98
1.1.2	Dans le cas général	98
1.2	Évaluation d'un indice de stabilité	99

1.2.1	Évaluation d'un détecteur binaire	99
1.2.2	Indice de la variance de la solution	100
1.2.3	Validation indirecte	102
2	Proposition d'indice de confiance par optimisation discrète	102
2.1	Proposition d'indice de confiance par programmation dynamique	102
2.1.1	Construction d'un indice à partir des messages	102
2.1.2	Valeur de l'indice SGM dans le cas gaussien	105
2.2	Proposition d'indice de confiance par coupure de graphe	107
2.2.1	Indice de stabilité	107
2.2.2	Adaptation du problème de poussage de flot	108
2.2.3	Borne inférieure de l'indice	109
2.2.4	Extension à la méthode d'Ishikawa	110
2.2.5	Extension à la méthode α -Expansion	110
3	Évaluation des indices	111
3.1	Indices issus de la programmation dynamique	111
3.2	Indices issus de la coupure de graphe	115
3.3	Indices pour le raffinement de la solution	115
4	Conclusion	119
Conclusions et perspectives		121
1	Résultats et contributions	121
1.1	Extension de la classe de fonctions optimisées par coupure de graphe	121
1.2	Qualité de l'approximation des schémas d'optimisation par fusion binaire	122
1.3	Interprétation de SGM par rapport à un champ de Markov formulé sur toute l'image	122
1.4	Modification des directions privilégiées dans la méthode SGM	122
1.5	Indice de confiance fondé sur SGM	122
1.6	Indice de confiance fondé sur α -expansion	123
2	Perspectives	123
2.1	Formulation de la méthode SGM dans le cadre Bayésien	123
2.2	Impact de la résolution de l'image sur la méthode SGM	123
2.3	Appliquer l'indice dans une chaîne de traitements	124
2.4	Utilisation des indices dans un cas maîtrisé	124
2.5	Utilisation des modèles obtenus par apprentissage	124
Bibliographie		125
Contributions		131
Résumé		
Abstract		

Table des figures

1	Scène acquise par une caméra. A droite, la scène en 3D, au centre la caméra, à gauche l'image résultante. Estimer l'image en sortie de la caméra, à partir de la scène, est le problème direct. Estimer la scène acquise à partir de l'image est le problème inverse.	2
2	Image gauche du couple 11 de la base KITTI 2015. Nous pouvons voir que la scène est relativement plane et l'intervalle de profondeur entre les objets les plus lointains et les plus proches est important.	3
3	La même image radar est présentée avec deux orientations différentes. L'interprétation des volumes est plus difficile sur la première image : nous avons tendance à interpréter la première image "en creux" et la seconde "en bosse". Cela vient du fait que dans les scènes naturelles, la lumière vient du haut et que cet <i>a priori</i> est pris en compte dans notre interprétation 3D de la scène.	5
1.1	Exemple de graphe orienté non pondéré, $G = (\{W, X, Y, Z\}, \{(W, X), (W, Z), (X, W), (X, Z), (Y, W)\})$	10
1.2	Structure élémentaire de graphe-ST (source, puits et deux nœuds)	10
1.3	Schéma d'un graphe après poussage de flot et deux coupes minimales.	15
1.4	Exemple d'un graphe avec trois coupes minimales. Les coupes minimales du graphe sont des coupes nulles du graphe résiduel (cf. Section 2.2.2)	16
1.5	Exemple de poussage de flot appliqué à la fonction de (1.7a). Les termes avant et après application de la somme alternée sont écrits en rouge et en bleu.	16
1.6	Structure de graphe permettant de représenter un problème multi-labels, l'exemple correspond à une chaîne de 4 pixels (p, q, r, s) dont les labels prennent des valeurs entre 0 et 4. Les liens en gris sont des arcs de capacité infinie du terme de contraintes, les liens en bleu correspondent aux arcs mis en jeu dans les termes U de P_G et les liens en rouge correspondent aux arcs en jeu dans les termes V de P_G . 18	18
1.7	Graphe multi-couches pour deux labels prenant leurs valeurs entre 0 et 4 avec une fonction de régularisation de la différence de label. Les capacités sont décomposées sur deux graphes différents pour une meilleure lisibilité. L'opérateur Δ^2 de la moitié de la dérivée seconde est définie dans (1.43).	21
1.8	Graphe élémentaire pour une fusion binaire, la valeur de la coupe est égale à la valeur de l'énergie à la constante $R_{pq}(l_p^2, l_q^1)$ près.	23
1.9	Graphe élémentaire pour un saut ternaire, à la constante $-R(l_x^-, l_y^-)$ près	24
1.10	$P(x_1, x_2) = x_1x_2 + x_1\bar{x}_2$ admet Q_1 et Q_2 comme relaxation qui atteignent la même borne ($RD = 0$) mais Q_1 donne la solution pour $P(\diamond, \diamond)$ et Q_2 donne $(0, \diamond)$. 27	27
2.1	La base de test, pour le dé-bruitage, est composée de l'image Lena et des quatre images ci-dessus. De gauche à droite : Pepper, Barbara, Boat, House.	38

2.2	Image originale et image bruitée (bruit gaussien indépendant d'écart-type $\sigma = 20$ et agrandissements centrés sur une zone d'intérêt	38
2.3	Résultats de dé-bruitage de l'image Lena sur la première ligne, et un agrandissement sur la deuxième ligne, pour trois fonctions de régularisation : r la fonction carré (première colonne), $ r $ la valeur absolue (deuxième colonne) et r la valeur absolue lissée (troisième colonne). Nous remarquons que la fonction carré donne des solutions floues et que valeur absolue donne un effet en marche d'escalier sur la joue de Lena. Ces effets ne sont plus visibles dans le cas de la valeur absolue lissée.	39
2.4	Comparaison des résultats de dé-bruitage du signal pour trois modèles. Certaines valeurs dans la solution sont fixées. Les points correspondant dans le signal sont représentés par des carrés noirs. Les autres points du signal, représentés par des triangles blancs, sont laissés libres dans la solution. La fonction d'attache aux données est $f(x) = x^2$ pour les trois modèles. La fonction de régularisation du modèle est, de haut en bas, $r_1(x) = 2x^2$, $r_2(x) = 4\sqrt{ x }$ et $r_3 = 10SEF_{-2,2}(x)$. Ces fonctions sont représentées dans la Figure 2.5. Avec le premier modèle, en haut, la fonction convexe donne une solution très lisse où le contour entre $x = 0$ et $x = 2$ est lissé. Avec deuxième modèle, au milieu, la fonction concave donne une solution constante par morceaux avec des ruptures nettes. Les contours sont préservés mais l'effet de marche d'escalier apparaît dans les zones de dégradé entre $x = 4$ et $x = 10$. Avec le dernier modèle, en bas, la forme de la fonction de régularisation permet à la fois de préserver les contours et de suivre les dégradés progressifs.	41
2.5	Fonctions de régularisation r utilisées dans l'exemple de la Figure 2.4	42
2.6	Résultats de remise à l'échelle de la régularisation en utilisant (2.8). Sur la première ligne les solutions où r est la valeur absolue avec les paramètres $\lambda = 20$, $\lambda = 50$, $\lambda = 100$, respectivement, et sur la deuxième ligne, les résultats avec régularisation avec la fonction $SEF_{0.5,4}$. Les paramètres calculés sont $\hat{\lambda}_2 = 83$, $\hat{\lambda}_2 = 212$, $\hat{\lambda}_2 = 430$. Visuellement l'intensité de la régularisation paraît similaire dans les trois cas entre le haut et le bas.	43
2.7	Évolution de la valeur de l'énergie au cours des itérations pour les trois méthodes d'optimisation par GC sur la base de test de dé-bruitage dans le cas du schéma d'optimisation Alterné-C. Les énergies sont données à une constante près et seule la première itération de chaque méthode est en dehors du graphique.	47
2.8	Couple d'images et leurs points épipolaires e^1 et e^2 . Pour une position m^1 de l'image I^1 , l'homologue de ce point, m^2 , est contraint d'être sur une droite dans l'image I^2 , appelée ligne épipolaire. Suivant la profondeur du point M le long de la droite (C^1, m^1) , le point m^2 se déplace sur la ligne épipolaire.	49
2.9	Vue de dessus d'un couple stéréoscopique en géométrie rectifiée. Nous observons qu'il existe une relation géométrique entre la profondeur du pixel et la différence de position horizontale en pixel. Cette différence est appelée la disparité.	50
2.10	Structure de graphe pour la reconstruction stéréoscopique de deux lignes en correspondance épipolaire dans les image I^1 et I^2 rectifiées. Les disparités croissent selon la première diagonale. Les lignes de vue de l'image 1 correspondent au lignes horizontales dans le graphe (en vert) et les lignes de vue de l'image 2 correspondent aux verticales (en violet), les contraintes le long des lignes de visée sont représentées en gris. Les liens de régularisations entre les lignes de vues correspondent aux diagonales (en rouge).	56
2.11	Structure de graphe présentée dans la Figure 2.10 dans le cas où les coûts $D_p(-1)$ sont infinis et après simplification.	57

2.12	Couples d'image de la base de test et la vérité de la carte de disparité dans la géométrie de l'image gauche. De gauche à droite et de haut en bas, le couple Tsukuba et les couples Venus, Cones et Teddy de la base Middlebury.	59
2.13	Image gauche du couple de la base KITTI15 et agrandissements de détails de l'image. Les aberrations chromatiques sont particulièrement visibles à la frontière de zones très claires ou très foncées.	60
2.14	Exemple d'un couple d'image de la base KITTI et de l'écart d'intensité entre les pixels de l'image gauche et de l'image droite pour un appariement à la disparité gauche vraie. Les écarts inférieurs à 5, inférieurs à 10 et supérieurs à 10 sont représentés respectivement en bleu, vert et rouge. Une large proportion des pixels est dans la dernière catégorie. Cela rend difficile une approche d'appariement par pixels. Dans cette image, cette zone correspond principalement à une différence de réflexion spéculaire sur la route. Nous remarquons aussi que les écarts inférieurs à 5 correspondent à des pixels saturés ou proches de la saturation. Ainsi, ces bonnes correspondances sont dues en partie à la faiblesse de la dynamique de l'image dans ces zones.	61
2.15	Carte de disparité obtenue sur le couple Cones au cours des itérations 2, 3, 4 et 5 de Alterné-C. Les changements par rapport à l'itération précédente sont en rouge.	62
2.16	Évolution de l'énergie au cours de l'optimisation pour la reconstruction du couple d'image Cones.	63
2.17	Évolution de l'énergie au cours de l'optimisation pour le couple 0 de la base KITTI15.	64
2.18	Image de disparité du couple 199 de la base KITTI15 optimisée par α -Expansion et par la méthode d'Ishikawa. La solution α -Expansion est très éloignée de la solution exacte du problème. De grandes portions de la solution sont planes et ne décrivent pas la scène qui a été acquise.	65
3.1	Exemple de chaîne de Markov 1D avec 5 nœuds.	69
3.2	Exemple de calcul d'une valeur Φ . La valeur $\Phi_{a-1 \rightarrow b}(0)$ est obtenue par calcul du maximum selon l_a du produit de $P(l_a)\Phi_{a \rightarrow b}(l_a)$ par $P(l_a, 0)$	71
3.3	Illustration de graphes acycliques sur la première ligne et de graphes avec cycles sur la seconde ligne. Seuls les problèmes qui s'écrivent sous la forme d'un graphe acyclique peuvent être traités par DP.	73
3.4	Chemins pris en compte dans l'étape d'agrégation de SGM pour un pixel donné (en rouge) pour 4, 8 et 16 directions. Les pixels pris en compte, l'ensemble noté χ_p^e , sont représentés en bleu-vert. Plus le nombre de directions est grand, plus la couverture de l'image est grande.	74
3.5	Images de disparité obtenues par optimisation selon les lignes à gauche et selon les colonnes à droite sur la même paire. Le modèle est une attache aux données en carré tronqué au-delà de la valeur 18 et la régularisation est une fonction Potts lissée de paramètre (60, 600).	74
3.6	Images de disparité obtenues par méthode SGM en prenant en compte 4, 8 et 16 directions. Le modèle est le même que dans la Figure 3.5. En comparaison, la solution devient plus lisse à mesure que le nombre de directions prises en compte augmente. Cela permet de réduire les artefacts de la méthode, mais détériore la reconstruction des objets fins.	76
3.7	Représentation des parties de l'image mise en jeu lorsque deux directions d'agrégation sont prises en compte simultanément comme dans les méthodes CAT et MGM	76

- 3.8 La figure représente un problème à quatre variables binaires représenté par un graphe correspondant à la fonction pseudo-booléenne $P(a, b, c, d) = \bar{a} + \bar{b} + ab + \bar{a}\bar{b} + ab + \bar{c}\bar{d} + 10(c + d + ac + \bar{a}\bar{c} + bd + \bar{b}\bar{d})$. Les valeurs d'attache aux données pour le label 0 et le label 1 sont données à coté de chaque nœud et les coûts de régularisation sont donnés sur les arcs. Nous prenons en compte quatre directions. Les coûts calculés par la méthode SGM et la méthode CAT pour deux valeurs de label et pour les quatre pixels sont donnés à droite de la figure. Nous pouvons voir que les valeurs calculées par CAT sont bien inférieures à celle calculées par SGM, néanmoins SGM donne l'affectation $P(0, 0, 0, 0) = 2$ et CAT donne l'affectation $P(1, 1, 0, 0) = 20$. La solution de CAT n'atteint pas nécessairement une énergie globale inférieure à celle de la solution de SGM. 77
- 3.9 Contributions des pixels au pixel rouge pour deux valeurs de (a) dans (3.27). Une contribution de 1 correspond à un bleu profond. Les valeurs suivant selon la perpendiculaire à la première diagonale suivent une loi binomiale. Dans le cas où le paramètre a est proche de 0.5, les pixels situés sur la première diagonale ont une plus forte contribution. Lorsque le paramètre est proche de 1, ce sont les pixels à la verticale qui ont une plus forte contribution. 78
- 3.10 Image gauche et extrait de carte de disparité (cycliste) obtenue pour le couple 2 de la base KITTI15 par la méthode SGM₄ après différents raffinements : (b) et (c) pour Opt.HV, (e), (f) et (g) par filtrage médian. 79
- 3.11 Extrait de l'image de disparité obtenue pour le couple 7 de la base KITTI par les méthodes SGM₄ SGM₈ et SGM₁₆. L'extrait correspond à un cycliste prenant un virage. 79
- 3.12 Cinq des dix couples de la base Middlebury ('Adirondack', 'ArtL', 'Motorcycle', 'Piano', 'Pipes', 'Playroom'). Sont présentés l'image gauche, image droite et la vérité de carte de disparité. 81
- 3.13 Cinq des dix couples de la base Middlebury ('Playtable', 'Recycle', 'Shelves', 'Teddy'). Sont présentés l'image gauche, image droite et la vérité de carte de disparité. 82
- 3.14 Dix premiers couples de la base KITTI15. Sont présentés l'image gauche et la carte de disparité "non-occ" 83
- 3.15 Influence sur nombre de directions prises en compte dans SGM (4, 8 et 16) sur la reconstruction du couple Adirondack de la base Middlebury, ainsi qu'un zoom. Nous observons que la solution est plus lisse et que les artefacts de la méthode SGM sont atténués lorsque le nombre de directions augmente. 84
- 3.16 Taux d'erreur des méthodes SGM₄, SGM₈ et SGM₁₆ sur les 10 couples de la base Middlebury en haut, et sur 10 premiers couples de la base KITTI15 en bas. Le taux d'erreur est la proportion de pixel à plus d'une disparité de la vérité de la base Middlebury (err.M) et à plus de 3 disparités de la vérité de la base KITTI (err.K). Le taux d'erreur err.K baisse à mesure que le nombre de directions augmente. Les résultats sont plus contrastés sur la base Middlebury. 84
- 3.17 Résultats de reconstruction pour les méthodes SGM, CAT et CAT₀ sur la première ligne ainsi que les variantes CAT₄, CAT₁₆ et CAT₃₂ sur la deuxième ligne. La solution CAT comporte des artefacts en ligne suivant la perpendiculaire aux contours. La méthode CAT₀ présente moins d'artefact au niveau des contours, mais des taches d'erreur supplémentaires. À mesure que le paramètre K augmente, la solution tend vers la solution de la méthode SGM. 87

3.18 Taux d’erreur des méthodes SGM dans 4 directions, CAT, CAT₀, CAT₄, CAT₁₆, CAT₃₂ et sur 10 couples de la base Middlebury (err.M) en haut, et sur les 10 premiers couples de la base KITTI15 (err.K) en bas. Sur la base Middlebury, la méthode CAT détériore les résultats par rapport à SGM. Les variantes de CAT donnent les meilleurs résultats pour $K = 8$ et surpasse ceux la méthode SGM. Sur la base KITTI, toutes les méthodes fondées sur CAT donnent de moins bons résultats que SGM. Les variantes de CAT donnent alors de meilleures résultats à mesure qu’elle s’approche de la méthode SGM. 87

3.19 Résultats de reconstruction des méthodes SGM, MGM, MGM_{0.9}, MGM_{0.8} et MGM_{0.7}. Les différents solutions forment un continuum entre la solution de la méthode SGM de tramage, et MGM qui présente des erreurs plus localisées. 88

3.20 Taux d’erreur des méthodes SGM, MGM, MGM_{0.9}, MGM_{0.8}, MGM_{0.7} et MGM_{0.6} sur 10 couples de la base Middlebury (err.M) en haut, sur les 10 premiers couples de la base KITTI15 (err.K) en bas. Sur la base Middlebury, la méthode MGM donne de meilleurs résultats que SGM sauf pour les couples ArtL et Motorcycle. Le meilleur résultat moyen est obtenu pour MGM_{0.8} 88

3.21 Image gauche du couple 199 de la base KITTI15 et cartes de disparité avec la méthode Ishikawa, α -Expansion, SGM₄ et SGM₁₆. Le dernier résultat est obtenu par α -Expansion, en initialisant par la solution de SGM dans 16 directions. 90

3.22 Couple 2 de la base KITTI12 sur la première ligne. Résultat de reconstruction avec SGM₄ et SGM₁₆ sur la deuxième ligne, et par la méthode d’Ishikawa avec le même paramètre de régularisation puis avec le paramètre λ divisé par 2. 92

3.23 Image gauche et cartes de disparité obtenues sur la paire 199 de la base KITTI15. Les résultats sont données de haut en bas par énergie globale en 4-connexité décroissante. Les méthodes sont de haut en bas, B11x11, SGM₄, SGM₈, SGM₁₆ et par GC (schéma Alterné-C). 94

4.1 Pour le couple 56 de la base KITTI15 affiché sur les deux premières images, la vérité d’occultation est obtenue par validation croisée (cf. Section 2.4.3 du Chapitre 2) et affichée en troisième position : en blanc, les pixels dont la disparité n’est pas cohérente entre les deux cartes de disparités. Cette vérité est comparée à ce qui est obtenu par un seuillage à 10 de l’indice de stabilité de la solution. Cette carte est affichée en quatrième position. Dans la dernière image, les vrais positifs (VP) sont en **vert**, les faux positifs (FP) en **rouge**, les faux négatifs (FN) en **bleu** et enfin, les vrais négatifs (VN) en noir. 101

4.2 Les différents indices pouvant être calculés au cours de la méthode SGM₄, sont montrés pour le couple Cones de la base Middlebury qui est affiché sur la première ligne. La solution obtenue avec la méthode SGM₄ et la vérité de disparité sur les pixels non-occultés sont affichées sur la deuxième ligne, puis, l’indice de stabilité proposé et l’indice de perturbation sont sur la troisième ligne et, enfin, l’entropie et l’indice proposé par [Drory *et al.*, 2014] sur la quatrième ligne. Les valeurs d’indice sont ré-étalées pour une meilleure visualisation. 104

4.3 Les différents indices proposés à l’aide de la méthode α -Expansion, sont montrés pour le couple Cones de la base Middlebury. La solution et l’indice de stabilité sont montrés sur la première ligne. L’indice de perturbation et l’entropie sont montrés sur la deuxième ligne. Les valeurs d’indice sont ré-étalées pour une meilleure visualisation. 111

4.4	Courbes représentant la capacité des indices à prédire les erreurs. Les reconstructions sont obtenues par la méthode SGM ₄ . Nous montrons, sur la première ligne (a), (b), les courbes pour le couple Cones, sur la deuxième ligne (c), (d), les courbes pour le couple Teddy et sur la troisième ligne (e), (f), les courbes correspondantes aux 200 couples de la base KITTI15.	113
4.5	Courbes représentant la capacité des indices à prédire les zones occultées. Nous montrons, sur la première ligne (a), (b), les courbes pour le couple Cones, sur la deuxième ligne (c), (d), les courbes pour le couple Teddy et sur la troisième ligne (e), (f), les courbes correspondantes aux 200 couples de la base KITTI15. Les reconstructions sont obtenues par la méthode SGM ₄	114
4.6	Distribution de l'erreur divisé par un indice. Ce rapport est normalisé par son l'écart-type empirique.	115
4.7	Courbes représentant la capacité des indices à prédire les erreurs dans le couple Cones (a) et (b), et le couple Teddy (c) et (d) ainsi que les occultations dans le couple Cones (e) et (f) et dans le couple Teddy (g) et (h). Les reconstructions sont obtenues par la méthode α -expansion.	116
4.8	De haut en bas, le couple 56 de la base KITTI15, carte de profondeur, l'indice de stabilité, la carte après filtrage utilisant l'indice, carte obtenue après repondération par l'indice.	117
4.9	De haut en bas, le couple 20 de la base KITTI15, carte de profondeur, l'indice de stabilité, la carte après filtrage utilisant l'indice, carte obtenue après repondération par l'indice	118

Liste des tableaux

1.1	Table de correspondance entre un arc dans le graphe et un terme de la posiforme.	11
1.2	Affectations réalisables et valeurs de label associées $l \in [0, 4]$, ($n = 5$)	17
1.3	L'expression $(\mathbf{x} \wedge \bar{\mathbf{y}}, \bar{\mathbf{y}} \wedge \mathbf{x})$ remplace l'affectation $(x, y) = (1, 1)$ par $(x, y) = (0, 0)$.	26
1.4	Table de vérité de l'opérateur \sqcup . Les couples d'affectation $(1, 1)$ mis en jeu sont remplacés par $(0, 0)$	26
1.5	Table de correspondance entre les termes de la posiforme et les arcs dans le graphe QPBO.	27
1.6	Proposition de label pour les différentes fusions	33
2.1	Comparaison entre α -Expansion et Ishikawa de l'énergie finale et du temps d'exécution pour un bruit gaussien et un bruit laplacien. Dans cette expérimentation, on a $\lambda = 0.8$ et les énergies sont données à un facteur 10. On remarque que la solution obtenue par α -Expansion est une solution minimale du problème.	44
2.2	Comparaison entre α -Expansion et Ishikawa de l'énergie finale et du temps d'exécution pour un bruit de Cauchy. Dans cette expérimentation, on a $\lambda = 0.04$ et les énergies sont données à un facteur 100. La solution α -Expansion n'est pas minimale en étant toutefois de valeurs très proches de celles du minimum de l'énergie. En pratique, les solutions sont elles-aussi très proches.	44
2.3	Comparaison des énergies et temps de calcul dans le cas où r est la fonction valeur absolue lissée pour différentes méthodes d'optimisation. Les valeurs d'énergie des schémas sont données en écart à l'énergie minimale. Les autres méthodes sont α -expansion optimisé par fusion relaxée, par QPBO, par QPBO-P, reformulé en fusion convexe et enfin le schéma Saut composé de deux fusions : le saut +1 et le saut -1.	45
2.4	Comparaison entre les schémas d'optimisation α -Expansion et Alterné-C pour trois méthodes d'optimisation. Les valeurs d'énergie sont données en écart à l'énergie QPBO-P pour le schéma Alterné-C, les temps d'exécution sont donnés en pourcentage du temps d'exécution de QPBO-P pour le schéma Alterné-C.	46
2.5	Comparaison des méthodes d'optimisation sur quatre couples d'images test de la base Middlebury dans le cas d'une régularisation avec la fonction valeur absolue. Les énergies finales des différents schémas d'optimisation sont données en écart à l'énergie minimale.	62
2.6	Comparaison des méthodes d'optimisation sur quatre couples de la base KITTI15 dans le cas d'une régularisation par la valeur absolue. Les énergies finales des différents schémas d'optimisation sont données en écart à l'énergie minimale.	63

2.7	Comparaison des énergies, temps de convergence et taux d'erreur pour 4 images, 5 schémas d'optimisation et 3 méthodes d'optimisation. Les valeurs d'énergie sont données en écart à l'énergie Alt-L par QPBO-P. Le taux d'erreur est le pourcentage de pixels non-occultés à plus d'une disparité par rapport la disparité vraie (err.M).	66
2.8	Comparaison des énergies, temps de convergence et taux d'erreur pour 4 images, 5 schémas d'optimisation et 3 méthodes d'optimisation. Les valeurs d'énergie sont données en écart à l'énergie Alt-L par QPBO-P. L'erreur est le pourcentage de pixels à plus de trois disparités de la carte de disparité "non-occ" (err.K).	67
3.1	Minimisation d'une énergie définie par le terme Data soumis à la régularisation par la valeur absolue. Les valeurs $E(\mathbf{l} l_p = k)$ sont obtenues par la somme des trois autres. Toutes les solutions qui sélectionnent des valeurs $E(\mathbf{l} l_p = k) = 8$, en rouge , sont des solutions minimales.	72
3.2	Taux d'erreur KITTI calculés sur toute la base KITTI15. Les raffinements sont l'optimisation par ligne puis les colonnes de la solution réalisé 1 et 2 fois (1 Opt. HV et 2 Opt. HV), et un filtrage médian de la solution avec deux tailles de fenêtre. Les raffinements sont réalisées sur les solutions des méthodes SGM ₄ , SGM ₈ et SGM ₁₆ . Opt.HV permet d'améliorer le taux d'erreur, mieux qu'un filtrage médian 5x5, mais moins bien qu'un filtrage médian 11x11	85
3.3	Résultats chiffrés pour la paire 199 de la base KITTI15. Les solutions sont présentées dans la Figure 3.23.	93
3.4	Résultats chiffrés pour la paire 199 de la base KITTI15. Les paramètres de régularisation sont choisis pour optimiser le critère err.K, respectivement avec Alterné-C et SGM4.	93
4.1	Matrice de confusion	100
4.2	Pourcentages moyens d'erreur à plus de trois pixels de disparité de la vérité terrain pour les bases KITTI12 et KITTI15 : sur la première ligne, les résultats lorsque l'indice n'est pas pris en dans la reconstruction, sur la deuxième ligne, lorsque la solution est filtrée avec l'indice et sur la troisième ligne, après repondération par l'indice. Chaque résultat est donnée avant et après d'appliquer un filtrage LRC sur la solution.	119

Liste des symboles

G	graphe
G_P	graphe associé à la posiforme P
\mathcal{V}	nœud du graphe
\mathcal{V}	ensemble des nœuds d'un graphe
\mathcal{S}	nœud source du graphe-ST
\mathcal{T}	nœud puits du graphe-ST
\mathcal{S}	ensemble des nœuds affectés à la source
\mathcal{T}	ensemble des nœuds affectés au puits
$\text{Coupe}(\mathcal{S}, \mathcal{T})$	valeur de la coupe associée à la partition des nœuds entre la source et le puits.
\mathcal{X}_u	nœud de G_P correspondant à la variable x_u de la posiforme P
\mathcal{X}	ensemble des nœuds du graphe G_P
\mathcal{X}^*	ensemble des nœuds du graphe autre que la source ou le puits
\mathcal{A}	ensemble des arcs d'un graphe
\mathcal{A}^*	ensemble des arcs d'un graphe ne faisant intervenir ni la la source, ni le puits
\mathcal{E}	ensemble des arêtes d'un graphe
c_{uv}	capacité entre les nœuds \mathcal{U} et \mathcal{V} , ou \mathcal{X}_u et \mathcal{X}_u
f_{uv}	flot entre les nœuds \mathcal{U} et \mathcal{V} , ou \mathcal{X}_u et \mathcal{X}_u
\mathcal{S}^*	ensemble des nœuds atteignables depuis la source
\mathcal{T}^*	ensemble des nœuds atteignables depuis le puits
◆	ensemble des nœuds atteignables ni depuis la source, ni depuis le puits
◇	affectation à la source ou au puits pour les nœuds de ◆
x	variable booléenne
\mathbf{x}	vecteur de variables booléennes
\mathbf{x}^{opt}	vecteur d'une affectation minimale de l'énergie
x_i^a	i-ème variable booléenne sur vecteur \mathbf{x}^a
\bar{x}	conjugué de la variable booléenne
$\bar{\mathbf{x}}$	vecteur de conjugués de variable booléenne
F	fonction pseudo-booléenne sous sa forme unique
P	posiforme
P_G	posiforme associée au graphe G

Res_G	posiforme résiduelle associée au graphe G
Res_S	posiforme résiduelle restreinte aux variable de \mathcal{S}^*
$Res_{\mathcal{T}}$	posiforme résiduelle restreinte aux variable de \mathcal{T}^*
Q	posiforme relaxée
p	pixel
I	ensemble ordonné de pixels
I^1 et I^2	images d'un couple stéréoscopique
N	ensemble des couples de pixels voisins
N'	ensemble des couples de pixels voisins pris dans un sens arbitraire
d_p	disparité associée au pixel p
l_p	label, i.e variable discrète, associé au pixel p
l	vecteur de labels
l^0	vecteur de labels courants
l^{opt}	vecteur de labels correspondant à une affectation minimale vis à vis de l'énergie
l^b	vecteur de labels bruités
\dot{l}	vecteur des valeurs vraies des labels
\hat{l}	vecteur des valeurs estimée des labels
l_i^a	i-ème label sur vecteur x^a
E	énergie multi-labels
E^C	énergie multi-label restreinte aux pixels de C
D_p	fonction de coût associée au pixel p
D'_p	fonction de coût associée au pixel p après repondération
D_p^1	fonction de coût associée au pixel p de l'image I^1
Reg_{pq}	fonction de coût associée au couple de pixels (p, q)
λ	paramètre multiplicatif sur les termes de régularisation
$(P1, P2)$	paramètres de la fonction de Pott lissée
\mathbb{E}	espérance
σ	écart-type
Σ	matrice de covariance
Π	matrice de précision
\mathbb{P}_p	probabilité du label l_p
$\mathbb{P}_{p,p+1}$	probabilité jointe des labels l_p et l_{p+1}
$\mathbb{P}_{a \rightarrow b}$	probabilité de la chaîne entre a et b , a exclu, sachant la valeur du label l_a .
$\Phi_{a \rightarrow b}$	probabilité minimale de la chaîne entre a et b , a exclu, sachant la valeur du label l_a .
$m_{p \rightarrow q}$	message associé à l'arc $(\mathcal{V}_p, \mathcal{V}_q)$
SGM	coût SGM

SGM^*	coût SGM dans sa version originale
ϱ	ensemble des directions prises en compte
Acc_p^{ϱ}	valeurs accumulées en le pixel p pour la direction r
L_p^{ϱ}	valeurs accumulées en le pixel p pour la direction r pour SGM^*
χ_p^{ϱ}	ensemble des pixels dans les ϱ directions autour du pixel p
$Stab_p^{GC}$	indice de stabilité du pixel p pour une méthode d'optimisation donnée, ici la coupure de graphe

Introduction

Les appareils photographiques sont devenus omniprésents. Les seules ventes de smartphones en 2013 représentent plus d'un milliard d'unités. Ce domaine, très porteur, a vu de nombreuses évolutions techniques que ce soit au niveau des capteurs : une diminution de la taille du capteur, une augmentation des résolutions et une baisse des coûts, qu'au niveau des traitements appliqués aux images. Cette évolution technique a permis la mise œuvre de nouvelles applications. Dans le domaine automobile, les véhicules sont de plus en plus souvent équipés de systèmes d'aide à la conduite (*Advanced driver-assistance systems*, ADAS). Nous utiliserons cet exemple pour illustrer certains enjeux actuels en traitement d'image.

Les ADAS concernent des tâches de conduites plus ou moins complexes, et vont de système anti-blocage des roues (ABS) jusqu'à l'automatisation complète du véhicule. Les gains potentiels de tels systèmes sont importants en terme de sécurité routière, de consommation énergétique et d'efficacité des transports. Toutefois, la mise en œuvre d'une conduite automatisée soulève de nombreuses questions qui mettent en jeu le rapport de l'Homme à la Machine. En particulier la fiabilité du système envisagé est souvent questionnée, par exemple : doit-on donner la possibilité au conducteur d'intervenir ? Que faire si la décision prise par le système n'est pas compatible avec celle du conducteur ? Ainsi, il y a un fort enjeu à évaluer la fiabilité des performances de ces systèmes. Cette capacité d'évaluation est d'autant plus importante que, comme toute technologie innovante, elle demande de se placer aux limites de la technique, avec des méthodes dont la connaissance et la maîtrise sont parfois partielles.

Les ADAS peuvent intégrer des caméras optiques afin d'acquérir une connaissance de l'environnement autour du véhicule. Les informations acquises consistent en un volume d'information chiffrée qui correspond à des valeurs d'énergie reçues par le capteur. Obtenir une compréhension de la scène routière à partir des images collectées est une tâche difficile. Cette tâche est souvent décomposée en sous-problèmes, qui prennent en entrée les images et/ou les résultats d'un autre sous-problème. Ainsi les données passent par une chaîne de traitement, parfois complexe, où les informations extraites des données sont propagées afin d'obtenir un résultat en sortie de chaîne. Certains de ces sous-problèmes sont posés comme des problèmes de traitement d'image : étalonnage des caméras, orientation relative des caméras, reconstruction stéréoscopique, classification d'image, segmentation... Ces problèmes font l'objet d'un intérêt scientifique particulier. Dans le contexte routier, nous pouvons citer le véhicule instrumenté Stéréopolis de l'IGN [Paparoditis *et al.*, 2012] équipé de caméras optiques, de LiDAR et de centrales inertielles.

Caractériser la solution d'une chaîne de traitements est une tâche difficile. Il est parfois possible de caractériser l'incertitude sur les données images en entrée de la chaîne, grâce à une modélisation du processus physique de formation de l'image. Cependant, la caractérisation de l'incertitude des sorties d'une des briques de la chaîne n'est pas toujours bien maîtrisée. Ces sorties étant utilisées en entrée d'autres briques de traitement, il est difficile de faire le lien entre l'incertitude sur les données et l'incertitude sur la solution en sortie de chaîne. De plus, l'incertitude des données n'est pas pas la seule source d'incertitude : chaque traitement peut occasionner des erreurs ou des pertes. Ainsi, il y a un enjeu important à utiliser des méthodes capables de prendre en compte une incertitude sur les entrées et à estimer une incertitude sur la sortie. Cette approche va dans le sens d'une meilleure compréhension des méthodes utilisées et d'une meilleure qualification des résultats.

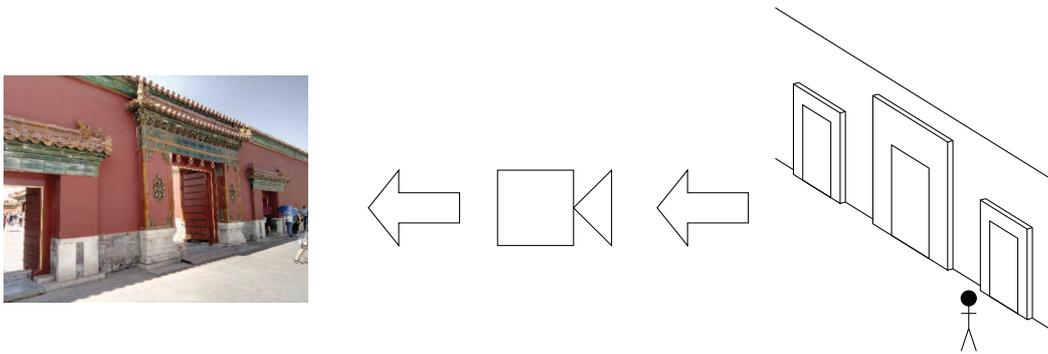


Figure 1 – Scène acquise par une caméra. A droite, la scène en 3D, au centre la caméra, à gauche l'image résultante. Estimer l'image en sortie de la caméra, à partir de la scène, est le problème direct. Estimer la scène acquise à partir de l'image est le problème inverse.

1 Reconstruction stéréoscopique

La modélisation de l'environnement autour du véhicule demande de décrire les objets de la scène routière en 3D. La reconstruction stéréoscopique consiste à reconstituer une scène en trois dimensions à partir de deux images de cette scène. Si la position des caméras et la géométrie d'acquisition est connue, il est possible de calculer la profondeur d'un objet de la scène 3D à partir des positions de ces objet dans les deux images. Le principe de ce calcul est celui de la triangulation en 3D des rayons issus des deux projections de l'objet. Historiquement la première application à un couple d'images est attribuée à Laussedat en 1849 [Sturm, 2011]. Toutefois, le calcul des profondeurs repose alors largement sur les capacités de l'humain à interpréter la scène : il s'agit le plus souvent de demander à un opérateur de pointer un même détail dans les images. Avec le développement de l'informatique, ce problème a été formalisé, afin de pouvoir le traiter numériquement et ainsi automatiser la reconstruction stéréoscopique.

Il existe d'autres approches qui consistent à déterminer la profondeur à partir d'images en utilisant le principe de la triangulation. Au lieu de considérer deux caméras, la méthode, appelée "Structure à partir du mouvement" (*Structure from Motion*, SfM), ne considère qu'une seule caméra en mouvement relatif par rapport à la scène [Faugeras et Lustman, 1988]. Aussi, les approches multi-images prennent en compte simultanément un ensemble de vues pour déterminer la profondeur dans la scène. Le volume des données en entrée augmente, ce qui augmente la difficulté du problème, de sorte que certaines simplifications faites en stéréoscopie ne sont plus possibles. Cependant ces méthodes peuvent exploiter la redondance des informations, ce qui amène généralement à des solutions plus robustes [Seitz *et al.*, 2006].

1.1 Problème direct et problème inverse

Le principe d'une caméra est de convertir une quantité d'énergie lumineuse entrante en une image. Le capteur est composé de capteurs élémentaires, appelés photosites, capables de convertir une quantité d'énergie reçue en un signal électrique qui est quantifié en information numérique. Chaque photosite correspond à un pixel et les pixels sont le plus souvent arrangés en une grille régulière rectangulaire. Ainsi, l'image numérique consiste en un tableau de pixels associés à une ou plusieurs valeurs. Le processus de formation de l'image est le problème direct. Les applications en traitement d'image consistent à déterminer les caractéristiques de la scène qui a été acquise à partir des images. Il s'agit donc de problèmes inverses, comme illustré dans la Figure 1.

La modélisation d'un problème inverse se fait généralement en deux étapes. La première étape consiste à étudier le modèle direct, puis la deuxième étape consiste à inverser ce modèle pour résoudre le problème inverse. On parle de problème bien posé lorsque la solution existe, qu'elle est unique et que la solution dépend de manière continue des données. Si le problème



Figure 2 – Image gauche du couple 11 de la base KITTI 2015. Nous pouvons voir que la scène est relativement plane et l’intervalle de profondeur entre les objets les plus lointains et les plus proches est important.

est bien posé, il y a une possibilité de trouver un algorithme stable qui donne la solution du problème. Cependant, la plupart des problèmes inverses sont mal-posés. Même dans l’éventualité où le problème est bien-posé, le problème est souvent mal-conditionné : une perturbation sur les données du problème aura tendance à être amplifiée dans la solution. La solution obtenue est alors instable et sensible au bruit sur les données. Par exemple, dans le cas d’un floutage par un noyau gaussien : le problème direct est bien posé et bien conditionné. Le problème inverse consiste en une dé-convolution par un noyau gaussien : le problème inverse est, en théorie en continu, bien posé, mais très mal-conditionné. Pour résoudre le problème inverse de manière assez stable, il est nécessaire d’introduire des connaissances supplémentaires qui sont ajoutées au modèle qui consistent en des *a priori* sur la solution. La solution est alors un compromis entre une solution en adéquation avec les données et une solution en adéquation avec les *a priori*.

1.2 Spécificités des scènes routières

Pour répondre au problème de l’évaluation des méthodes de reconstruction, des bases de données ont été mises en place. Nous pouvons citer par exemple la base d’évaluation Middlebury [Scharstein *et al.*, 2014a]. Toutefois, le problème de la reconstruction en contexte routier possède des caractéristiques qui ont motivé la mise en place de bases de données spécifiques, comme par exemple la base de données KITTI [Menze et Geiger, 2015].

Le problème de la reconstruction stéréoscopique fait intervenir une fonction de similarité entre parties d’image. Or les scènes routières possèdent des larges zones homogènes qui posent des problèmes d’ambiguïtés pour la reconstruction comme nous pouvons l’observer dans la Figure 2. Il s’agit par exemple de la chaussée, qui est globalement uniforme et grise. Les scènes acquises en extérieur sont sujettes à des variations des conditions d’éclairage. De plus l’aspect des objets peut varier entre les images. Les vitres et la peinture des véhicules sont très spéculaires, ainsi que, dans une moindre mesure, la chaussée.

La structure géométrique de la scène est en partie connue. En effet, la chaussée, qui occupe une part importante de l’image, est vue comme un plan incliné. De plus, les scènes urbaines sont généralement formées de nombreux plans comme les façades de bâtiments et les panneaux. La scène comporte aussi des objets fins, comme les supports des panneaux ou les câbles, qui sont connus pour être difficiles à reconstruire.

1.3 Formulation dans le cadre Bayésien

Les premières méthodes de reconstruction consistent en un appariement de zones de l’image qui se ressemblent localement. La forme d’un objet d’une image à l’autre est modifiée du fait du changement de point de vue. Les comparaisons sont réalisées sur des voisinages de petite

taille pour limiter cet effet. Or, si la taille des voisinages est trop petite, le problème devient trop ambigu, et peu de correspondances seront trouvées entre les images. La détermination des profondeurs ne peut que se faire de manière éparse. La profondeur est déterminée seulement sur les pixels qui ont bien pu être appariés. Une approche est de se restreindre à un ensemble de points d'intérêt très fiables avant de mettre en œuvre l'appariement. Ces points peuvent correspondre à des contours [Canny, 1986], à des coins [Harris et Stephens, 1988] ou à des zones de l'image qui sont robustes à certaines transformations [Lowe, 1999].

Une formulation alternative du problème considère la solution comme l'optimum d'une énergie qui fait le compromis entre la similarité des positions image appariées et la régularité de la solution. Ce type d'approche permet d'obtenir des solutions dites denses où une valeur de profondeur est attribuée à chaque pixel. La formulation bayésienne donne une manière de dériver cette énergie. La formule de Bayes permet en effet d'écrire la probabilité a posteriori comme la fonction d'un produit de probabilité. En appliquant le principe du maximum de vraisemblance, la solution du problème est alors obtenue par l'affectation qui maximise la probabilité a posteriori. Ce type de formulation a permis d'éclairer les hypothèses sous-jacentes au choix d'une énergie en lui donnant une interprétation probabiliste.

Dans ce cadre, l'a priori souvent utilisé est que des points voisins sont à des profondeurs proches. Avec les méthodes d'optimisation discrète, un tel a priori, pris en compte sur les pixels en 4-connexité deux à deux, amène à optimiser un champ de Markov 2D du premier ordre [Geman et Geman, 1984]. Ce problème est difficile numériquement. Cependant, au début des années 2000, une résolution approchée est proposée par coupure de graphe (*Graph Cut*, GC). Appliquées à la reconstruction stéréoscopique, ces techniques d'optimisation ont permis d'obtenir des résultats denses, avec un coût de calcul pas trop élevé, bien que l'appariement soit fait au niveau du pixel, sans prendre en compte de voisinage. Plus généralement, cette approche a permis une meilleure compréhension des hypothèses faites dans le modèle. Toutefois, en donnant des résultats denses, la question du traitement à appliquer aux zones occultées dans les images se pose.

Plus tard, dans le contexte de la reconstruction stéréoscopique, la méthode SGM [Hirschmüller, 2008] propose une alternative approchée à la résolution d'un champ de Markov 2D moins coûteuse, grâce à l'utilisation de la programmation dynamique : le champ 2D est décomposé en alignements de pixels autour d'une position donnée.

1.4 Approche par apprentissage

L'apprentissage consiste à déterminer un modèle à partir d'exemples. Pour que le modèle appris ne soit pas dépendant des exemples en entrée mais généralise bien ces entrées, il est important que les données en entrée de l'apprentissage soient en très grand nombre et variées. En reconstruction la première introduction de l'apprentissage du modèle s'est limitée à la fonction de similarité entre les images [Paar *et al.*, 1998].

Dans le début des années 2010, les méthodes d'apprentissage par réseau de neurones ont fait de grands progrès en performance, avec les réseaux de neurones dits profonds. Depuis lors, cette approche tend à être appliquée à tous les problèmes image. A l'heure actuelle, ce sont les mesures de similarité obtenues par apprentissage qui donnent les meilleures performances sur les différentes bases d'évaluation de la reconstruction stéréoscopique. Depuis, des méthodes visant à apprendre l'intégralité du processus de reconstruction entre les données et les résultats ("du début à la fin"), donnent des résultats intéressants [Kendall *et al.*, 2017]. Cette approche va dans le sens d'une réduction du fractionnement des problèmes en sous-problèmes et limite potentiellement les pertes d'information lors des différentes étapes du traitement. Toutefois, le risque de sur-apprentissage est important, du fait que les bases d'évaluation en reconstruction stéréoscopique sont assez petites et peu nombreuses. Néanmoins, nous devrions voir la taille des bases augmenter dans le futur, avec notamment le développement récent de bases d'images synthétiques [Mayer *et al.*, 2016].

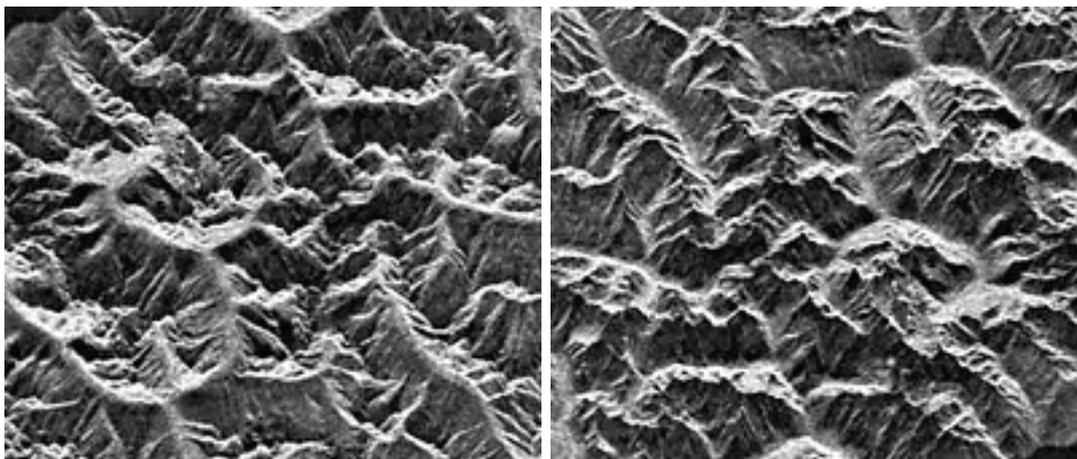


Figure 3 – La même image radar est présentée avec deux orientations différentes. L'interprétation des volumes est plus difficile sur la première image : nous avons tendance à interpréter la première image "en creux" et la seconde "en bosse". Cela vient du fait que dans les scènes naturelles, la lumière vient du haut et que cet *a priori* est pris en compte dans notre interprétation 3D de la scène.

La force de ces méthodes par apprentissage est qu'elles permettent la fusion de nombreux indices de profondeur, de manière analogue à la vision humaine, comme la déformation de la texture. Toutefois ces méthodes présentent plusieurs inconvénients. Tout d'abord, le temps d'apprentissage est assez long, et il est souvent nécessaire de refaire l'apprentissage quand on change de système. Ensuite, le modèle obtenu ne peut pas être facilement interprété et compris pour connaître ses limites. De manière générale, la mise en œuvre des méthodes par apprentissage ne va pas dans le sens d'une meilleure compréhension des phénomènes modélisés.

Deux images sont nécessaires pour former une intersection en trois dimensions. Nous avons ainsi besoin de nos deux yeux pour percevoir la profondeur. Toutefois, même en fermant un œil, nous sommes toujours capable d'interpréter le relief de scène et d'acquérir une connaissance sur la structure en trois dimensions de la scène. Il est connu que l'humain utilise un grand nombre d'indices liés à la profondeur et différents *a priori* : déformation de la texture, taille connue de certains objets, dégradés sur les surface courbes, forte probabilité d'éclairage par le haut, comme illustré dans la Figure 3. Ainsi, comme dans l'approche bayésienne, ce que nous voyons est un compromis entre le signal reçu par l'œil et nos *a priori*. Il est donc possible de créer des images où les *a priori* ne sont pas respectés, ce qui donne lieu à une interprétation erronée de la scène que l'on nomme illusion d'optique. Avant l'introduction de l'apprentissage par réseaux profonds en reconstruction, quelques méthodes ([Saxena *et al.*, 2006; Delage *et al.*, 2006]) ont été proposées par apprentissage pour déterminer une carte de profondeur à partir d'une seule image. Ce type d'approche tend à se développer avec l'arrivée de l'apprentissage profond [Cao *et al.*, 2017; Lee *et al.*, 2017].

2 Méthode d'optimisation

Les problèmes en vision et en image sont souvent formulés comme la recherche d'une affectation minimale vis-à-vis d'une énergie. Le plus souvent, cela conduit à des problèmes combinatoires dont le nombre de variables est de l'ordre du million. Prenons exemple d'un problème simple dont la solution est de taille 10×10 pixels et où chaque pixel peut prendre 10 valeurs différentes. Le nombre d'affectations possibles est de 10^{100} . Pour comparaison, le nombre d'atomes dans l'Univers est estimé à 5.5×10^{79} . Cette combinatoire rend les problèmes difficiles et des méthodes d'optimisation adaptées doivent être mises en œuvre.

La classe de complexité des problèmes que l'on rencontre en imagerie est soit P (polynomial), soit NP (non-polynomial), dans le cadre de la théorie de la complexité. Il est nécessaire, pour que les temps de résolution soient raisonnables, que l'algorithme s'exécute en un temps polynomial. Les algorithmes qui s'exécutent en un temps polynomial que nous connaissons résolvent des problèmes P. Il n'y a pas, pour l'heure, d'algorithme en temps polynomial pour des problèmes NP et l'existence d'un tel algorithme ou son inexistence n'a pas été résolue. C'est la célèbre question "P=NP" ou "P≠NP". Ainsi, lorsque le problème est NP, comme c'est le cas pour la minimisation d'une énergie dérivée d'un champ de Markov du premier ordre, nous avons deux alternatives : résoudre un problème approché qui est P, ou se placer dans un cas particulier qui est P.

On distingue l'optimisation continue, où les variables à optimiser prennent des valeurs sur un espace continu, de l'optimisation discrète, où les variables à optimiser prennent des valeurs discrètes. Ces deux familles de méthodes ont chacune leurs particularités et certains problèmes sont plus faciles à traiter avec l'une ou l'autre de ces formulations. Par exemple, le problème déjà évoqué de la dé-convolution dans une image est plus compliqué à optimiser dans une formulation discrète que dans une formulation continue. La formulation discrète demande ainsi de prendre en compte des interactions d'ordre élevé dont l'optimisation est difficile. Par contre, pour le problème de la reconstruction stéréoscopique, la formulation discrète semble la plus naturelle.

2.1 Optimisation par coupure de graphe

L'optimisation par coupure de graphe trouve son origine dans les problèmes de transport. Dans [Tolstoi, 1930], l'auteur cherche à déterminer la capacité maximale qui peut être transportée sur le réseau de chemin de fer soviétique (ciment, ...). Ces travaux sont repris dans [Harris et Ross, 1955], dans le but d'acquérir des connaissances stratégiques sur les républiques soviétiques (rapport dé-classifié en 1999, [Schrijver, 2002]). Les auteurs formulent alors un certain nombre de problèmes dont, le problème suivant :

"Considérez un réseau qui relie deux villes en passant par un nombre de villes intermédiaires, où chaque lien du réseau se voit attribuer un nombre qui représente sa capacité. En supposant un état d'équilibre, trouver le flot maximal d'une ville donnée à une autre".

Dans cette formulation, on trouve, de manière succincte, la définition d'un graphe composé de nœuds et de liens pondérés par des capacités et celle du flot maximal. Dans [Ford et Fulkerson, 1962], un algorithme est proposé pour résoudre le problème du flot maximal. À peu près à la même époque, le lien est fait entre le problème du flot maximal et celui de la coupe minimale ([Elias *et al.*, 1956]). Si l'on considère un réseau composé d'une unique voie entre deux points dont la capacité varie, le flot maximal entre ces deux points est égal à la section minimale le long de la voie. La dualité du problème du flot maximal et de la coupe minimale est une généralisation de ce constat à l'ensemble d'un réseau. La coupe minimale peut être formulée comme, "quels liens doivent être retirés dans le réseau tels que :

- il ne soit plus possible de circuler de la ville A à la ville B,
- la somme des capacités des liens retirés soit minimale."

D'un point de vue stratégique, la valeur du flot maximal donne, par exemple, la quantité de ravitaillement qui peut être envoyée et la coupe donne les voies du réseau qui, si elles étaient neutralisées, impactent avec certitude cette quantité maximale. La principale condition pour pouvoir utiliser une méthode par coupure de graphe est la sous-modularité de l'énergie que l'on cherche à optimiser. Cela se traduit par des fortes limitations sur la forme des énergies pouvant être utilisées.

Au début des années 2000, des avancées importantes sont proposées dans les méthodes par coupure de graphe :

- L'algorithme Maxflow ([Boykov et Kolmogorov, 2004]) permet dans la pratique une accélération de l'optimisation par rapport aux algorithmes antérieurs.

- Le cas de la régularisation L1 est résolu exactement par la méthode proposée dans Ishikawa [2003].
- La méthode d’optimisation α -Expansion ([Veksler, 1999]) permet de résoudre, de manière approchée, certains problèmes multi-labels non-convexes.
- Dans [Boros et Hammer, 2002], la résolution par coupure de graphe est étendue, de manière approchée, à la résolution d’un problème binaire quadratique.

Ces avancées ont provoqué un regain d’intérêt pratique et théorique pour les méthodes par coupure de graphe.

2.2 Optimisation par programmation dynamique

La programmation dynamique est une méthode d’optimisation proposée par Richard Bellman [Bellman, 1952]. Elle repose sur le principe dit de Bellman qui peut s’énoncer simplement comme : si un chemin optimal entre A et B passe par C , alors les sous-chemins entre A et C et C et B sont des sous-chemins optimaux. Ainsi, un chemin optimal est composé de sous-chemins optimaux. Ce principe permet de limiter, dans un problème combinatoire, l’exploration des affectations possibles, en excluant celles qui comportent des sous-chemins non-optimaux. Les premiers problèmes résolus par cette méthode sont des problèmes de commande optimale, comme par exemple, le calcul de la trajectoire de lancement d’un satellite qui minimise la consommation de carburant [Bellman et Dreyfus, 1958]. Une des limites de cette méthode est que le champ que l’on cherche à optimiser ne doit pas comporter de cycles. Ainsi, pour l’appliquer en pratique, il faut soit résoudre un problème alternatif qui se représente comme une chaîne acyclique, soit adopter des méthodes de résolution approchées.

3 Indices de confiance dans la solution

3.1 Incertitude des données

La connaissance des incertitudes sur les données est nécessaire pour prendre en compte chaque donnée de façon adéquate, et pour ainsi obtenir de meilleurs résultats que lorsque l’on suppose une qualité identique de chaque donnée. Pour connaître avec précision l’incertitude sur les données, il est nécessaire de lister et de modéliser chaque source de perturbation dans les images. Certaines de ces perturbations possèdent des modèles physiques connus, comme par exemple, le bruit photonique des capteurs. Toutefois ce type d’approche demande des connaissances détaillées des processus d’acquisition et de production des données et sont difficiles à mettre en œuvre lorsque les conditions d’acquisition et de production ne sont pas bien maîtrisées, comme dans le cas d’acquisitions mobiles.

On peut regretter que dans la plupart des cas, l’incertitude sur les données est peu ou pas prise en compte en imagerie. Pour une image, un modèle simple est de considérer que chaque pixel a la même distribution indépendante avec un écart type différent pour chaque pixel. Cet écart-type peut alors être utilisé pour pondérer les attaches aux données dans le modèle. En reconstruction stéréoscopique, une approche est de détecter dans les images des caractéristiques connues pour induire des ambiguïtés dans la reconstruction. Il s’agit par exemple de détecter les zones homogènes de l’image, des textures répétitives. Une grande variété d’indices a été proposée, ce qui a motivé des propositions pour obtenir de tels indices par apprentissage [Seki et Pollefeys, 2016]. Si, avec cette approche, on vise plus à prédire les zones qui vont être sources d’erreurs dans la reconstruction, cet indice peut aussi être utilisé comme une incertitude lors de la reconstruction afin d’améliorer le résultat final.

3.2 Caractériser la solution

Avec certaines méthodes de résolution, comme les moindres carrés, il est possible d’avoir la relation explicite entre l’incertitude sur les données et l’incertitude sur la solution. Cependant, dans le cas général, ce lien n’est pas explicite et le plus souvent il est difficile à exprimer. Dans le contexte de la reconstruction stéréoscopique, l’erreur due à la triangulation peut se calculer et permet d’obtenir une incertitude en fonction de la profondeur. Toutefois une telle caractérisation est insuffisante car elle est assez partielle. Pour avoir une information de confiance plus complète, une approche consiste à calculer une solution à l’aide d’un modèle simple, puis à apprendre un indice à partir de cette solution et des données comme dans [Haeusler *et al.*, 2013; Spyropoulos *et al.*, 2014; Park et Yoon, 2015] et plus récemment, en apprentissage profond, dans [Poggi et Mattocchia, 2017; Kim *et al.*, 2017]. De tels indices sont alors utilisés, soit pour raffiner le modèle initial, soit pour pondérer un modèle plus compliqué.

En partant du constat que l’optimisation des énergies représente un volume de calcul important et de nombreuses valeurs intermédiaires, il peut être intéressant d’utiliser ces informations pour acquérir une connaissance sur la solution. Un tel indice dérive du processus d’optimisation et lui est donc propre.

4 Plan de la thèse

Dans un premier chapitre 1, les méthodes par coupure de graphe sont présentées. Cette présentation visera à montrer comment un problème binaire peut être formulé comme un problème de coupure de graphe et comment cette méthode peut être étendue à la résolution de problèmes multi-labels. Nous porterons une attention particulière aux méthodes de résolution approchée par fusion binaire et nous présenterons quelques variantes que nous avons proposées et mises en œuvre. Nous verrons dans le Chapitre 2 comment formuler, par coupure de graphe, deux problèmes image particuliers : le dé-bruitage et la reconstruction stéréoscopique. Cette présentation sera l’occasion de décrire les modèles fréquemment utilisés pour la reconstruction stéréoscopique et d’étudier les caractéristiques particulières de ces deux classes de problèmes. Nous y décrirons aussi les choix de modèles qui nous ont semblé les plus pertinents.

Ensuite dans le Chapitre 3, nous nous focaliserons sur une autre méthode d’optimisation discrète utilisant la programmation dynamique. Nous détaillerons en particuliers différentes variantes de la reconstruction stéréoscopique fondées sur la programmation dynamique : les méthodes SGM, MGM et CAT. Nous comparerons les résultats de ces méthodes avec des résultats obtenus par coupure de graphe. Cette étude nous a conduit à proposer une interprétation originale de la méthode SGM.

Enfin dans le dernier chapitre 4, nous proposerons des indices de stabilité générés à partir des valeurs calculées - ou pouvant être calculées - au cours du processus d’optimisation. Nous ferons cela aussi bien dans le contexte de la programmation dynamique que dans le contexte de la coupure de graphe. Ces indices sont évalués et nous montrerons les améliorations sur les résultats pouvant être obtenues par la prise en compte de ces indices lors d’une étape de raffinement.

Chapitre 1

Optimisation par coupe de graphe

Les problèmes images sont souvent *NP*-difficiles. Cependant dans certains cas, il existe des algorithmes d'optimisation qui résolvent exactement le problème en temps polynomial. Dans ce chapitre, nous allons nous intéresser à l'une de ces méthodes : la formulation par coupure de graphe. C'est une méthode d'optimisation discrète qui met en jeu une catégorie de graphe, appelé graphe-ST, permettant de représenter une fonction de variables binaires d'ordre 2 et sous-modulaire (Section 1). Ces fonctions peuvent être minimisées en temps polynomial par différents algorithmes (Section 2). Des méthodes ont été proposées pour étendre la formulation par coupure de graphe à d'autres types de fonction (Section 3). Enfin nous reviendrons plus en détail sur l'une de ces méthodes : l'approximation de la solution multi-labels par un schéma d'optimisation binaire (Section 4).

1 Représentation d'une fonction par un graphe

La coupe de graphe est réalisée sur une catégorie de graphe appelée graphe-ST ou réseau de flots. Il permet de représenter des énergies de variables binaires. Cette énergie peut s'écrire sous la forme d'une fonction pseudo-booléenne (FPB). Cependant seule une classe de FPB peut être représentée par un graphe.

1.1 Coupure de graphe

Un graphe est une structure composée de nœuds et d'arcs reliant un couple de nœuds, $G = (\mathcal{V}, \mathcal{A})$ où \mathcal{V} est un ensemble de nœuds, parfois appelés sommets (*vertex*) et \mathcal{A} un ensemble d'arcs. Un arc relie une paire orientée de nœuds dans \mathcal{V}^2 et on note $(\mathcal{V}_u, \mathcal{V}_v)$ l'arc qui part du nœud \mathcal{V}_u vers le nœud \mathcal{V}_v . Les arcs $(\mathcal{V}_u, \mathcal{V}_v)$ et $(\mathcal{V}_v, \mathcal{V}_u)$ sont appelés arcs frères, ces deux arcs forment une arête notée $\{\mathcal{V}_u, \mathcal{V}_v\}$. Les arcs et les nœuds peuvent être pondérés, c'est-à-dire associés à une valeur. Nous considérerons des arcs pondérés par des coefficients *positifs* appelés capacités, on notera c_{uv} la capacité de l'arc $(\mathcal{V}_u, \mathcal{V}_v)$. Il existe des graphes non-orientés. Ils peuvent être vus comme des cas particuliers de graphes orientés où les capacités des arcs frères sont égales. Ils sont définis comme un ensemble de nœuds et un ensemble d'arêtes $G = (\mathcal{V}, \mathcal{E})$, $\mathcal{E} \subset \mathcal{V}^2$. Un exemple de graphe orienté est donné dans la Figure 1.1.

Le graphe utilisé pour la coupe de graphe est appelé graphe-ST. Il s'agit d'un graphe orienté sans boucle. Une boucle est un arc qui a le même nœud de départ et d'arrivée. Ainsi dans le cas sans boucle, un lien part d'un nœud et arrive à un autre nœud. Ce graphe possède deux nœuds particuliers - la source \mathcal{S} et le puits \mathcal{T} . Un exemple de graphe est représenté Figure 1.2. Le graphe-ST est aussi appelé réseau de flots : l'ensemble de la structure représente un réseau dans lequel peut s'écouler un flot de proche en proche de la source vers le puits dans la limite des capacités des arcs. Ce flot s'écoule le long d'un chemin. Un chemin $P = (\mathcal{V}', \mathcal{A}')$ dans le graphe G est une suite consécutive d'arcs du graphe orienté, formellement $\mathcal{V}' \subset \mathcal{V}, \mathcal{A}' = \{(\mathcal{V}'_0, \mathcal{V}'_1), \dots, (\mathcal{V}'_{n-1}, \mathcal{V}'_n)\} \subset \mathcal{A}$.

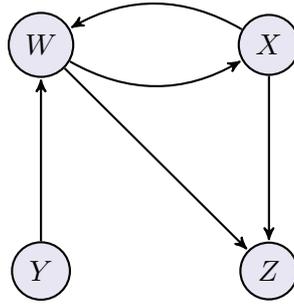


Figure 1.1 – Exemple de graphe orienté non pondéré, $G = (\{W, X, Y, Z\}, \{(W, X), (W, Z), (X, W), (X, Z), (Y, W)\})$.

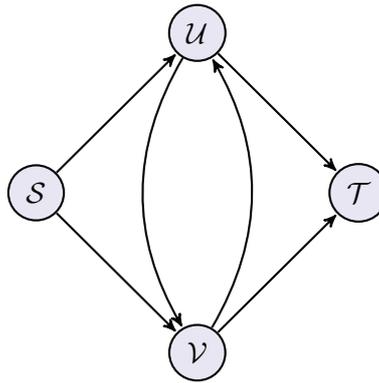


Figure 1.2 – Structure élémentaire de graphe-ST (source, puits et deux nœuds)

Une coupe de graphe est une partition des nœuds du graphe en deux parties disjointes, l'une contenant la source \mathcal{S} l'autre le puits \mathcal{T} . L'ensemble des nœuds de la partie "source", et de la partie "puits" sont notées respectivement \mathcal{S} et \mathcal{T} . Une coupe est associée à une valeur qui est la somme des capacités des arcs dont l'origine est dans \mathcal{S} et l'arrivée dans \mathcal{T} ,

$$\text{Coupe}(\mathcal{S}, \mathcal{T}) = \sum_{\nu_u \in \mathcal{S}, \nu_v \in \mathcal{T}, (\nu_u, \nu_v) \in \mathcal{A}} c_{uv} \quad (1.1)$$

1.2 Fonctions pseudo-booléennes

Les fonctions pseudo-booléennes (FPB) sont des polynômes de variables booléennes à valeur 0 ou 1,

$$F(\mathbf{x}) = a_\emptyset + \sum_i a_i x_i + \sum_{i < j} a_{ij} x_i x_j + \sum_{i < j < k} a_{ijk} x_i x_j x_k + \dots \quad (1.2)$$

L'écriture (1.2) est appelée forme unique. Le degré du polynôme correspond à l'ordre de la FPB. Il peut être intéressant de réécrire la fonction à l'aide d'une constante et d'une somme de termes à coefficient positif. Une telle écriture est appelée posiforme, notée P . Pour ce faire certaines variables sont remplacées par leur conjuguée

$$\bar{x} = 1 - x. \quad (1.3)$$

Une fonction pseudo-booléenne possède plusieurs posiformes. Il existe un lien entre une coupe de graphe et les FPB, avec $G_P(\mathcal{X}, \mathcal{A})$ le graphe correspondant à la posiforme P . Chaque nœud du graphe est représenté par une variable booléenne, qui correspond à l'affectation du nœud \mathcal{X}_p à la source ou au puits, nous choisissons la convention :

$$\begin{aligned} x_p = 0, \mathcal{X}_p \in \mathcal{S} \\ x_p = 1, \mathcal{X}_p \in \mathcal{T}. \end{aligned} \quad (1.4)$$

La capacité d'un arc entre en jeu dans la coupe si son origine est du coté de la source ($x_{origine} = 0$) et son arrivée du coté du puits ($x_{arrivée} = 1$) et uniquement dans cette configuration. Cela se traduit dans la posiforme par un terme quadratique $\bar{x}_{origine}x_{arrivée}$ multiplié par la capacité de l'arc considéré. Ainsi ces variables permettent d'écrire les valeurs de coupe du graphe comme des fonctions pseudo-booléennes.

$$Coupe(\mathcal{S}, \mathcal{T}) = \sum_{(\mathcal{V}_u, \mathcal{V}_v) \in \mathcal{A}} c_{uv} \bar{x}_u x_v = P_G(\mathbf{x}), \quad (1.5)$$

où \mathbf{x} est l'ensemble des variables x associées aux nœuds. La posiforme P_G est donc associée au graphe G ; c'est une FPB quadratique. Par définition la source est du coté de la source ($x_s = 0$) et le puits est du coté du puits ($x_t = 1$), ainsi les termes les mettant en jeu se réécrivent comme des termes unaires et le terme entre la source et le puits devient une constante. On définit l'ensemble des nœuds distincts de la source et du puits, $\mathcal{X}^* = \mathcal{X} / \{\mathcal{S}, \mathcal{T}\}$ et les arcs mettent en jeu uniquement ces nœuds $\mathcal{A}^* = \mathcal{A} \cap \mathcal{X}^{*2}$, la posiforme s'écrit alors :

$$P_G(\mathbf{x}) = c_\emptyset + \sum_{\mathcal{X}_u \in \mathcal{X}^*} c_{su} x_u + c_{ut} \bar{x}_u + \sum_{(\mathcal{V}_u, \mathcal{V}_v) \in \mathcal{A}^*} c_{uv} \bar{x}_u x_v \quad (1.6)$$

et la construction du graphe à partir de cette posiforme est donnée dans le Tableau 1.1.

Arc dans le graphe	Capacité	Terme dans la posiforme
$\mathcal{X}_u \rightarrow \mathcal{X}_v$	c_{uv}	$\bar{x}_u x_v c_{uv}$
$\mathcal{S} \rightarrow \mathcal{X}_u$	c_{su}	$x_u c_{su}$
$\mathcal{X}_u \rightarrow \mathcal{T}$	c_{st}	$\bar{x}_u c_{st}$

Tableau 1.1 – Table de correspondance entre un arc dans le graphe et un terme de la posiforme.

Le graphe donné dans la Figure 1.4(a), page 16, s'écrit comme une posiforme et sous la forme unique suivante :

$$P(x, y, z) = 2x + \bar{y} + \bar{z} + 2\bar{x}y + 2\bar{y}z \quad (1.7a)$$

$$= 2 + 2x + y + z - 2xy - 2yz \quad (1.7b)$$

1.3 La sous-modularité et représentation en graphe

La notion de sous-modularité est une notion centrale pour caractériser la capacité de représentation des graphes. Elle possède de nombreuses définitions équivalentes. Si F est une fonction d'un ensemble, la sous-modularité s'écrit :

$$F(X \cap Y) + F(X \cup Y) \leq F(X) + F(Y) \quad (1.8)$$

où \cap est l'opérateur d'intersection et \cup l'opérateur d'union. Appliquée aux fonctions pseudo-booléennes, la sous-modularité s'écrit :

$$F(\mathbf{x} \vee \mathbf{y}) + F(\mathbf{x} \wedge \mathbf{y}) \leq F(\mathbf{x}) + F(\mathbf{y}) \quad (1.9)$$

où \vee est l'opérateur "ou" (maximum de deux booléens) et \wedge est l'opérateur "et" (minimum de deux booléens). Une autre définition équivalente met en jeu la dérivée seconde de la fonction :

$$\frac{\partial^2 F(\mathbf{x})}{\partial x_u \partial x_v} \leq 0 \quad (1.10)$$

La dérivée d'une fonction pseudo-booléenne peut être évaluée de la manière suivante :

$$\frac{\partial P(\mathbf{x})}{\partial x_u} = P(\mathbf{x}|x_u = 1) - P(\mathbf{x}|x_u = 0) \quad (1.11)$$

Lorsque la fonction P est quadratique, la condition devient, lorsque la fonction est écrite sous sa forme unique,

$$a_{uv} \leq 0, \forall (x_u, x_v) \in \mathbf{x}^2 \quad (1.12)$$

Cela équivaut à des coefficients c_{uv} non-négatifs dans P . Parmi les FPB quadratiques, seul un sous-ensemble peut être représenté par un graphe. En dérivant la forme générale de la posiforme issue d'un graphe données dans (1.5),

$$\frac{\partial^2 P_G(\mathbf{x})}{\partial x_u \partial x_v} = -(c_{uv} + c_{vu}) \leq 0 \quad (1.13)$$

Puisque $c_{uv} \geq 0$ et $c_{vu} \geq 0$, toutes les fonctions quadratiques représentables par un graphe sont sous-modulaires. Inversement on montre aisément que toute fonction linéaire ou affine est représentable par un graphe en appliquant la construction du Tableau 1.1

$$\begin{aligned} F(\mathbf{x}) &= a_\emptyset + \sum_i a_i x_i \\ &= a_\emptyset + \sum_i -a_i^- + a_i^- \bar{x}_i + a_i^+ x_i \end{aligned} \quad (1.14)$$

où $a^+ = \max(a, 0)$ et $a^- = \max(-a, 0)$. Dans le cas où f est une fonction quadratique sous-modulaire, on a :

$$\begin{aligned} F(\mathbf{x}) &= Lin(\mathbf{x}) + \sum_{i < j} a_{ij} x_i x_j \\ &= Lin(\mathbf{x}) + \sum_{i < j} -a_{ij} \bar{x}_i x_j \end{aligned} \quad (1.15)$$

où Lin est une forme linéaire quelconque. Si les coefficients a_{ij} sont négatifs, la fonction F peut s'écrire comme un graphe en appliquant la construction du Tableau 1.1. Un problème peut être représenté par un graphe, si et seulement si, il existe une écriture du problème sous la forme d'une fonction pseudo-booléenne quadratique sous-modulaire.

1.4 Contraction du graphe

La contraction de graphe est une opération qui permet de simplifier un graphe en fusionnant plusieurs nœuds. Le nœud restant représente les nœuds précédents sous la contrainte qu'ils aient la même affectation. Soit \mathbf{W} , l'ensemble des nœuds à contracter et \mathcal{W} le nœud restant, les arcs entrants ou sortants de \mathbf{W} ont alors respectivement pour arrivée et pour entrée un nouveau nœud \mathcal{W} . Dans le cas particulier où \mathbf{W} contient la source (le puits), \mathcal{W} est alors la source (le puits) et les arcs entrants (sortants) sont retirés du graphe.

De façon équivalente sur la posiforme P_G il suffit de substituer les variables de nœuds de \mathbf{W} par la variable x_w , ce qui permet de regrouper les termes qui sont à la frontière de \mathbf{W} et annuler les termes qui sont internes à \mathbf{W} . Dans le cas où \mathbf{W} contient la source (le puits), on pose $x_w = 0$ ($x_w = 1$). Les termes des arcs à la frontière de \mathbf{W} s'annulent ou deviennent des termes unaires.

2 Minimisation d'un problème par coupure de graphe

La coupure de graphe est une méthode qui permet d'optimiser globalement des FPB sous-modulaires d'ordre 2 par recherche d'une coupe minimale. Ce problème, lié au problème de maximisation du flot, peut être résolu par plusieurs algorithmes. Les opérations réalisées par les algorithmes peuvent être interprétées comme des manipulations algébriques sur une FPB.

2.1 Recherche du flot maximal

2.1.1 Flot maximal/Coupe minimale

Un flot f et la valeur de flot $|f|$ sont définis comme

$$\begin{aligned} 0 \leq f_{uv} \leq c_{uv}, & \quad \forall (u, v) \in \mathcal{A}, \\ \sum_{u|(u,v) \in \mathcal{A}} f_{uv} = \sum_{u|(v,u) \in \mathcal{A}} f_{vu}, & \quad \forall v \in \mathcal{X}^*, \\ |f| = \sum_{u|(s,u) \in \mathcal{A}} f_{su}. & \end{aligned} \quad (1.16)$$

La première condition est que le flot est limité par les capacités des arcs. Lorsque le flot d'un arc est égal à la capacité, on dit que l'arc est saturé. La deuxième condition est que le flot entrant dans un nœud (hors source et puits) est toujours égal au flot sortant du nœud, enfin $|f|$ est la valeur du flot qui s'écoule de la source vers le puits. C'est une valeur positive et bornée par la somme des capacités du graphe, elle possède donc une limite appelée flot maximal. Lorsque le flot ne peut plus augmenter, on dit que le graphe est saturé.

Le problème de coupe minimale consiste à trouver la partition des nœuds qui minimise la valeur de la coupe :

$$\min_{\mathcal{S}, \mathcal{T}} \text{Coupe}(\mathcal{S}, \mathcal{T}) \quad (1.17)$$

Le problème du flot maximal est équivalent au problème de coupe minimale [Elias *et al.*, 1956]. Il est montré que la valeur de la coupe minimale est égale à la valeur de flot maximal et qu'une coupe est minimale, si et seulement si, les arcs qu'elle met en jeu sont saturés par le flot maximal.

2.1.2 Poussage de flot

Il existe deux familles d'algorithmes permettant de calculer le flot maximal et d'en déduire une coupe minimale. La première famille appelée poussage/ré-étiquetage (Goldberg et Tarjan [1988]). Elle fait intervenir la notion de *préflot*. A la différence du flot, le préflot en entrée d'un nœud peut être en excès par rapport au préflot en sortie. La seconde famille résout le problème par poussage de flot [Ford et Fulkerson, 1962]. Il s'agit de trouver itérativement un chemin de la source au puits dans lequel s'écoule un flot, appelé chemin augmentant, et d'appliquer le poussage de flot sur ce chemin.

Soit \mathcal{A}' un ensemble d'arc représentant un chemin dans le graphe, on définit $f_{\mathcal{A}'}$ le flot maximal qui peut s'écouler dans long du chemin comme :

$$f_{\mathcal{A}'} = \min_{(u,v) \in \mathcal{A}'} c_{uv} \quad (1.18)$$

Un chemin augmentant est chemin qui relie la source au puits, dont le flot maximal est non-nul et qui ne possède pas de boucle, c'est à dire que les nœuds mis en jeu dans le chemin sont distincts. L'opération de poussage de flot consiste à retrancher le flot $f_{\mathcal{A}'}$ aux capacités des arcs mis en jeu dans le chemin et à l'ajouter aux capacités des arcs frères. L'opération de poussage de flot consiste donc en un transfert de capacité entre arcs frères :

$$\forall (u, v) \in \mathcal{A}' \cap \mathcal{A}^*, \begin{cases} c_{uv} \leftarrow c_{uv} - f_{\mathcal{A}'} \\ c_{vu} \leftarrow c_{vu} + f_{\mathcal{A}'} \end{cases} \quad (1.19)$$

Le graphe modifié est appelé graphe résiduel et les capacités de ses arcs sont appelées capacités résiduelles. L'opération est répétée sur le graphe résiduel jusqu'à ce qu'aucun chemin augmentant ne soit trouvé.

Il est à noter que le poussage de flot, tel que défini ci-dessus, introduit des capacités sur des arcs dont l'arrivée est la source. De tels arcs n'interviennent jamais dans un chemin augmentant.

En effet, tout chemin augmentant commence par la source et ne peut repasser par la source car il ne doit pas former de boucle. C'est pourquoi, dans la plupart des algorithmes, ces arcs n'apparaissent pas dans le graphe résiduel. Il en est de même des arcs qui ont pour origine le puits. Dans ce cas, l'opération de poussage de flot s'écrit :

$$\begin{aligned} c_{uv} &\leftarrow c_{uv} - f_{\mathcal{A}'}, \forall (u, v) \in \mathcal{A}' \\ c_{uv} &\leftarrow c_{uv} - f_{\mathcal{A}'}, \forall (u, v) \in \mathcal{A}' \cap \mathcal{A}^* \end{aligned} \quad (1.20)$$

2.1.3 Algorithme Maxflow

Les algorithmes de poussage de flot diffèrent dans la recherche de chemin augmentant. Une des premières méthodes est de construire un arbre de recherche depuis la source et le puits jusqu'à trouver un chemin augmentant, de pousser le flot et d'itérer. La particularité de l'algorithme Maxflow [Boykov et Kolmogorov, 2004], [Kolmogorov *et al.*, 2014] par rapport aux algorithmes antérieurs de poussage de flot réside dans la réutilisation des arbres de l'étape antérieure. Cette évolution a permis des gains importants en terme de temps de calcul et a fait le succès de cet algorithme. L'algorithme se décompose en trois étapes réalisées itérativement : 1) la croissance de l'arbre source et de l'arbre puits, 2) si un chemin augmentant est trouvé, le poussage du flot, 3) l'adoption des nœuds qui auraient été séparés de leur arbre. L'algorithme s'arrête lorsque plus aucun chemin augmentant n'est trouvé.

2.2 Poussage de flot et FPB

Nous avons vu dans la Section 1.2 qu'un graphe peut être représenté par une fonction pseudo-booléenne. Il existe un lien entre les chemins augmentants et le poussage de flot avec l'arithmétique des fonctions booléennes.

2.2.1 Chemin augmentant et somme alternée

Trouver un ensemble d'arcs formant un chemin augmentant revient à appliquer à la posiforme une identité remarquable appelée somme alternée,

$$x_0 + \sum_{i=0}^n \bar{x}_i x_{i+1} + \bar{x}_{n+1} = 1 + \sum_{i=0}^n x_i \bar{x}_{i+1} \quad (1.21)$$

Le terme de gauche correspond au flot le long du chemin augmentant qui va être retranché dans le graphe résiduel, et le terme de droite correspond aux capacités ajoutées aux arcs frères du graphe résiduel. Un chemin augmentant de valeur f , $\mathcal{S} \rightarrow \mathcal{V}_0 \rightarrow \mathcal{X}_1 \rightarrow \mathcal{X}_2 \rightarrow \mathcal{T}$ correspond au terme $f(x_0 + \bar{x}_0 x_1 + \bar{x}_1 x_2 + \bar{x}_2)$ dans la posiforme. Trouver et pousser un chemin augmentant dans un graphe revient à chercher une somme alternée dans la posiforme et pousser un flot revient à appliquer l'identité 1.21. Un exemple de correspondance entre poussage de flot et posiforme est montré dans la Figure 1.5.

2.2.2 Problème de maximisation-minimisation

Soit P_G la posiforme représentant le graphe initial, K une constante et Res_G la posiforme du graphe résiduel tels que $P_G(\mathbf{x}) = K + Res_G(\mathbf{x})$, alors

$$K \leq \min_{\mathbf{x}} P_G(\mathbf{x}) \leq K + Res_G(\mathbf{x}) \quad (1.22)$$

S'il existe une affectation \mathbf{x}^0 telle que $Res_G(\mathbf{x}^0) = 0$ alors $P_G(\mathbf{x}^0) = \min_{\mathbf{x}} P_G(\mathbf{x}) = K$. L'algorithme Maxflow cherche donc à résoudre un problème maximisation-minimisation : maximisation

de la borne K et minimisation de la posiforme Res . La question est de savoir si à l'issue des différents poussages de flot, il existe une configuration qui annule la posiforme résiduelle et comment trouver une telle affectation.

Les chemins augmentants peuvent être trouvés en construisant deux arbres de recherche dans le graphe résiduel : l'un partant de source qui s'écoule dans le graphe, l'autre partant du puits et qui remonte dans le graphe. Dès qu'un nœud se trouve dans les deux arbres à la fois, un chemin augmentant est trouvé, et un flot est poussé. L'étape est répétée jusqu'à ce qu'il n'y ait plus de nouveau chemin augmentant. Le temps de convergence est fini pour des capacités entières. Il est possible que le temps de convergence soit infini lorsque les capacités sont réelles. A l'issue de ces étapes aucun nœud n'est atteignable à la fois de la source et du puits. Les nœuds peuvent être classés en trois catégories : atteignables depuis la source notée \mathcal{S}^* , depuis le puits \mathcal{T}^* , depuis aucun des deux \diamond . Affecter le premier groupe \mathcal{S}^* à \mathcal{S} ($x = 0$) et le second \mathcal{T}^* à \mathcal{T} ($x = 1$) ne coûte rien. De même affecter les nœuds du troisième groupe \diamond tous à \mathcal{S} ou tous à \mathcal{T} ne coûte rien. Il existe donc deux affectations qui annulent la posiforme résiduelle qui sont déduites directement du graphe résiduel,

$$Coupe(\{\mathcal{S}^*, \diamond\}, \mathcal{T}^*) = Coupe(\mathcal{S}^*, \{\diamond, \mathcal{T}^*\}) = \min_x P_G(x) \quad (1.23)$$

Les deux coupes minimales du graphe sont représentées dans la Figure 1.3. A titre d'exemple, le poussage de flot de la fonction de (1.7a) est décrit sous la forme de graphe dans la Figure 1.5. A l'issue du poussage de flot sur ces graphes, aucun nœud ne peut être atteint depuis la source ou depuis le puits. Ils peuvent être affectés tous à la source ou tous au puits. Il est à noter que l'on n'obtient que deux solutions parmi les trois possibles.

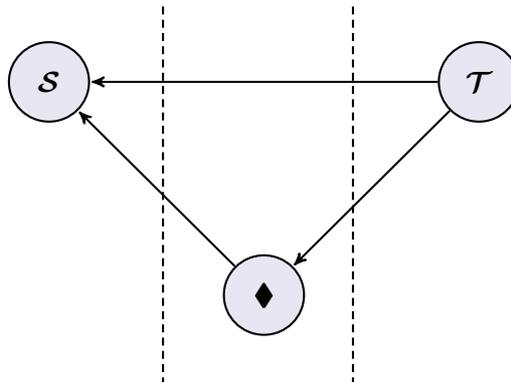


Figure 1.3 – Schéma d'un graphe après poussage de flot et deux coupes minimales.

3 Extension de la formulation par coupure de graphe

Dans la Section 1, nous avons vu que les graphes-ST correspondent à des FPB quadratiques sous-modulaires. Il s'agit d'une limitation dans l'utilisation des graphes, cependant différentes constructions et méthodes ont été proposées afin d'étendre les coupes de graphes à des familles plus larges de problèmes. Sous certaines conditions, les problèmes multi-labels peuvent être représentés par des graphes et résolus de manière exacte (Section 3.1). Le cas multi-labels peut être traité de manière itérative par une méthode approchée qui consiste à décomposer le problème en sous-problèmes binaires : les fusions binaires (Section 3.2). Dans le cas non-sous-modulaire, une solution partielle peut être trouvée avec une construction de graphe particulière (Section 3.3). Enfin les coupures de graphe ont été étendues à des fonctions d'ordre supérieur par réduction d'ordre de la FPB (Section 3.4).

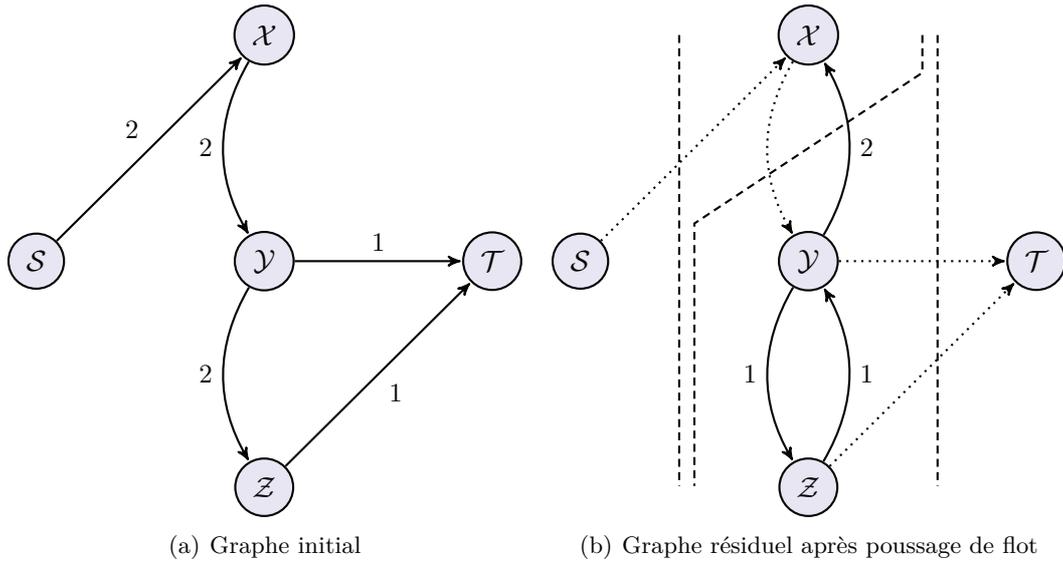


Figure 1.4 – Exemple d'un graphe avec trois coupes minimales. Les coupes minimales du graphe sont des coupes nulles du graphe résiduel (cf. Section 2.2.2)

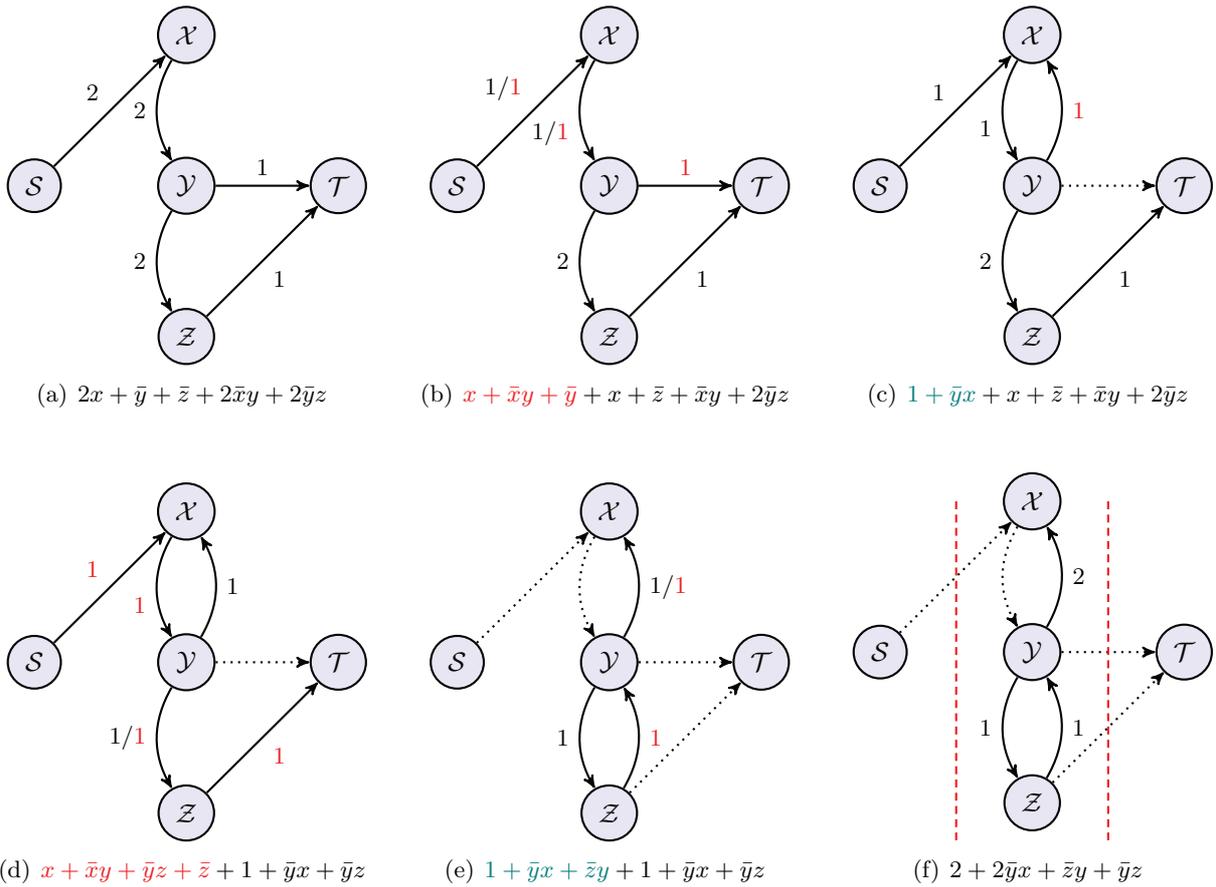


Figure 1.5 – Exemple de poussage de flot appliqué à la fonction de (1.7a). Les termes avant et après application de la somme alternée sont écrits en rouge et en bleu.

3.1 Résolution exacte des problèmes multi-labels

Nous souhaitons minimiser une énergie multi-labels de la forme :

$$E(\mathbf{l}) = \sum_{p \in \mathcal{I}} D_p(l_p) + \sum_{(p,q) \in \mathcal{N}'} R_{pq}(l_p, l_q) \tag{1.24}$$

1	0	1	2	3	4
x_4	0	1	1	1	1
x_3	0	0	1	1	1
x_2	0	0	0	1	1
x_1	0	0	0	0	1

Tableau 1.2 – Affectations réalisables et valeurs de label associées $l \in [0, 4]$, ($n = 5$)

où \mathbf{I} est un ensemble ordonné de pixels, \mathbf{N} l'ensemble des couples de pixels voisins de l'image et \mathbf{N}' l'ensemble $\{(p, q) \in \mathbf{N} | p < q, p \in I, q \in I\}$, \mathbf{l} est le vecteur de labels associés aux pixels p . l_p est un label qui prend sa valeur dans $[0, n - 1]$, D_p est la fonction d'attache aux données en p et R_{pq} la fonction de régularisation entre p et q . Sous certaines conditions, un problème multi-labels peut être résolu exactement par coupure de graphe comme montré dans [Ishikawa, 2003].

3.1.1 Représentation du problème par un graphe

Cette construction utilise une représentation des labels à l'aide de variables binaires. Le label attribué à une inconnue l_p est codé par un vecteur binaire \mathbf{x}^p , chaque variable de ce vecteur est donc notée x_k^p avec $k \in [1, n - 1]$. Les capacités associées au vecteur \mathbf{x}^p sont notées $c_{k,k+1}^p$ avec $c_{0,1}^p$ la capacité de la source vers le premier nœud et $c_{n-1,n}^p$ la capacité du dernier nœud au puits. Sous la contrainte que la coupe n'intersecte qu'une seule fois le vecteur de nœuds associé au label, la valeur du label optimal se déduit de la position dans le vecteur de cette intersection :

$$\forall p \in \mathbf{I}, l_p \text{ optimal} = l_p \iff \begin{cases} x_k^p = 0 \text{ si } k \leq l_p \\ x_k^p = 1 \text{ sinon} \end{cases} \quad (1.25)$$

Ainsi les configurations réalisables sont de la forme $\mathbf{0}^l \cdot \mathbf{1}^{n-1-l}$ où $\mathbf{0}^l$ est une chaîne de zéros de taille l , $\mathbf{1}^{n-1-l}$ est une chaîne de uns de taille $n-1-l$ (avec la convention que $\mathbf{0}^0$ et $\mathbf{1}^0$ sont des chaînes vides). L'opérateur de concaténation est noté " \cdot ". Les affectations réalisables pour $n = 5$ sont données dans le Tableau 1.2. Il y a 2^{n-1} configurations binaires du vecteur, on écarte les configurations non-désirées, c'est à dire celles où l'intersection n'est pas unique, en pénalisant les transitions de 1 vers 0 dans le sens source/puits du vecteur,

$$c_{l+1,l}^p = \infty, \quad \forall l \in [1, n - 2]. \quad (1.26)$$

Ces termes sont regroupés en un terme de contrainte,

$$\text{Contrainte}(\mathbf{x}) = \infty \sum_{p \in \mathbf{I}} \sum_{k=1}^{n-2} \bar{x}_{k+1}^p x_k^p \quad (1.27)$$

La structure du graphe présentée dans cette section diffère de celle décrite dans Ishikawa [2003]. En effet, une couche de nœuds supplémentaires située entre le premier nœud de chaque vecteur et la source est utilisée dans [Ishikawa, 2003]. Elle est reliée à la source par des liens infinis. Ces nœuds sont donc toujours atteignables depuis la source et n'interviennent pas dans l'optimisation (les variables associées sont constamment égales à zéro) et peuvent être assimilés à la source par contraction du graphe pour se ramener à la construction décrite ci-après.

Pour toutes les configurations réalisables de \mathbf{x} , nous cherchons l'égalité en valeur entre la posiforme et l'énergie multi-labels correspondante,

$$P_G(\mathbf{x}) = E(\mathbf{l}) \quad (1.28)$$

La forme générale de $P_G(\mathbf{x})$ est :

$$P_G(\mathbf{x}) = \sum_{p \in \mathbf{I}} U_p(\mathbf{x}^p) + \sum_{(p,q) \in \mathbf{I}^2, p < q} V_{pq}(\mathbf{x}^p, \mathbf{x}^q) + \text{Contrainte}(\mathbf{x}) \quad (1.29)$$

avec

$$U_p(\mathbf{x}^p) = c_\emptyset^p + c_0^p x_1^p + \sum_{a=1}^{n-2} c_a^p \bar{x}_a^p x_{a+1}^p + c_{n-1}^p \bar{x}_{n-1}^p \quad (1.30)$$

et

$$V_{p,q}(\mathbf{x}^p, \mathbf{x}^q) = \sum_{a=1}^{n-1} \sum_{b=1}^{n-1} c_{a,b}^{p,q} \bar{x}_a^p x_b^q + c_{b,a}^{q,p} \bar{x}_b^q x_a^p \quad (1.31)$$

avec $c_{u,v}^{p,q}$, $(u, v) \in [1, n-1]^2$ la capacité entre le u -ième nœud du vecteur correspondant au pixel p et le v -ième nœud du vecteur correspondant au pixel q . Pour toutes les coupes réalisables du graphe, l'équation (1.28) devient :

$$E(\mathbf{l}) = \sum_{p \in \mathcal{I}} c_\emptyset^p + c_{l_p, l_{p+1}}^p + \sum_{(p,q) \in \mathcal{N}, p > q} \sum_{a=1}^{l_p} \sum_{b=l_q+1}^{n-1} c_{a,b}^{p,q} + \sum_{a=l_p+1}^{n-1} \sum_{b=1}^{l_q} c_{b,a}^{q,p} \quad (1.32)$$

La structure de graphe correspondante est donnée dans la Figure 1.6.

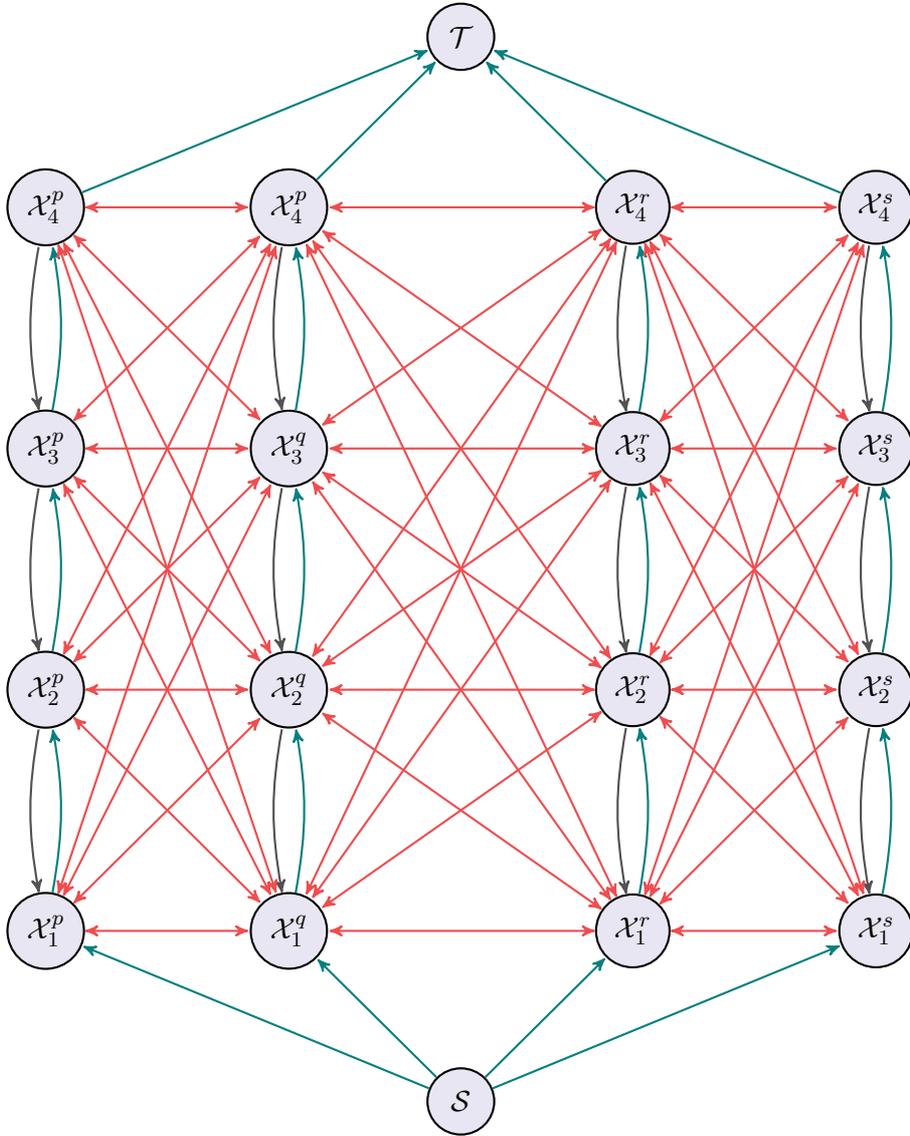


Figure 1.6 – Structure de graphe permettant de représenter un problème multi-labels, l'exemple correspond à une chaîne de 4 pixels (p, q, r, s) dont les labels prennent des valeurs entre 0 et 4. Les liens en gris sont des arcs de capacité infinie du terme de contraintes, les liens en bleu correspondent aux arcs mis en jeu dans les termes U de P_G et les liens en rouge correspondent aux arcs en jeu dans les termes V de P_G .

3.1.2 Contraintes sur la fonction de régularisation

Soit $E_{pq}(u, v) = E(\mathbf{l})_{l_p=u, l_q=v}$, par différenciation des énergies les capacités d'un couple d'arcs frères peuvent être isolées

$$c_{uv}^{pq} + c_{vu}^{qp} = E_{pq}(u-1, v) + E_{pq}(u, v-1) - E_{pq}(u, v) - E_{pq}(u-1, v-1) \quad (1.33)$$

Si $(p, q) \notin \mathbf{N}$, $c_{uv}^{pq} + c_{vu}^{qp} = 0$ alors $c_{uv}^{pq} = c_{vu}^{qp} = 0$ (les capacités sont non-négatives). Si $(p, q) \in \mathbf{N}$, alors (1.33) se simplifie après substitution dans (1.24),

$$c_{uv}^{pq} + c_{vu}^{qp} = R_{pq}(u-1, v) + R_{pq}(u, v-1) - R_{pq}(u, v) - R_{pq}(u-1, v-1) \quad (1.34)$$

Les capacités étant des valeurs non négatives, une condition nécessaire pour que l'énergie soit représentable est donc

$$0 \leq R_{pq}(u-1, v) + R_{pq}(u, v-1) - R_{pq}(u, v) - R_{pq}(u-1, v-1), \forall (p, q) \in \mathbf{N} \quad (1.35)$$

On remarque que la condition ne fait intervenir que la fonction de régularisation. Dans le cas fréquent où R_{pq} est une fonction r de la différence de label $d = l_p - l_q$ alors cette condition se réécrit plus simplement :

$$0 \leq r(d+1) + r(d-1) - 2r(d) \quad (1.36)$$

La convexité de la fonction r est donc une condition suffisante pour que l'énergie soit représentable. Les fonctions de régularisation convexes souvent utilisées sont la valeur absolue et le carré. Cette condition sur la fonction de régularisation constitue cependant une forte limitation pratique de l'utilisation de cette structure de graphe.

3.1.3 Construction

Il existe de nombreuses constructions équivalentes. On peut jouer sur deux aspects, 1) la balance entre les capacités des arcs frères qui relie deux vecteurs différents, 2) la disposition des coûts qui ne dépendent que d'un label. Nous prendrons les arcs frères entre vecteur de capacité égale,

$$c_{v,u}^{q,p} = c_{u,v}^{p,q} = \frac{1}{2} (R_{pq}(u-1, v) + R_{pq}(u, v-1) - R_{pq}(u, v) - R_{pq}(u-1, v-1)) \quad (1.37)$$

La somme des capacités se simplifie :

$$\begin{aligned} \sum_{b=b_1+1}^{b_2} c_{a,b}^{p,q} &= \frac{1}{2} (\cancel{R_{pq}(a-1, b_1+1)} + R_{pq}(a, b_1) - \cancel{R_{pq}(a, b_1+1)} - R_{pq}(a-1, b_1)) \\ &+ \frac{1}{2} (\cancel{R_{pq}(a-1, b_1+2)} + R_{pq}(a, b_1+1) - \cancel{R_{pq}(a, b_1+2)} - \cancel{R_{pq}(a-1, b_1+1)}) \\ &+ \dots \\ &+ \dots \\ &+ \frac{1}{2} (R_{pq}(a-1, b_2) + \cancel{R_{pq}(a, b_2-1)} - R_{pq}(a, b_2) - \cancel{R_{pq}(a-1, b_2-1)}) \\ &= \frac{1}{2} (R_{pq}(a-1, b_2) + R_{pq}(a, b_1) - R_{pq}(a, b_2) - R_{pq}(a-1, b_1)) \end{aligned} \quad (1.38)$$

En appliquant la double somme,

$$\sum_{a=a_1+1}^{a_2} \sum_{b=b_1+1}^{b_2} c_{a,b}^{p,q} = \frac{1}{2} (R_{pq}(a_1, b_2) + R_{pq}(a_2, b_1) - R_{pq}(a_1, b_1) - R_{pq}(a_2, b_2)) \quad (1.39)$$

En remplacement dans (1.32),

$$\begin{aligned}
E(\mathbf{l}) &= \sum_{p \in \mathbf{I}} c_{\emptyset}^p + c_{l_p, l_{p+1}}^p + \sum_{(p,q) \in \mathbf{N}'} \sum_{a=1}^{l_p} \sum_{b=l_q+1}^{n-1} c_{a,b}^{p,q} + \sum_{a=l_p+1}^{n-1} \sum_{b=1}^{l_q} c_{b,a}^{q,p} \\
&= \sum_{p \in \mathbf{I}} c_{\emptyset}^p + c_{l_p, l_{p+1}}^p \\
&\quad + \frac{1}{2} \sum_{(p,q) \in \mathbf{N}'} R_{pq}(0, n-1) + R_{pq}(l_p, l_q) - R_{pq}(0, l_q) - R_{pq}(l_p, n-1) \\
&\quad + \frac{1}{2} \sum_{(p,q) \in \mathbf{N}'} R_{pq}(n-1, 0) + R_{pq}(l_p, l_q) - R_{pq}(n-1, l_q) - R_{pq}(l_p, 0)
\end{aligned} \tag{1.40}$$

En remplaçant $E(\mathbf{l})$ par son expression (1.24) nous obtenons après simplification l'égalité suivante :

$$\begin{aligned}
\sum_{p \in \mathbf{I}} c_{\emptyset}^p + c_{l_p, l_{p+1}}^p &= \sum_{p \in \mathbf{I}} D_p(l_p) - \frac{1}{2} \sum_{(p,q) \in \mathbf{N}'} R_{pq}(n-1, 0) + R_{pq}(0, n-1) \\
&\quad + \frac{1}{2} \sum_{(p,q) \in \mathbf{N}} R_{pq}(l_p, 0) + R_{pq}(l_p, n-1)
\end{aligned} \tag{1.41}$$

En séparant les termes qui dépendent de l_p et les termes constants, on obtient les capacités à affecter aux arcs du graphe :

$$\begin{aligned}
c_u^p &= k_{\emptyset}^p + D_p(u) + \frac{1}{2} \sum_{q, (p,q) \in \mathbf{N}} (R_{pq}(u, 0) + R_{pq}(u, n-1)) \\
c_{\emptyset}^p &= -k_{\emptyset}^p - \frac{1}{2} \sum_{(p,q) \in \mathbf{N}'} (R_{pq}(n-1, 0) + R_{pq}(0, n-1))
\end{aligned} \tag{1.42}$$

avec k_{\emptyset}^p une constante qui permet de garantir la positivité des capacités c_u^p .

La construction d'Ishikawa [2003] met en jeu des arcs supplémentaires qui relient chaque nœud à la source et au puits. Ainsi les coûts qui dépendent des fonctions R_{pq} dans les capacités c_u^p sont transférés sur ces nouveaux arcs. Cette construction met en jeu plus d'arcs mais permet de dissocier les capacités qui correspondent à l'attache aux données et ceux qui correspondent à la régularisation.

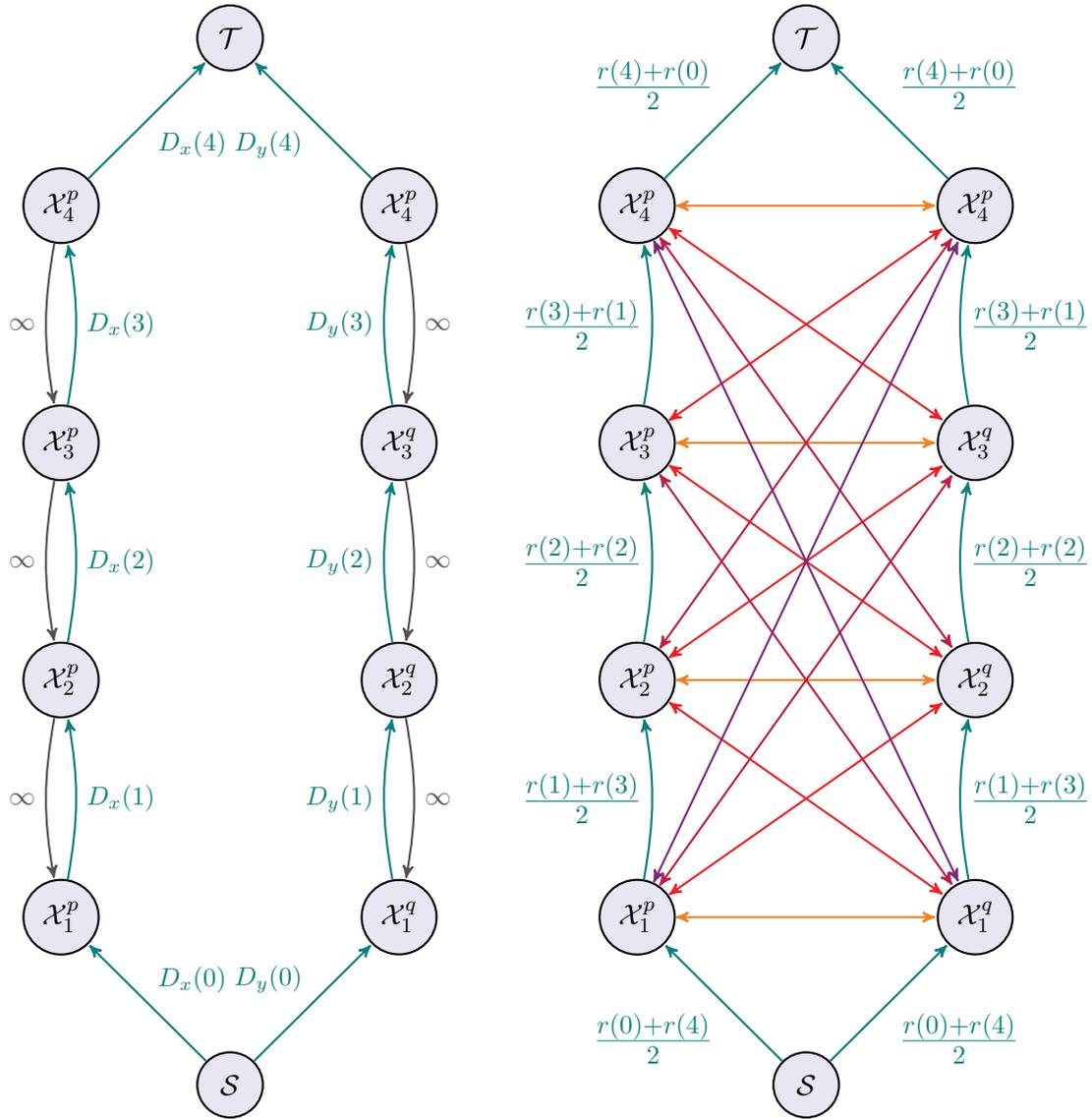
3.1.4 Cas particuliers

Tout d'abord, considérons des fonctions D_p et R_{pq} non-négatives. Dans le cas très fréquent où R est une fonction r paire de la différence de label $d = u - v$, on a :

$$\begin{aligned}
c_{u,v}^{p,q} &= \frac{1}{2} (r(d+1) + r(d-1) - 2r(d)) = \Delta^2(d) \\
c_{u,u+1}^p &= D_p(u) + \frac{1}{2} \sum_{q, (p,q) \in \mathbf{N}} r(u) + r(n-1-u) \\
c_{\emptyset}^p &= - \sum_{(p,q) \in \mathbf{N}'} r(n) + r(0)
\end{aligned} \tag{1.43}$$

où Δ^2 est la moitié de la dérivée seconde discrète de r . En conséquence, une condition suffisante pour que l'énergie soit représentable par un graphe est que la fonction r soit convexe. La construction du graphe dans ce cas est représentée sur un graphe à deux labels dans la Figure 1.7.

Dans le cas où r est λ fois la valeur absolue, le terme $\frac{1}{2} \sum_{(p,q) \in \mathbf{N}} r(u) + r(n-1-u)$ de la capacité $c_{u,u+1}^p$ est constant et égal à $n-1$. Cette constante sur chacune des capacités peut être



(a) Termes d'attache aux données et contraintes.

(b) Termes de régularisation; les couleurs représentent les capacités : $\Delta^2(0)$, $\Delta^2(1)$, $\Delta^2(2)$, $\Delta^2(3)$.

Figure 1.7 – Graphe multi-couches pour deux labels prenant leurs valeurs entre 0 et 4 avec une fonction de régularisation de la différence de label. Les capacités sont décomposées sur deux graphes différents pour une meilleure lisibilité. L'opérateur Δ^2 de la moitié de la dérivée seconde est définie dans (1.43).

transférée sur le terme constant c_\emptyset^p ce qui permet de l'annuler. Après simplification, on a :

$$\begin{aligned}
 c_{uu}^{pq} &= \lambda \\
 c_{uv}^{pq} &= 0; \quad u \neq v \\
 c_{u,u+1}^p &= D_p(u) \\
 c_\emptyset^p &= 0.
 \end{aligned} \tag{1.44}$$

La construction dans ce cas particulier peut être trouvée dans Veksler [1999].

Dans le cas où r est λ fois le carré, la construction se simplifie en

$$\begin{aligned} c_{uv}^{pq} &= \lambda \\ c_{u,u+1}^p &= D_p(u) + \frac{\lambda}{2} \sum_{(p,q) \in \mathcal{N}} (u^2 + (n-1-u)^2) \\ c_{\emptyset}^p &= -\lambda \sum_{(p,q) \in \mathcal{N}'} (n-1)^2. \end{aligned} \quad (1.45)$$

3.2 Résolution approchée des problèmes multi-labels

Nous avons vu dans la section précédente qu'un problème multi-labels défini par (1.24) ne peut être résolu exactement que pour une classe de fonctions de régularisation, c'est pourquoi des méthodes approchées ont été développées pour traiter les autres cas. Les méthodes que nous allons présenter mettent en jeu des fusions binaires. Elles consistent à ne prendre en compte que deux valeurs possibles pour chacune des variables mises en jeu. La résolution de chacun de ces sous-problèmes permet de diminuer itérativement la valeur de l'énergie.

3.2.1 Fusion binaire

Soit deux vecteurs de labels \mathbf{l}^1 et \mathbf{l}^2 , on définit le vecteur fusionné $\mathbf{l}_x^{1,2}$ paramétré par le vecteur de variables booléennes \mathbf{x} ,

$$\mathbf{l}_x^{1,2} = \mathbf{l}^1 \cdot \mathbf{x} + \mathbf{l}^2 \cdot \bar{\mathbf{x}} \quad (1.46)$$

où l'opérateur point "." est le produit terme à terme. Le problème de fusion binaire cherche une affectation minimale \mathbf{x}_0 telle que

$$\begin{aligned} F(\mathbf{x}_0) &= \min_{\mathbf{x}} F(\mathbf{x}) \\ F(\mathbf{x}) &= E(\mathbf{l}_x^{1,2}) \end{aligned} \quad (1.47)$$

où F est une fonction pseudo-booléenne. En remplaçant E par son expression (1.24),

$$\begin{aligned} F(\mathbf{x}) &= \sum_{p \in \mathcal{I}} D(l_p^1) x_p + D(l_p^2) \bar{x}_p \\ &+ \sum_{(p,q) \in \mathcal{N}'} R_{pq}(l_p^1, l_q^1) x_p x_q + R_{pq}(l_p^1, l_q^2) x_p \bar{x}_q + R_{pq}(l_p^2, l_q^1) \bar{x}_p x_q + R_{pq}(l_p^2, l_q^2) \bar{x}_p \bar{x}_q \\ &= \text{Lin}(\mathbf{x}) + \sum_{(p,q) \in \mathcal{N}'} (R_{pq}(l_p^1, l_q^1) + R_{pq}(l_p^2, l_q^2) - R_{pq}(l_p^1, l_q^2) - R_{pq}(l_p^2, l_q^1)) x_p x_q \end{aligned} \quad (1.48)$$

où Lin est une forme linéaire. La contrainte de sous-modularité sur la fonction F est alors :

$$R_{pq}(l_p^1, l_q^1) + R_{pq}(l_p^2, l_q^2) \leq R_{pq}(l_p^1, l_q^2) + R_{pq}(l_p^2, l_q^1), \forall (p, q) \in \mathcal{N}' \quad (1.49)$$

À partir de la fonction F écrite sous la forme :

$$\begin{aligned} F(\mathbf{x}) &= \sum_{p \in \mathcal{I}} D(l_p^1) x_p + D(l_p^2) \bar{x}_p \\ &+ \sum_{(p,q) \in \mathcal{N}'} -R_{pq}(l_p^2, l_q^1) \\ &+ \sum_{(p,q) \in \mathcal{N}'} R_{pq}(l_p^1, l_q^1) x_p + R_{pq}(l_p^2, l_q^1) \bar{x}_p \\ &+ \sum_{(p,q) \in \mathcal{N}'} R_{pq}(l_p^2, l_q^1) x_j + R_{pq}(l_p^2, l_q^2) \bar{x}_q \\ &+ \sum_{(p,q) \in \mathcal{N}'} (R_{pq}(l_p^1, l_q^2) + R_{pq}(l_p^2, l_q^1) - R_{pq}(l_p^1, l_q^1) - R_{pq}(l_p^2, l_q^2)) x_p \bar{x}_q \end{aligned} \quad (1.50)$$

la construction de graphe associée peut être déduite des règles du Tableau 1.1. La figure 1.8 représente cette construction pour une fonction de deux variables. On peut montrer que la capacité du lien entre deux variables est toujours positive du fait de la contrainte de sous-modularité; les termes linéaires sont positifs moyennant la réécriture suivante :

$$ax + b\bar{x} = m + (a - m)x + (b - m)\bar{x} \quad (1.51)$$

avec $m = \min(a, b)$.

3.2.2 Changement de variables

Échanger un label entre l^1 et l^2 à la position p ne modifie pas l'ensemble des solutions réalisables par fusion mais change la condition sur la fonction de régularisation, l'inégalité change de sens

$$R_{pq}(l_p^1, l_q^1) + R_{pq}(l_p^2, l_q^2) \geq R_{pq}(l_p^1, l_q^2) + R_{pq}(l_p^2, l_q^1), \forall (p, q) \in \mathbf{N} \quad (1.52)$$

Échanger un label à la position p revient à substituer x_p par $y_p = \bar{x}_p$. Ainsi par un changement de variable, un problème de fusion non-sous-modulaire peut devenir sous-modulaire. Toutefois trouver dans le cas général un changement de variable qui rend le problème sous-modulaire est un problème d'optimisation binaire difficile et l'existence d'une solution n'est pas toujours garantie. Par exemple la fonction $F(x_1, x_2, x_3) = x_1x_2 - x_2x_3 - x_1x_3$ ne peut pas être rendue sous-modulaire par changement de variable.

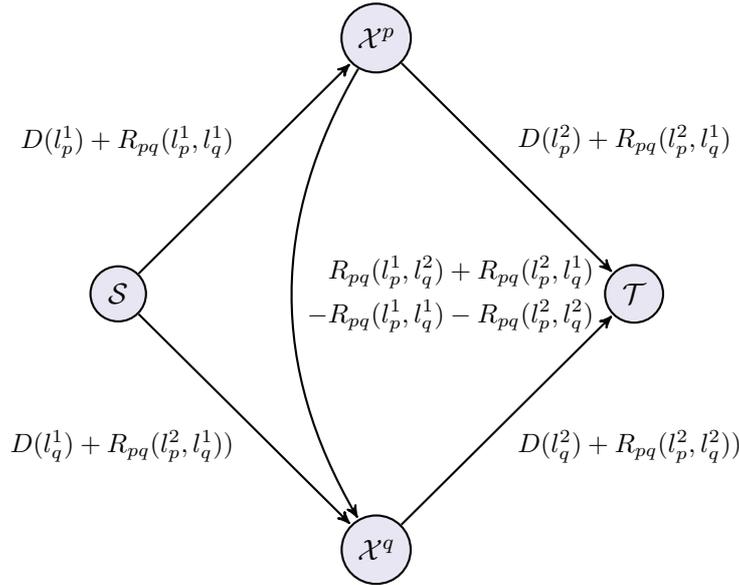


Figure 1.8 – Graphe élémentaire pour une fusion binaire, la valeur de la coupe est égale à la valeur de l'énergie à la constante $R_{pq}(l_p^2, l_q^1)$ près.

3.2.3 Cas de la fusion ternaire

Le problème que l'on cherche à résoudre est une fusion ternaire avec trois propositions pour chaque label. Les propositions de labels sont réarrangées en trois vecteurs l^- , l^0 et l^+ tels que :

$$l_p^- \leq l_p^0 \leq l_p^+, \forall p \quad (1.53)$$

Il s'agit d'un problème multi-labels limité à 3 labels. La construction de graphe correspondante revient à celle décrite dans le cas multi-labels (Section 3.1), en restreignant les labels possibles. Une construction pour résoudre ce problème est donnée Figure 1.9.

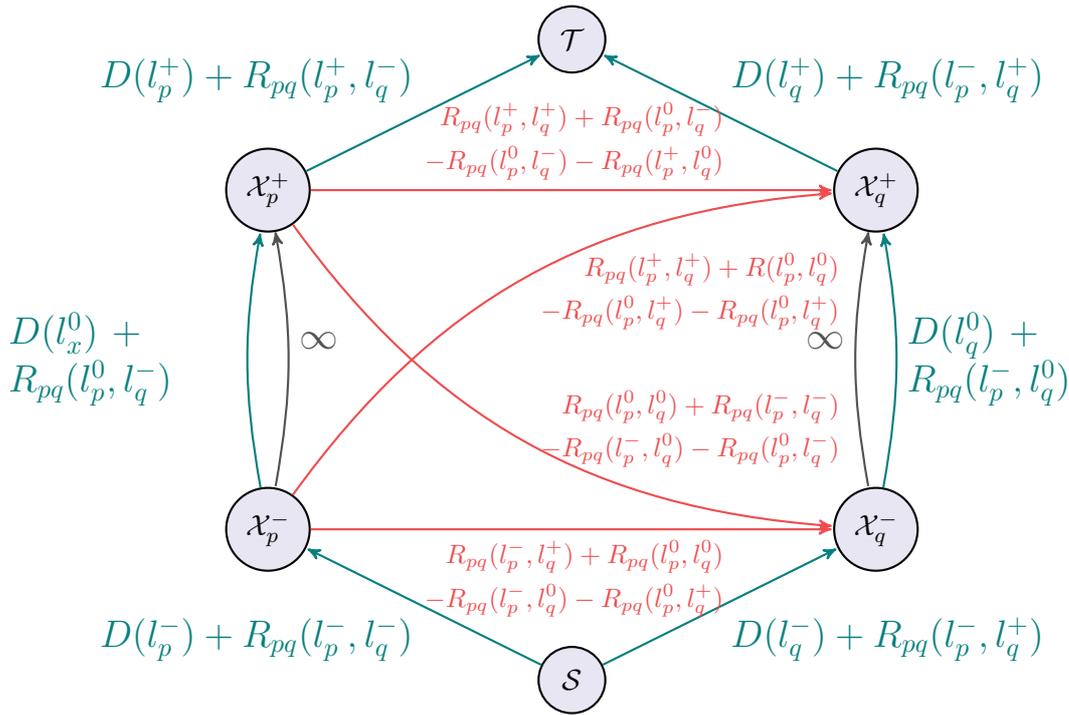


Figure 1.9 – Graphe élémentaire pour un saut ternaire, à la constante $-R(l_x^-, l_y^-)$ près

On peut définir le saut ternaire où les trois propositions sont : la solution courante, la solution courante plus une constante δ sur chaque label et la solution courante moins une constante δ sur chaque label. Le saut ternaire $-1/+1$ permet alors une descente de gradient de l'énergie simultanée sur toutes les variables.

3.3 Minimisation approchée des fonctions quadratiques (QPBO)

Nous avons vu que l'énergie d'un problème doit pouvoir s'écrire sous la forme d'une fonction pseudo-booléenne quadratique sous-modulaire pour pouvoir être représentable par un graphe et ainsi être résolue par coupure de graphe. Est-il possible de construire un graphe pour approcher le minimum d'une FPB quadratique non-sous-modulaire ?

3.3.1 Roof duality

Dans (1.22), le problème Maxflow est écrit sous la forme d'une minimisation-maximisation. Prenons maintenant P une posiforme quelconque, Res est obtenu par recherche de chemin augmentant :

$$K \leq \min_{\mathbf{x}} P(\mathbf{x}) \leq K + Res(\mathbf{x}) \quad (1.54)$$

Dans ce cas plus général, il n'y a pas de garantie que la constante K atteigne le minimum de P et la minimisation de la fonction Res , qui est elle-aussi quelconque, n'est pas forcément évidente. La méthode proposée est d'approcher la fonction $P(\mathbf{x})$ par une fonction $Q(\mathbf{x}, \mathbf{y})$ quadratique telle que $Q(\mathbf{x}, \bar{\mathbf{x}}) = P(\mathbf{x})$ et qui minore $P(\mathbf{x})$

$$\min_{\mathbf{x}, \mathbf{y}} Q(\mathbf{x}, \mathbf{y}) \leq \min_{\mathbf{x}} P(\mathbf{x}) \quad (1.55)$$

Minimiser Q donne une borne minimale à la fonction P . Nous imposons maintenant que Q soit sous-modulaire. Nous pouvons trouver deux vecteurs $(\mathbf{x}^{opt}, \mathbf{y}^{opt})$ qui minimisent Q :

$$\min_{\mathbf{x}, \mathbf{y}} Q(\mathbf{x}, \mathbf{y}) = K' + Res(\mathbf{x}^{opt}, \mathbf{y}^{opt}) = K' \quad (1.56)$$

L'écart RD entre le minimum de la fonction Q et le minimum de la fonction P est appelé "roof duality" :

$$\begin{aligned} RD &= \min_{\mathbf{x}} P(\mathbf{x}) - \min_{\mathbf{x}, \mathbf{y}} Q(\mathbf{x}, \mathbf{y}) \\ &= \min_{\mathbf{x}} Res(\mathbf{x}, \bar{\mathbf{x}}) \end{aligned} \quad (1.57)$$

En conséquence $RD = 0$ si est seulement si une des affectations minimales de Q respecte la contrainte $\mathbf{y}^{opt} = \bar{\mathbf{x}}^{opt}$. Nous cherchons parmi les fonctions Q celle qui minimise la valeur de RD , ce qui revient à chercher la fonction dont le minimum est maximal. Dans la pratique, le minimum n'est pas toujours disponible, c'est donc une borne inférieure b_{min} qui est maximisée :

$$\max_{Q, b_{min}} b_{min} \text{ sous les contraintes, } \begin{cases} Q(\mathbf{x}, \mathbf{y}) \geq b_{min}, \forall (\mathbf{x}, \mathbf{y}) \\ Q(\mathbf{x}, \bar{\mathbf{x}}) = P(\mathbf{x}), \forall \mathbf{x} \\ Q \text{ sous-modulaire} \end{cases} \quad (1.58)$$

3.3.2 Symétrie

La fonction Q se décompose en une composante symétrique et une composante asymétrique :

$$\begin{aligned} Q(\mathbf{x}, \mathbf{y}) &= Sym(\mathbf{x}, \mathbf{y}) + Asym(\mathbf{x}, \mathbf{y}) \\ Sym(\mathbf{x}, \mathbf{y}) &= \frac{1}{2}(Q(\mathbf{x}, \mathbf{y}) + Q(\bar{\mathbf{y}}, \bar{\mathbf{x}})) \\ Asym(\mathbf{x}, \mathbf{y}) &= \frac{1}{2}(Q(\mathbf{x}, \mathbf{y}) - Q(\bar{\mathbf{y}}, \bar{\mathbf{x}})) \end{aligned} \quad (1.59)$$

Kahl et Strandmark [2012] montrent que si Q est une solution du problème (1.58), Sym est aussi une solution de ce problème :

— Le minimum de Sym est supérieur ou égal au minimum de Q :

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} Q(\mathbf{x}, \mathbf{y}) &= \min_{\mathbf{x}, \mathbf{y}} Q(\bar{\mathbf{y}}, \bar{\mathbf{x}}) \\ &= \min_{\mathbf{x}, \mathbf{y}} Sym(\mathbf{x}, \mathbf{y}) - Asym(\mathbf{x}, \mathbf{y}) \\ &\leq \min_{\mathbf{x}, \mathbf{y}} Sym(\mathbf{x}, \mathbf{y}) \end{aligned} \quad (1.60)$$

— En remplaçant \mathbf{y} par $\bar{\mathbf{x}}$, on obtient :

$$Sym(\mathbf{x}, \bar{\mathbf{x}}) = P(\mathbf{x}) \quad (1.61)$$

— Si Q est sous-modulaire alors Sym , qui est une somme de la fonction Q , est aussi sous-modulaire.

Ainsi, nous nous restreignons aux fonctions Q symétriques

$$Q(\mathbf{x}, \mathbf{y}) = Q(\bar{\mathbf{y}}, \bar{\mathbf{x}}) \quad (1.62)$$

Avec Q symétrique et sous-modulaire (1.9) on a :

$$\begin{aligned} Q(\mathbf{x}, \mathbf{y}) &= \frac{1}{2}(Q(\mathbf{x}, \mathbf{y}) + Q(\bar{\mathbf{y}}, \bar{\mathbf{x}})) \geq \\ &\frac{1}{2}(Q(\mathbf{x} \vee \bar{\mathbf{y}}, \bar{\mathbf{x}} \vee \mathbf{y}) + Q(\mathbf{x} \wedge \bar{\mathbf{y}}, \bar{\mathbf{x}} \wedge \mathbf{y})) = Q(\mathbf{x} \wedge \bar{\mathbf{y}}, \bar{\mathbf{x}} \wedge \mathbf{y}) \end{aligned} \quad (1.63)$$

Cela signifie que remplacer (\mathbf{x}, \mathbf{y}) par $(\mathbf{x} \wedge \bar{\mathbf{y}}, \bar{\mathbf{x}} \wedge \mathbf{y})$ permet de diminuer l'énergie. Cette opération consiste à remplacer les couples d'affectation $(1, 1)$ par $(0, 0)$ comme nous pouvons le voir dans la table de vérité Tableau 1.3. De plus si l'affectation (\mathbf{x}, \mathbf{y}) est minimale, l'affectation $(\mathbf{x} \wedge \bar{\mathbf{y}}, \bar{\mathbf{y}} \wedge \mathbf{x})$ est elle aussi minimale. Nous pouvons donc restreindre la recherche du minimum de Q aux vecteurs d'affectations $(0, 0)$, $(0, 1)$ et $(1, 0)$.

$(\mathbf{x} \wedge \bar{\mathbf{y}}, \bar{\mathbf{y}} \wedge \mathbf{x})$	0	1
0	(0,0)	(1,0)
1	(0,1)	(0,0)

Tableau 1.3 – L’expression $(\mathbf{x} \wedge \bar{\mathbf{y}}, \bar{\mathbf{y}} \wedge \mathbf{x})$ remplace l’affectation $(x, y) = (1, 1)$ par $(x, y) = (0, 0)$.

$(\mathbf{x}_1, \mathbf{y}_1) \sqcup (\mathbf{x}_2, \mathbf{y}_2)$	$(\mathbf{x}_1, \mathbf{y}_1)$			
$(\mathbf{x}_2, \mathbf{y}_2)$	(0,0)	(0,1)	(1,0)	(1,1)
(0,0)	(0,0)	(0,1)	(1,0)	(0,0)
(0,1)	(0,1)	(0,1)	(0,0)	(0,0)
(1,0)	(1,0)	(0,0)	(1,0)	(0,0)
(1,1)	(0,0)	(0,0)	(0,0)	(0,0)

Tableau 1.4 – Table de vérité de l’opérateur \sqcup . Les couples d’affectation $(1, 1)$ mis en jeu sont remplacés par $(0, 0)$.

3.3.3 Autarcie et persistance

Soit $(\mathbf{x}^{opt}, \mathbf{y}^{opt})$ une affectation minimale de Q et \mathbf{x} une affectation quelconque, on définit le vecteur \mathbf{x}' tel que :

$$x'_i = \begin{cases} x_i^{opt} & \text{si } x_i^{opt} = \bar{y}_i^{opt} \\ x_i & \text{sinon} \end{cases} \quad (1.64)$$

La propriété d’autarcie est que $P(\mathbf{x}') \leq P(\mathbf{x})$. La propriété de persistance est que si \mathbf{x} minimise P alors \mathbf{x}' minimise P . L’autarcie implique donc la persistance. Une conséquence de cette propriété est que les éléments de $(\mathbf{x}^{opt}, \mathbf{y}^{opt})$ qui sont $(0, 1)$ et $(1, 0)$ donnent une affectation partielle des affectations minimales de P .

Kahl et Strandmark [2012] donnent la démonstration de cette propriété lorsque la fonction Q est symétrique et sous-modulaire. On définit les vecteurs \mathbf{x}^* et \mathbf{y}^* tels que $(\mathbf{x}^*, \mathbf{y}^*) = (\mathbf{x}^{opt} \vee \bar{\mathbf{y}}^{opt}, \bar{\mathbf{x}}^{opt} \vee \mathbf{y}^{opt})$; d’après (1.63), $\mathbf{x}^*, \mathbf{y}^*$ est aussi une affectation minimale. Pour écrire cette transformation en opérations booléennes, on définit l’opérateur noté \sqcup tel que

$$(\mathbf{x}_1, \mathbf{y}_1) \sqcup (\mathbf{x}_2, \mathbf{y}_2) = ((\mathbf{x}_1 \vee \mathbf{x}_2) \wedge \overline{(\mathbf{y}_1 \vee \mathbf{y}_2)}, (\mathbf{y}_1 \vee \mathbf{y}_2) \wedge \overline{(\mathbf{x}_1 \vee \mathbf{x}_2)}) \quad (1.65)$$

La table de vérité de cette transformation est donnée dans le Tableau 1.4. Cet opérateur remplace les couples $(1, 1)$ par $(0, 0)$. En prenant $y_1 = \bar{x}_1$, on obtient la transformation décrite en (1.64). Ainsi l’affectation \mathbf{x}' s’écrit,

$$(\mathbf{x}', \bar{\mathbf{x}}') = ((\mathbf{x}, \bar{\mathbf{x}}) \sqcup (\mathbf{x}^*, \mathbf{y}^*)) \sqcup (\mathbf{x}^*, \mathbf{y}^*) \quad (1.66)$$

Or Q étant sous-modulaire, par définition :

$$Q(\mathbf{x}, \bar{\mathbf{x}}) + Q(\mathbf{x}^*, \mathbf{y}^*) \geq Q(\mathbf{x} \vee \mathbf{x}^*, \bar{\mathbf{x}} \vee \mathbf{y}^*) + Q(\mathbf{x} \wedge \mathbf{x}^*, \bar{\mathbf{x}} \wedge \mathbf{y}^*) \quad (1.67)$$

$Q(\mathbf{x}^*, \mathbf{y}^*)$ est minimal et Q est symétrique d’où :

$$Q(\mathbf{x}, \bar{\mathbf{x}}) \geq Q(\mathbf{x} \vee \mathbf{x}^*, \bar{\mathbf{x}} \vee \mathbf{y}^*) \quad (1.68)$$

En appliquant l’inégalité (1.68) sur le terme de droite on obtient :

$$Q(\mathbf{x} \vee \mathbf{x}^*, \bar{\mathbf{x}} \vee \mathbf{y}^*) \geq Q((\mathbf{x} \vee \mathbf{x}^*) \wedge \overline{(\bar{\mathbf{x}} \vee \mathbf{y}^*)}, (\bar{\mathbf{x}} \vee \mathbf{y}^*) \wedge \overline{(\mathbf{x} \vee \mathbf{x}^*)}) = Q((\mathbf{x}, \bar{\mathbf{x}}) \sqcup (\mathbf{x}^*, \mathbf{y}^*)) \quad (1.69)$$

Ainsi appliquer l’opérateur d’union \sqcup par $(\mathbf{x}^*, \mathbf{y}^*)$ diminue l’énergie, en appliquant cette opération deux fois on obtient,

$$Q(\mathbf{x}, \bar{\mathbf{x}}) \geq Q((\mathbf{x}, \bar{\mathbf{x}}) \sqcup (\mathbf{x}^*, \mathbf{y}^*)) \sqcup (\mathbf{x}^*, \mathbf{y}^*) = Q(\mathbf{x}', \bar{\mathbf{x}}') \quad (1.70)$$

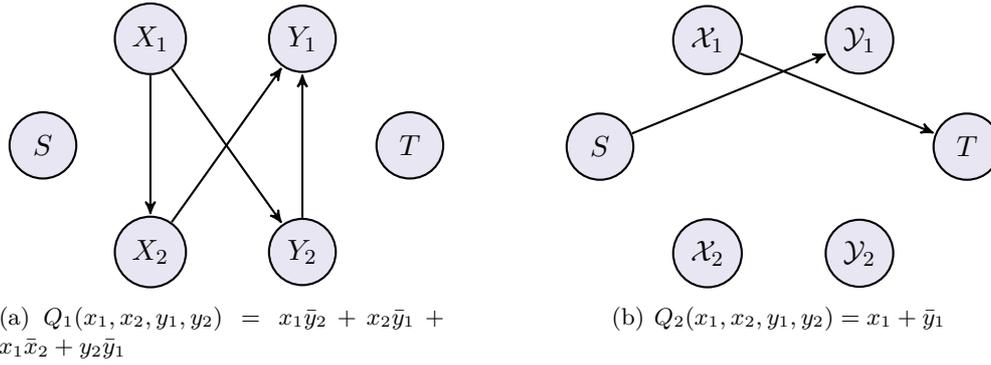


Figure 1.10 – $P(x_1, x_2) = x_1x_2 + x_1\bar{x}_2$ admet Q_1 et Q_2 comme relaxation qui atteignent la même borne ($RD = 0$) mais Q_1 donne la solution pour P (\diamond, \diamond) et Q_1 donne $(0, \diamond)$.

d'où,

$$P(\mathbf{x}) \geq P(\mathbf{x}') \quad (1.71)$$

Ainsi, la solution partielle obtenue par la minimisation de Q est un sous-ensemble d'une affectation minimale de P . Dans le cas où $RD = 0$, la solution est la plus complète; elle correspond alors à l'affectation commune à tous les minima (les valeurs restantes pouvant être fixées toutes à 0 ou toutes à 1). En principe minimiser RD permet d'obtenir une solution plus complète. Toutefois pour une même borne de RD , la solution partielle peut être plus ou moins complète comme l'illustre la Figure 1.10.

3.3.4 QPBO

La construction proposée par Boros et Hammer [2002] connue sous le nom de *QPBO* (Optimisation de fonction pseudo-booléenne quadratique) donne la meilleure borne RD dans le cas quadratique comme montré dans Kolmogorov [2010] Kahl et Strandmark [2012]. Soit une FPB quadratique d'écriture générale,

$$F(\mathbf{x}) = a_\emptyset + \sum_{p \in \mathbf{I}} a_i x_p + \sum_{(p,q) \in \mathbf{N}} a_{pq} x_p x_q \quad (1.72)$$

réécrit comme la posiforme

$$P(\mathbf{x}) = a_\emptyset + \sum_{p \in \mathbf{I}} c_p^+ x_p + \sum_{p \in \mathbf{I}} c_p^- \bar{x}_p + \sum_{(p,q) \in \mathbf{N}} c_{pq}^+ x_p x_q + \sum_{(p,q) \in \mathbf{N}} c_{pq}^- \bar{x}_p x_q \quad (1.73)$$

avec les valeurs c non-négatives et $c_{pq}^+ = \max(a_{pq}, 0)$ et $c_{pq}^- = \max(-a_{pq}, 0)$. Pour chaque variable x , une variable y est introduite. Le couple de variable (x_u, y_u) est représenté par les nœuds \mathcal{X}_u et \mathcal{Y}_u . La règle générale de construction pour un terme c_{uv} est un arc $\mathcal{Y}_u \rightarrow \mathcal{X}_v$ et un arc $\mathcal{X}_v \rightarrow \mathcal{Y}_u$ de capacité $\frac{c_{uv}}{2}$. Les termes unaires peuvent être interprétés comme des termes faisant intervenir la source ou le puits avec un autre nœud, d'où les règles de la construction du graphe décrites dans le Tableau 1.5.

Terme dans la posiforme	Arcs dans le graphe	Capacité
$c_p^+ x_u$	$\mathcal{S} \rightarrow \mathcal{X}_p, \mathcal{Y}_p \rightarrow \mathcal{T}$	$c_p^+ / 2$
$c_p^- \bar{x}_p$	$\mathcal{S} \rightarrow \mathcal{Y}_p, \mathcal{X}_p \rightarrow \mathcal{T}$	$c_p^- / 2$
$c_{pq}^+ \bar{x}_p x_q$	$\mathcal{X}_p \rightarrow \mathcal{X}_q, \mathcal{Y}_p \rightarrow \mathcal{Y}_q$	$c_{pq}^+ / 2$
$c_{pq}^- x_p x_q$	$\mathcal{Y}_p \rightarrow \mathcal{X}_q, \mathcal{X}_p \rightarrow \mathcal{Y}_q$	$c_{pq}^- / 2$

Tableau 1.5 – Table de correspondance entre les termes de la posiforme et les arcs dans le graphe QPBO.

Tous les termes de la posiforme sont dédoublés et disposés de manière symétrique. A l'issue de l'optimisation trois cas sont possibles pour une paire de nœuds,

$$(\mathcal{X}_p, \mathcal{Y}_p) \in \{(\mathcal{S}^*, \mathcal{T}^*), (\mathcal{T}^*, \mathcal{S}^*), (\diamond, \diamond)\} \quad (1.74)$$

Dans le cas où le couple de nœuds est atteignable, si $\mathcal{P} \in \mathcal{S}^*$, x_p est affecté à 0; inversement si $\mathcal{P} \in \mathcal{T}^*$, x_p est affecté à 1. Affecter les nœuds non-atteignables tous à la source ou tous au puits, permet de minimiser Q , cependant comme la contrainte $x = \bar{y}$ n'est pas respectée, il n'y a pas de garantie d'optimalité pour P . Néanmoins grâce à la propriété de persistance, l'affectation partielle est un sous-ensemble d'affectation minimale. Si la fonction P est sous-modulaire, alors les variables x et y peuvent être séparées et par symétrie $Q(x, y) = \frac{1}{2}(P(x) + P(\bar{y})) = P(x)$. $P(x)$ correspond alors à la construction de la fonction P dans le cas sous-modulaire.

3.3.5 Les heuristiques

Même dans le cas où la RD est minimale, toutes les affectations ne sont pas connues. L'affectation de ces variables a un impact sur l'optimalité de la solution. Ainsi plusieurs heuristiques ont été utilisées pour diminuer l'énergie de la solution et le nombre de variables non affectées.

La méthode la plus simple est d'affecter arbitrairement toutes les variables à 0 ou toutes à 1 dans P (donc à $(0, 1)$ et $(1, 0)$ dans Q). Les deux solutions sont minimales dans Q mais peuvent avoir des valeurs différentes dans P . Une variante est de calculer les deux possibilités et de choisir le minimum des deux. Toutefois lorsque le problème est difficile, de telles approches ne permettent pas de minimiser efficacement la fonction. Woodford *et al.* [2009] proposent de séparer les couples de variables non affectés en composantes connexes comme dans [Billionnet et Jaumard, 1989], et d'appliquer la méthode précédente sur chaque composante.

Une autre approche est de fixer arbitrairement un couple de variables et de renouveler le poussage de flot, et de réitérer l'opération. Ainsi, un nœud ou plus est déterminé à chaque étape. Le résultat obtenu est le minimum du problème initial sous contraintes d'égalité de certaines variables. S'il existe un minimum de P qui respecte ces contraintes, alors l'affectation est minimale autrement la solution est sous-optimale.

Boros *et al.* [2008] proposent la technique du "Probe" (QPBO-P) dont Rother *et al.* [2007] donnent une implémentation. Le principe est de sélectionner un couple de variables (x, y) dans Q , de les fixer alternativement à $(0, 1)$ et $(1, 0)$ et de pousser le flot dans les deux cas. Ainsi de nouveaux couples peuvent être affectés, on s'intéresse à deux cas de figure, 1) un couple de variables possède la même affectation dans les deux cas. Alors cette affectation fait partie du minimum global et on retire les nœuds correspondants par contraction du graphe, 2) le couple change d'affectation dans les deux cas, il peut avoir le même comportement ou inverse. Si le comportement est inverse, alors la signification de la variable est échangée de sorte que l'ensemble de ces couples de variables ont le même comportement dans la solution, et ces nœuds ayant la même affectation sont contractés. De plus une dernière étape de simplifications permet éventuellement de réduire l'amplitude des termes non-sous-modulaires, ce qui en principe augmente le nombre de variables affectées dans la solution.

3.4 Réduction d'ordre

Les algorithmes présentés précédemment ne traitent que des FPB quadratiques ce qui limite les applications possibles. Toutefois, il est possible de réécrire le problème de minimisation d'une fonction d'ordre supérieur à l'aide d'une fonction quadratique. Une telle réécriture du problème est appelée réduction d'ordre. Commençons par un exemple simple : on veut trouver une fonction quadratique qui égale un terme cubique

$$x_i x_j x_k = a x_i x_j + b x_j x_k + c x_i x_k + d x_i + e x_j + d x_k + f \quad (1.75)$$

Il y a 8 affectations possibles qui donnent chacune une équation et seulement 7 coefficients à trouver. Le système n'a donc pas de solution générale, ainsi les stratégies de réduction d'ordre doivent aborder le problème différemment. La première stratégie est d'ajouter des variables au problème. La première de ces méthodes, proposée par Rosenberg [1975], est de substituer un couple de variables par une variable additionnelle et une équation de contrainte qui garantit l'égalité entre les anciennes variables et la nouvelle. Les autres méthodes cherchent une fonction quadratique qui partage les mêmes minima que la fonction initiale. Une autre stratégie est d'identifier une configuration non-réalisable de la fonction, et enfin, une autre piste est de rechercher une relaxation quadratique de la fonction d'ordre supérieur.

3.4.1 Méthode par substitution et équation de contrainte

Réduire un terme en xy d'un ordre dans la posiforme P peut être réalisé à l'aide d'une fonction pseudo-booléenne $P_c(x, y, z)$ qui remplit les conditions :

$$P_c(x, y, z) = 0 \iff xy = z \quad (1.76)$$

$$P_c(x, y, z) \geq 0 \iff xy \neq z \quad (1.77)$$

Les produits de variables xy sont remplacés par une nouvelle variable z et la posiforme P_c est ajoutée avec un large coefficient pour forcer l'égalité entre les variables xy et z dans le minimum de la posiforme P ainsi modifiée. Boros et Hammer [2002] proposent la posiforme,

$$P_1(x, y, z) = xy - 2xz - 2yz + 3z. \quad (1.78)$$

L'un des termes quadratiques a un coefficient positif ainsi P_1 n'est pas sous-modulaire. Ainsi la réduction d'un ordre ne préserve pas la sous-modularité. D'autres fonctions P sont possibles, mais aucune n'est sous-modulaire. Dans la pratique, ajouter une fonction non sous-modulaire avec un large coefficient diminue les performances d'optimisation de $QPBO$ (pourcentage de la solution fixée par l'algorithme). Pour cette raisons d'autres techniques sont préférées.

3.4.2 Méthode de substitution par une fonction équivalente

La méthode consiste à remplacer un terme d'ordre trois par une FPB quadratique d'une variable additionnelle dont le minimum par rapport à cette nouvelle variable est égal au terme initial. Cette approche a été proposée par Kolmogorov et Zabih [2004], et Freedman et Drineas [2005].

$$ax_ix_jx_k = \min_y (aP_2(x_i, x_j, x_k, y)) \quad (1.79)$$

Le terme peut donc être remplacée par P_2 lors de la minimisation et conserver la solution sur les variables de départ. Dans la cas d'une fonction cubique et lorsque le coefficient devant un terme cubique est négatif une réduction possible est,

$$ax_ix_jx_k = \min_y ay(x_i + x_j + x_k - 2) \quad (1.80)$$

Dans le cas où a est positif, l'expression (1.80) se récrit en fonction de \bar{x} et en remplaçant \bar{x} par $1-x$,

$$ax_ix_jx_k = \min_y a(y(x_i + x_j + x_k - 1) + x_ix_j + x_jx_k + x_kx_i - x - y - z - 1) \quad (1.81)$$

A l'aide de ces deux formules, il est possible de réduire itérativement un terme cubique en termes quadratiques. Ces formules peuvent être généralisées à la réduction quadratique d'un terme d'ordre supérieur. Soit un terme d'ordres d , la forme linéaire S_1 et la forme quadratique S_2 :

$$\begin{aligned} S_1 &= \sum_{i=1}^d x_i \\ S_2 &= \sum_{i=1}^d \sum_{j=i+1}^d x_ix_j \end{aligned} \quad (1.82)$$

dans le cas où a est négatif, Freedman et Drineas [2005] proposent une fonction qui n'introduit pas de terme non sous-modulaire,

$$a \prod_{i=1}^d x_i = \min_y ay(S_1 - (d - 1)) \quad (1.83)$$

Dans le cas où a est positif, Ishikawa [2011] propose deux formules de réduction en fonction de la parité de d : quand d est pair

$$a \prod_{i=1}^d x_i = aS_2 + \min_y a \sum_{i=1}^{d/2-1} y_i(-2S_1 + 4i - 1) \quad (1.84)$$

et quand d est impair

$$a \prod_{i=1}^d x_i = aS_2 + \min_y a \sum_{i=1}^{(d-3)/2} y_i(-2S_1 + 4i - 1) + ay_{(d-1)/2}(-S_1 + d - 2) \quad (1.85)$$

Ces deux dernières réductions introduisent des termes non sous-modulaires (aS_2) et respectivement $d/2 - 1$ et $(d - 3)/2$ variables additionnelles.

3.4.3 Autres méthodes

Dans Ishikawa [2014] la réduction est réalisée sans ajout de variable, on cherche à identifier des configurations non réalisables de la solution. Il s'agit d'une configuration qui ne fait pas partie d'un minimum global. Il n'est donc plus nécessaire d'égaliser la fonction d'origine pour cette configuration. Ainsi (1.75) doit être respectée sauf dans une configuration. Avec une équation en moins, le système peut posséder une solution et permet de réduire un terme cubique.

Dans Fix *et al.* [2011] et Boros et Gruber [2014], les termes ne sont pas réduits un à un, mais par groupe. Gallagher *et al.* [2011] proposent de choisir entre les différentes formules de réduction et d'optimiser le problème du choix de la réduction selon un critère (minimiser la somme des termes non-sous-modulaires, ou l'amplitude de ces termes).

D'autres approches recherchent la "Generalized roof duality" (*GRD*) Kahl et Strandmark [2012]. Cela consiste à relaxer une fonction quelconque par une fonction quadratique sous-modulaire dont la borne inférieure (*RD*) est maximale. Pour une fonction quadratique, l'approche est équivalente à *QPBO*. Pour les ordres supérieurs, le problème devient difficile et des heuristiques ont été proposées pour approcher la *GRD*.

4 Schéma d'optimisation des problèmes multi-label

De nombreux problèmes en image sont écrits sous la forme d'une énergie multi-label. Ces labels sont des valeurs discrètes qui prennent des valeurs dans un intervalle. Nous considérons dans cette partie une énergie multi-label d'ordre 1 de la forme,

$$E(\mathbf{l}) = \sum_{p \in \mathcal{I}} D_p(l_p) + \sum_{(p,q) \in \mathcal{N}'} R_{pq}(l_p, l_q) \quad (1.86)$$

Nous avons vu qu'une telle énergie peut être optimisée globalement par coupure de graphe. Cependant, cette méthode demande de traiter un graphe volumineux et donc, est généralement longue à optimiser. De plus, cette méthode ne s'applique qu'à une classe réduite de fonctions de régularisation. Nous avons vu dans la Section 3.2 que la solution peut être approchée par un sous-problème : la fusion binaire. Elle consiste à limiter les variables du problème multi-labels à deux propositions de valeurs pour chaque pixel p : l_p^1 et l_p^2 . Lorsque le problème de fusion est sous-modulaire, la fusion peut être résolue exactement. Dans le cas général, des méthodes approchées

doivent être mis en œuvre, néanmoins il est possible de garantir la décroissance de l'énergie multi-labels (Section 4.1). Cette solution après fusion peut être utilisée comme une proposition pour une seconde fusion et approcher impérativement un minimum global de l'énergie multi-label : la solution courante \mathbf{l}^0 est initialisée et mise à jour selon un schéma d'optimisation qui consiste en un ensemble fini de fusions binaires. Différents schémas d'optimisation ont été proposés. Certains ne mettent en jeu que des fusions sous-modulaires qui peuvent donc être résolues par GC (Section 4.2), d'autres nécessitent la mise en œuvre des fusions approchées (Section 4.3).

4.1 Garantie de décroissance de l'énergie

Soit \mathbf{l}^0 une solution courante. A chaque fusion une nouvelle solution \mathbf{l}^* peut être construite. Une condition importante pour la convergence de la méthode est que l'énergie de la solution courante décroisse après chaque mise à jour. En effet, dans le cas où l'énergie est discrète et possède une borne minimale, la convergence de la solution est assurée. Une option est d'évaluer l'énergie $E(\mathbf{l}^*)$ et de remettre à jour la solution courante, si et seulement si

$$E(\mathbf{l}^*) \leq E(\mathbf{l}^0) \quad (1.87)$$

Le calcul de \mathbf{l}^* ayant un coût, il est préférable de construire des solutions qui assurent la décroissance de l'énergie. On distingue deux cas : le problème peut être résolu exactement (Section 4.1.1) ou une méthode approchées doit être mise en œuvre (Section 4.1.2).

4.1.1 Cas Sous-modulaire

Dans le cas où le problème de fusion est sous-modulaire, le minimum global du problème de fusion peut être trouvé par coupure de graphe avec la structure décrite dans la Section 3.2. Si \mathbf{l}^0 appartient à l'ensemble des fusion $\mathbf{l}_x^{1,2}$, c'est à dire qu'il existe une affectation de \mathbf{x}^0 telle que :

$$\mathbf{l}_{x^0}^{1,2} = \mathbf{l}^0 \quad (1.88)$$

La solution \mathbf{l}^* alors obtenue est nécessairement d'énergie inférieure à la solution courante :

$$\min_x E(\mathbf{l}_x^{1,2}) \leq E(\mathbf{l}^0) \quad (1.89)$$

c'est le cas lorsque l'on utilise \mathbf{l}^0 pour \mathbf{l}^1 lors de la fusion. Il est avantageux d'être dans le cas sous-modulaire car on obtient le minimum global du sous-problème ; la décroissance de l'énergie est alors rapide.

4.1.2 Fusions approchées

Soit $P(\mathbf{x})$ la FPB qui décrit l'énergie des solutions fusionnées, lorsque cette énergie n'est pas sous-modulaire, des méthodes d'optimisation approchées doivent être mises en œuvre. Une première approche est de limiter l'opération de fusion à un sous-ensemble de labels de sorte que le problème devienne sous-modulaire. La fusion devient alors une fusion partielle. Une deuxième approche est de relaxer le problème en ajoutant une FPB P' à valeurs positives afin de rendre l'énergie sous-modulaire. Soit \mathbf{x}^0 l'affectation qui donne le vecteur de labels de la solution courante \mathbf{l}^0 , si $P'(\mathbf{x}^0) = 0$ alors :

$$\min_x P(\mathbf{x}) \leq \min_x (P(\mathbf{x}) + P'(\mathbf{x})) \leq P(\mathbf{x}^0) \quad (1.90)$$

Il suffit donc de trouver la solution optimale du problème relaxé pour diminuer l'énergie. L'ajout de la fonction P' limite l'optimalité de la solution et donc l'efficacité de la méthode, mais cette approche peut être suffisante pour résoudre des énergies avec peu de termes non-sous-modulaires et de faible amplitude. La fonction P' qui modifie le moins l'énergie est celle qui annule les termes

quadratiques non-sous-modulaires de P . Cette méthode sera dénommée "fusion relaxée" dans la suite du manuscrit. La fusion partielle peut être vue comme un cas particulier où P' est une fonction qui interdit le changement de certaines variables.

Enfin la troisième approche consiste à appliquer la méthode $QPBO$. La solution partielle issue de l'optimisation garantit de diminuer l'énergie du fait de la propriété d'autarcie (Section 3.3.3). L'optimalité de la solution peut être améliorée à l'aide d'une des heuristiques proposées (Section 3.3.5). Cette approche est adaptée aux fonctions fortement non-sous-modulaires. Toutefois, elle conduit à des graphes plus volumineux (deux fois plus de variables) et est coûteuse en calculs pour une solution éventuellement peu complète. Il y a donc un enjeu à écrire, autant que possible, les problèmes sous la forme d'une fonction sous-modulaire pour une meilleure optimisation.

4.2 Schémas avec fusions binaires sous-modulaires

Pour certaines catégories de fonctions de régularisation, le problème de fusion peut être écrit comme une fonction sous-modulaire. Dans la Section 4.2.1, nous verrons le cas particulier des fonctions de régularisation convexes. D'autres types de fonctions de régularisation peuvent être utilisés dans des cas particuliers de fusion et conduire à des fusions sous-modulaires comme décrit dans les Section 4.2.2 et 4.2.3.

4.2.1 Fusions dans le cas convexe

Quel que soit le type de fusion utilisé, dans un cas où la régularisation est une fonction convexe de la différence des labels, un échange de variables particulier permet de rendre la fusion sous-modulaire. Plutôt que de prendre le problème de fusion $(\mathbf{l}^1, \mathbf{l}^2)$, on traite le problème de fusion $(\mathbf{l}^{min}, \mathbf{l}^{max})$ avec $\mathbf{l}^{min} = \mathbf{l}^1 \vee \mathbf{l}^2$ et $\mathbf{l}^{max} = \mathbf{l}^1 \wedge \mathbf{l}^2$, où \vee et \wedge sont le minimum et le maximum terme à terme. En effet nous allons montrer qu'il y a bien sous modularité. La condition de sous modularité est

$$R_{pq}(\mathbf{l}_i^{min}, \mathbf{l}_j^{min}) + R_{pq}(\mathbf{l}_i^{max}, \mathbf{l}_j^{max}) \leq R_{pq}(\mathbf{l}_i^{min}, \mathbf{l}_j^{max}) + R_{pq}(\mathbf{l}_i^{max}, \mathbf{l}_j^{min}) \quad (1.91)$$

Soit le vecteur de différence de labels $\mathbf{dl} = \mathbf{l}^{max} - \mathbf{l}^{min}$, ses composantes sont par définition positives. Si on considère que la régularisation est une fonction de la différence de labels k , alors la condition 1.91 se réécrit avec $k = \mathbf{l}_i^{min} - \mathbf{l}_j^{min}$:

$$r(k) + r(k + dl_i - dl_j) \leq r(k + dl_i) + r(k - dl_j) \quad (1.92)$$

Si la fonction r est convexe, on a $r(k)$ inférieur à la corde entre les dl_j et dl_i :

$$r(k) \leq \frac{dl_j}{dl_i + dl_j} r(k + dl_i) + \frac{dl_i}{dl_i + dl_j} r(k - dl_j) \quad (1.93)$$

et on a $r(k + dl_i - dl_j)$ inférieur à la corde entre les dl_j et dl_i :

$$r(k + dl_i - dl_j) \leq \frac{dl_i}{dl_i + dl_i} r(k + dl_j) + \frac{dl_j}{dl_i + dl_j} r(k - dl_j) \quad (1.94)$$

En sommant les deux inégalités 1.93 et 1.94, on retrouve l'inégalité 1.92. Cela permet de vérifier la condition de sous-modularité. Ainsi, toute fusion binaire avec une fonction de régularisation de la différence de labels convexe s'écrit sous la forme d'une FPB quadratique sous-modulaire et peut donc être minimisée par coupure de graphe.

4.2.2 α - β -Échange

Deux fusions, le ré-étiquetage et l'échange de labels, permettent d'optimiser une grande partie des fonctions de régularisation. La fusion par ré-étiquetage consiste à proposer, pour tous

Nom	l^1	l^2
Ré-étiquetage	l_i^0	$\begin{cases} \beta & \text{si } l_i^0 = \alpha \\ l_i^0 & \text{sinon} \end{cases}$
Échange	$\begin{cases} \alpha & \text{si } l_i^0 = \beta \\ l_i^0 & \text{sinon} \end{cases}$	$\begin{cases} \beta & \text{si } l_i^0 = \alpha \\ l_i^0 & \text{sinon} \end{cases}$
Expansion	l_i^0	α
Saut	l_i^0	$l_i^0 + \delta$

Tableau 1.6 – Proposition de label pour les différentes fusions

les labels α de la solution courante, le label β et le reste de la solution est inchangée. La fusion par échange de labels (Veksler [1999]) consiste à réaliser simultanément un ré-étiquetage α/β et β/α . Par changement de variables, l'échange d'étiquette peut s'écrire comme un ré-étiquetage : les labels α sont transférés dans une proposition de solution et les labels β dans l'autre. La condition de sous-modularité sur la fonction de régularisation pour ces deux fusions est,

$$R_{pq}(\alpha, \alpha) + R_{pq}(\beta, \beta) \leq R_{pq}(\beta, \alpha) + R_{pq}(\alpha, \beta) \quad (1.95)$$

Pour une fonction de régularisation r de la différence de labels, la condition se simplifie en :

$$2r(0) \leq r(\beta - \alpha) + r(\alpha - \beta) \quad (1.96)$$

Il suffit donc que la fonction r soit nulle en zéro et positive pour que le problème soit sous-modulaire. Cette condition est respectée par la plupart des fonctions de régularisation. Cela permet de mettre en œuvre l'échange de labels pour beaucoup d'énergies. Le schéma d'optimisation α - β -Échange consiste à réaliser successivement l'ensemble des échanges paramétrés par $(\alpha, \beta) \in L^2$. Le nombre de fusions binaires à réaliser peut être important du fait de la combinatoire. Comme seule une partie de l'image est remise en jeu, nous avons observé que les performances d'optimisation sont limitées.

4.2.3 α -Expansion

La fusion expansion (Veksler [1999]) consiste à proposer le label α pour tous les labels. La contrainte de sous-modularité sur la fonction de régularisation est alors :

$$R_{pq}(l_i, l_j) + R_{pq}(\alpha, \alpha) \leq R_{pq}(l_i, \alpha) + R_{pq}(\alpha, l_j) \quad (1.97)$$

Notons que si $R_{pq}(\alpha, \alpha) = 0$ alors R_{pq} est une distance. La condition de sous-modularité dans le cas où la régularisation est une fonction r de la différence de labels est :

$$r(l_i - l_j) + r(0) \leq r(l_i - \alpha) + r(\alpha - l_j) \quad (1.98)$$

Si $r(0) = 0$, on déduit que la fonction r vérifiant (1.98) est sous-additive. Une condition suffisante sur r est qu'elle soit concave et croissante sur \mathbb{R}^+ ainsi que concave et décroissante sur \mathbb{R}^- [Veksler, 1999]. L'avantage de la fusion par expansion est que la proposition de labels est indépendante de la solution courante et qu'elles respectent l'a priori fronto-parallèle pour la reconstruction stéréoscopique. Si la solution est constante par morceaux, ce sont des pans entiers de la solution qui sont proposés lors de la fusion. Le schéma d'optimisation α -Expansion consiste à réaliser successivement l'ensemble des expansions paramétrées par $\alpha \in L$. Ces fusions peuvent être prises dans un ordre croissant ou dans un ordre aléatoire. Cette méthode a montré des résultats intéressants pour de nombreux problèmes où l'approximation de la solution est bonne pour un temps de calcul raisonnable.

4.3 Schéma d'optimisation par fusion approchée

Il n'est pas toujours possible de trouver une écriture des fusions qui puisse être optimisée globalement. Ainsi, des schémas d'optimisation approchée ont été proposés. Ils ont été proposés dans le contexte de fonctions de régularisation particulières, convexes seulement autour de zéro (Sections 4.3.1 et 4.3.2). D'autres schémas ont été proposés notamment pour traiter les problèmes binaires obtenus à l'issue d'une réduction d'ordre et qui sont souvent fortement non-sous-modulaires.

Les deux premières méthodes utilisent une fusion particulière : la fusion par saut. Elle consiste à fusionner la solution courante avec la solution courante plus une constante δ sur chaque label. La condition de sous modularité sur R_{pq} est alors :

$$R_{pq}(l_i, l_j) + R_{pq}(l_i + \delta, l_j + \delta) \leq R_{pq}(l_i, l_j + \delta) + R_{pq}(l_i + \delta, l_j) \quad (1.99)$$

Le saut se rapproche des fusions du cas convexe (Section 4.2.1) par le fait que que les labels d'une des propositions sont supérieurs ou égaux aux labels correspondants dans l'autre proposition. Cette fusion explore l'énergie autour de la solution courante. L'énergie multi-label étant souvent non-convexe, elle possède de nombreux minima locaux. Ainsi la fusion par saut ne donne pas de bonnes approximations de la solution globale lorsque la solution courante en est très éloignée.

4.3.1 α -Expansion Quantifiée

Cette méthode introduite par Jeziarska *et al.* [2011] s'applique à des fonctions de régularisation convexes autour de zéro et constantes au delà d'un seuil T . Le schéma d'optimisation est composé de fusions par expansion partielle appelée "Fusion quantifiée" et de fusions par saut. Les "Fusions quantifiées" de paramètre $\alpha - T$ sont des expansions limitées aux pixels dont les labels sont en dehors de l'intervalle $[\alpha - T, \alpha + T]$. Cette fusion possède une écriture sous-modulaire. Les "Fusions quantifiées" sont échantillonnées selon T . Cela permet de diminuer le nombre de fusions du schéma et ainsi d'être plus rapide. Les valeurs précises de la solution sont obtenues en appliquant des fusions par saut dont le paramètre est dans l'intervalle $[-T, T]$. La fonction de régularisation n'étant pas convexe, le problème de saut n'est pas sous-modulaire. Les auteurs choisissent de résoudre le problème avec une approximation convexe de la fonction de régularisation et de rejeter cette solution si elle possède une énergie supérieure à l'énergie de la solution courante initiale.

4.3.2 Expansion-Saut alternés

Dans Paget *et al.* [2015], nous proposons d'alterner des fusions par expansion suivies de fusions par saut à plusieurs pas. L'objectif est de pouvoir traiter des fonctions convexes autour de zéros et concaves pour des grandes valeurs. Ces deux fusions sont complémentaires : la première est sous-modulaire dans la partie concave de la fonction, et la seconde dans la partie convexe. Initialement dans l'article, les non-sous-modularités sont gérées en réalisant des fusions partielles. Dans le manuscrit, nous utilisons des fusions relaxées car elles impactent moins l'énergie de la fusion. Les deux schémas α -Expansion Quantifiée et Expansion-Saut alternés seront mis en œuvre et comparés sur les problèmes de dé-bruitage et de reconstruction stéréoscopique dans le chapitre suivant.

4.3.3 Autres schémas

LogCut Les labels peuvent être écrits en base 2 de sorte que chaque label est écrit comme une chaîne de 0 et de 1. Le problème est d'optimiser les valeurs de bits de la solution. Le nombre de variables par label est en \ln_2 du nombre de valeurs de label. C'est donc une écriture plus

compacte que le problème décrit dans la Section 3.1. Toutefois, l'ordre de la fonction pseudo-booléenne peut être très élevé : même en considérant un problème sans régularisation, l'ordre de la FPB est aussi en \ln_2 du nombre de valeurs de label. Dans [Lempitsky *et al.*, 2010], un bit par label par label est remis en jeu à la fois pour limiter l'ordre du problème. Ils proposent d'ajouter une constante aux labels avant de la convertir en base 2 afin d'explorer d'autres valeurs de label. De plus, les auteurs proposent de réaliser les fusions en parallèle plutôt que successivement. n branches sont traitées en parallèle réalisant chacune un sous-ensemble des fusions par expansions. Les résultats des n branches sont fusionnés deux à deux. La sous-modularité du problème de fusion des branches n'est pas garantie. Cela nécessite la mise en œuvre des méthodes approchées.

Bruit-Flou Dans [Ishikawa, 2011], deux de deux fusions sont proposées : la fusion par bruitage et la fusion par floutage. La fusion par bruitage permet d'explorer de manière aléatoire l'espace des solutions. L'inconvénient de cette fusion est que la proposition faite possède une énergie de régularisation élevée du fait de l'absence de structure. Ainsi les solutions fusionnées ont tendance elles aussi à avoir des valeurs élevées. La fusion par floutage permet de proposer une version plus lisse que la solution courante. Le schéma d'optimisation alterne des fusions par floutage et par bruitage. Ce schéma est utilisé pour minimiser des fonctions issues d'une réduction d'ordre qui possèdent souvent des termes non-sous-modulaires de fortes amplitudes.

5 Conclusion

La formulation par coupure de graphe permet de minimiser en temps polynomial une énergie binaire sous-modulaire. Certains problèmes image peuvent être écrits sous cette forme comme les problèmes multi-labels avec une régularisation convexe. Cependant, dans les autres cas, le problème doit être approché en le décomposant en sous-problèmes binaires, ce qui fait perdre les garanties d'optimalité sur la solution. Ces problèmes ne sont pas nécessairement sous-modulaires ce qui impose d'utiliser des méthodes approchées telles que *QPBO*. Par ailleurs, il est possible de réécrire des fonctions d'ordre élevé à l'aide d'une fonction quadratique. Néanmoins, l'opération de réduction introduit souvent des termes non-sous-modulaires ce qui réduit l'efficacité de l'optimisation en pratique. Différents schémas d'optimisation ont été proposés pour approcher la solution globale, adaptés à certaines fonctions de régularisation ou à certains types de problèmes. En résumé, les méthodes par coupure de graphe peuvent être étendues pour résoudre de manière approchée des problèmes multi-labels d'ordre élevé et non-sous-modulaires par décomposition en sous-problèmes binaires, réduction d'ordre et relaxation sous-modulaire sans garantie sur les performances d'optimisation tant en termes de temps de calculs qu'en termes d'optimalité de la solution.

Chapitre 2

Coupure de graphe appliquée aux problèmes image

La formulation par coupure de graphe (*Graph Cut*, GC) consiste à écrire la solution d'un problème comme le minimum d'une énergie binaire sous-modulaire. Dans ce chapitre, nous appliquerons les GC à deux problèmes image : le dé-bruitage et la reconstruction stéréoscopique. Dans la Section 1, nous présentons le problème de dé-bruitage. Le traitement de ce problème sera l'occasion d'observer l'influence de la régularisation sur la solution et de comparer les méthodes d'optimisation. Dans la Section 2, nous étudierons la modélisation du problème de reconstruction par coupures de graphe. Les résultats de reconstruction selon les différentes méthodes d'optimisation sont présentés dans la Section 3.

1 Formulation par coupure de graphe appliquée au dé-bruitage

Le dé-bruitage est un cas particulier de la restauration d'image. Il consiste à estimer, à partir d'une image bruitée, une image où le bruit est atténué, voire entièrement retiré. Les premières approches ont consisté en un lissage de la solution ne préservant pas les contours, comme le filtrage gaussien. Le problème du flou sur les contours peut être éliminé par filtrage non-linéaire, comme le filtre médian. Toutefois, la nécessité d'un traitement différencié des zones homogènes et des contours est rapidement apparu, comme par exemple dans le filtrage bilatéral [Tomasi et Manduchi, 1998]. Parallèlement, une autre approche consiste à opérer le dé-bruitage en considérant des patch, comme dans les moyennes non-locales [Buades *et al.*, 2005]. Ce type de méthode a connu de grandes avancées et s'est beaucoup développé grâce aux méthodes par apprentissage et notamment l'apprentissage profond [Vincent *et al.*, 2010]. Dans cette section, nous étudierons une manière simple de modéliser le problème du dé-bruitage (Section 1.1). La résolution du problème sera l'occasion d'illustrer l'effet de la régularisation sur la solution (Section 1.2) puis de comparer les différentes méthodes de résolution par GC : tout d'abord dans le cas où une méthode exacte existe (Section 1.3), ensuite dans un cas où la mise en œuvre d'une méthode approchée est nécessaire (Section 1.4).

1.1 Modélisation du problème

L'objectif du dé-bruitage est d'estimer l'image sans bruit $\hat{\mathbf{l}}$ en supposant que l'image observée \mathbf{l}^b est l'image originale $\hat{\mathbf{l}}$ soumise à un bruit \mathbf{B} . En faisant l'hypothèse que le bruit est additif, $\mathbf{l}^b = \hat{\mathbf{l}} + \mathbf{B}$, $\mathbf{B} \sim \mathbb{P}_b(\mathbf{B})$. La vraisemblance de \mathbf{l}^b sachant $\hat{\mathbf{l}}$ est égale à la distribution du bruit, $\mathbb{P}(\mathbf{l}^b | \hat{\mathbf{l}}) = \mathbb{P}_b(\mathbf{B})$. L'image originale n'étant pas connue, nous devons faire une hypothèse supplémentaire pour pouvoir évaluer cette vraisemblance : la probabilité du bruit est substituée par celle des résidus $\mathbb{P}(\mathbf{l}^b | \hat{\mathbf{l}}) = \mathbb{P}_b(\mathbf{l}^b - \hat{\mathbf{l}})$. Dans le cas d'un bruit centré, la solution obtenue en appliquant le principe du maximum de vraisemblance (*Maximum Likelihood*, ML) est égale à l'entrée, c'est-à-dire qu'aucun traitement n'a été réalisé sur l'image. Il est donc nécessaire d'introduire un a priori



Figure 2.1 – La base de test, pour le dé-bruitage, est composée de l'image Lena et des quatre images ci-dessus. De gauche à droite : Pepper, Barbara, Boat, House.



Figure 2.2 – Image originale et image bruitée (bruit gaussien indépendant d'écart-type $\sigma = 20$ et agrandissements centrés sur une zone d'intérêt

sur la solution pour espérer séparer le bruit du signal. L'a priori sur la solution doit correspondre aux caractéristiques recherchées de la solution. L'a priori utilisé est que deux pixels voisins dans l'image ont une valeur proche. Le plus souvent, les hypothèses faites sont que l'écart des valeurs entre deux pixels voisins suit une loi $l_i - l_j \sim \mathbb{P}_\epsilon$. Cette loi est la même sur toute l'image et le bruit sur chaque pixel est indépendant d'un pixel à l'autre. A partir de ce modèle, en appliquant le principe du maximum a posteriori qui implique de maximiser $\mathbb{P}(\mathbf{l}|\mathbf{l}^b)$ et après application de la règle de Bayes, le $-\log$ de cette probabilité est de la forme :

$$E(\mathbf{l}) = \sum_{i \in \mathbf{I}} f(l_i^b - l_i) + \lambda \sum_{(i,j) \in \mathbf{N}'} r(l_i - l_j) \quad (2.1)$$

où f est le $-\log$ de la distribution du bruit \mathbf{B} et r le $-\log$ de la distribution de l'écart \mathbb{P}_ϵ de valeurs entre des pixels voisins. Le paramètre λ dans cette énergie est le rapport d'échelle des lois de probabilité. Il peut être considéré comme un paramètre additionnel qui permet de régler l'intensité de la régularisation. Dans la pratique, il est nécessaire de régler ce paramètre pour arriver au compromis désiré entre les deux termes de (2.1). Ces fonctions peuvent être définies à une constante près sans modifier la solution du problème. Nous prendrons $f(0) = 0$ et $r(0) = 0$.

Dans cette section, nous utiliserons des images test fréquentes dans la littérature. Il s'agit des images en niveau de gris Lena, Pepper, Barbara, Boat et House montrées dans les Figure 2.1 et 2.2. La Figure 2.2 représente l'image Lena originale et l'image bruitée, ainsi qu'un agrandissement sur une zone d'intérêt.

1.2 Comparaison des fonctions de régularisation

Le choix de la fonction de régularisation a un impact fort sur l'aspect de la solution obtenue. Nous étudierons dans un premier temps les résultats issus d'une régularisation convexe (Section 1.2.1). Nous illustrerons l'effet de certaines caractéristiques de la fonction de régularisation sur le résultat (Section 1.2.2). Enfin, nous proposerons une méthode pour rendre comparables des solutions obtenues avec différentes fonctions de régularisation (Section 1.2.3).



Figure 2.3 – Résultats de dé-bruitage de l’image Lena sur la première ligne, et un agrandissement sur la deuxième ligne, pour trois fonctions de régularisation : r la fonction carré (première colonne), r la valeur absolue (deuxième colonne) et r la valeur absolue lissée (troisième colonne). Nous remarquons que la fonction carré donne des solutions floues et que valeur absolue donne un effet en marche d’escalier sur la joue de Lena. Ces effets ne sont plus visibles dans le cas de la valeur absolue lissée.

1.2.1 Cas où la fonction r est convexe

Un des premiers modèles en dé-bruitage est le modèle de Tikhonov où les fonctions f et r sont, toutes deux, des fonctions carré. L’énergie est quadratique, et il est possible de l’optimiser par une méthode explicite. Le problème d’une telle énergie est que la solution est une version floutée de l’entrée : elle ne conserve donc pas les contours. C’est pour cela que la régularisation par la norme $L1$ a été ensuite introduite. Elle est équivalente à prendre la fonction r égale à la valeur absolue. Ce modèle possède des propriétés de parcimonie : les solutions minimales ont tendance à être constantes par morceaux et par conséquent les contours sont préservés. L’inconvénient est que ce modèle s’adapte mal à des variations lentes de l’intensité qui sont fréquentes dans les images. Il en résulte un effet en marche d’escalier dans ces zones (voir Figure 2.3). Cet effet peut être corrigé en introduisant une version lissée sur un pixel de la fonction valeur absolue qui possède une cuvette convexe autour de zéro :

$$\begin{aligned} r(x) &= a|x|, & \text{si } |x| \leq 1 \\ r(x) &= b|x| + a - b, & \text{sinon.} \end{aligned} \quad (2.2)$$

avec $a \leq b$. La Figure 2.3 présente les résultats pour ces trois fonctions de régularisation sur l’image Lena. Dans le cas de la fonction carré, nous observons bien que la solution est floue et que les contours ne sont pas préservés. Dans le cas de la valeur absolue, les contours sont préservés (sur le nez et l’œil par exemple) mais un effet en marche d’escalier apparaît dans les zones de dégradé lent (la joue par exemple). La solution avec la valeur absolue lissée a ses contours préservés et des variations progressives de l’intensité dans les zones de dégradé.

Les énergies où r est convexe peuvent donner des solutions satisfaisantes cependant certaines propriétés recherchées pour la régularisation, comme la sous-additivité sur \mathbb{R}^+ , ne peuvent pas être obtenues dans la familles des fonctions strictement convexes.

1.2.2 Sous-additivité et sur-additivité des fonctions de régularisation

La solution du dé-bruitage est formulée comme un compromis entre deux critères : la proximité à l'image bruitée et la régularité de la solution. Or, le plus souvent, la solution ne peut respecter simultanément les deux critères et différents écarts peuvent être observés : l'écart de la solution à la référence en un pixel, l'écart de valeur de label entre deux pixels voisins. Un coût dans l'énergie est associé à chacun de ces écarts en appliquant les fonctions f et r . Nous rappelons que $r(0) = 0$. Deux propriétés ont un impact important sur la forme des solutions obtenues : la sous-additivité et la sur-additivité. La sous-additivité de la fonction r s'écrit :

$$r(x + y) \leq r(x) + r(y) \quad (2.3)$$

de même la super-additivité de la fonction r s'écrit :

$$r(x) + r(y) \leq r(x + y) \quad (2.4)$$

où x et y sont des valeurs d'écarts. Pour des écarts de signe opposé, la fonction r respecte la propriété de sous-additivité. Cela signifie que compenser des écarts de signe opposé diminue le coût de régularisation, ce qui est un des effets recherchés par la régularisation. Si les écarts sont de même signe, on distingue deux cas :

- Si r est concave sur \mathbb{R}^+ , alors r est sous-additif. Ainsi regrouper les écarts permet de diminuer le coût de régularisation.
- r est convexe sur \mathbb{R}^+ , alors r est sur-additif. Ainsi regrouper les écarts permet de diminuer le coût de régularisation.

En présence d'un contour entre deux plateaux à des valeurs différentes, la transition se fera de manière différente en fonction de la forme de r . Dans le cas d'une fonction convexe, la transition sera progressive et donc sera assez lisse. Avec une fonction concave, une transition rapide sera privilégiée. Cela explique pourquoi les fonctions concaves sur \mathbb{R}^+ conservent les contours. Inversement les dégradés progressifs sont bien reconstruits avec des fonctions convexes mais un effet en marche d'escalier peut apparaître avec les fonctions concaves. La fonction valeur absolue est un cas limite, car elle est à la fois concave et convexe sur \mathbb{R}^+ : les relations (2.3) et (2.4) deviennent des égalités sur \mathbb{R}^+ . Il existe donc de nombreuses configurations avec un coût équivalent. En pratique, les méthodes d'optimisation ont tendance à sélectionner des solutions parcimonieuses parmi l'ensemble de ces solutions de sorte qu'elles peuvent présenter des marches d'escalier.

Une solution pour éviter l'effet en marches d'escalier, tout en conservant les contours, est d'utiliser une fonction convexe autour de zéro, et donc relativement lisse, et concave pour des plus grandes valeurs. On trouve dans cette catégorie le carré tronqué, la valeur absolue lissée et la famille des exponentielles lissées (*Smooth Exponential Family*, SEF, [Caraffa et al., 2015]) pour des valeurs de a inférieures à 1. Les SEF sont une famille de fonctions paramétrées par la forme a et l'échelle s :

$$SEF_{a,s}(x) = \frac{(1 + (x/s)^2)^{a/2} - 1}{a/2} \quad (2.5)$$

Pour $a = 2$, *SEF* est une fonction carrée. Ces fonctions ont la particularité d'être proches de la fonction carré autour de zéro et proches d'une puissance pour de grandes valeurs. Pour des valeurs de forme inférieures à 1, la fonction est concave pour des grandes valeurs. L'intérêt de telles fonctions est illustré dans les Figures 2.4 et 2.5. Elles permettent d'obtenir des solutions qui présentent des dégradés progressifs ainsi que des contours nets.

1.2.3 Réglage du paramètre de régularisation

Nous souhaitons comparer l'effet de la forme de la fonction de régularisation sur la solution. Changer la forme de la fonction r modifie globalement l'intensité de la régularisation comme le fait le paramètre λ . Il est alors difficile de différencier les changements dus à la sur- ou sous-régularisation, de ceux dus à la forme elle-même de la fonction. Il peut donc être nécessaire

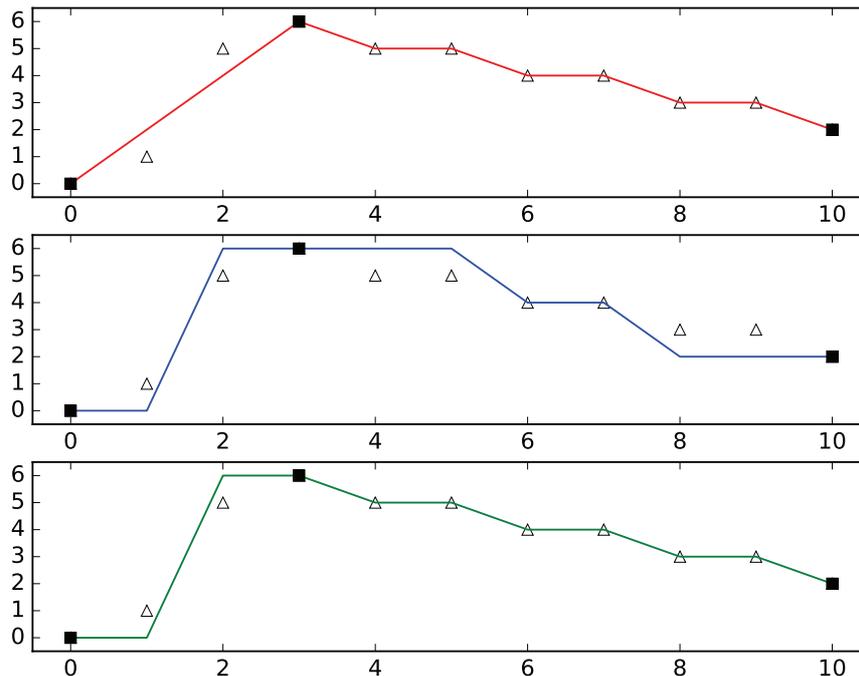


Figure 2.4 – Comparaison des résultats de dé-bruitage du signal pour trois modèles. Certaines valeurs dans la solution sont fixées. Les points correspondant dans le signal sont représentés par des carrés noirs. Les autres points du signal, représentés par des triangles blancs, sont laissés libres dans la solution. La fonction d'attache aux données est $f(x) = x^2$ pour les trois modèles. La fonction de régularisation du modèle est, de haut en bas, $r_1(x) = 2x^2$, $r_2(x) = 4\sqrt{|x|}$ et $r_3 = 10SEF_{-2,2}(x)$. Ces fonctions sont représentées dans la Figure 2.5. Avec le premier modèle, en haut, la fonction convexe donne une solution très lisse où le contour entre $x = 0$ et $x = 2$ est lissé. Avec deuxième modèle, au milieu, la fonction concave donne une solution constante par morceaux avec des ruptures nettes. Les contours sont préservés mais l'effet de marche d'escalier apparaît dans les zones de dégradé entre $x = 4$ et $x = 10$. Avec le dernier modèle, en bas, la forme de la fonction de régularisation permet à la fois de préserver les contours et de suivre les dégradés progressifs.

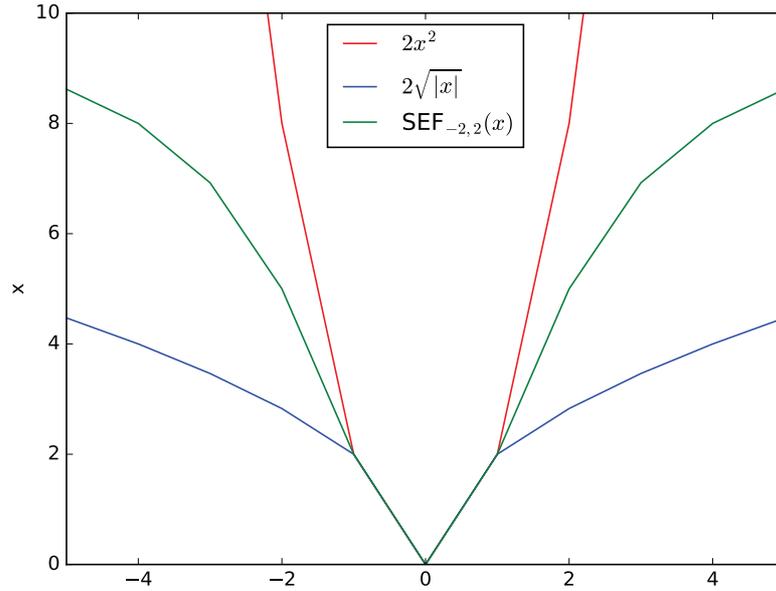


Figure 2.5 – Fonctions de régularisation r utilisées dans l'exemple de la Figure 2.4

d'adapter automatiquement le paramètre λ pour que les régularisations soient d'intensité comparable, ce qui permet une comparaison plus aisée des solutions. Une possibilité est d'optimiser le paramètre λ en fonction d'un critère d'écart à une référence (par exemple le PSNR). D'une part, cette méthode nécessite d'optimiser un nombre important d'énergie et d'autre part, une référence doit être disponible. Enfin, cette approche est discutable car, suivant le critère de comparaison, cela ne permet pas forcément d'atteindre l'objectif : il n'est pas garanti que les deux solutions aient un aspect sur- ou sous-régularisé équivalent. Nous proposons donc une méthode pour rendre comparable les effets de deux modèles, connaissant la solution obtenue par de l'un de ces modèles. Soit E_1 et E_2 deux énergies formulées avec la même fonction d'attache aux données et des fonctions de régularisation différentes, r_1 et r_2 :

$$\begin{aligned} E_1(\mathbf{l}) &= \sum_{i \in \mathbf{I}} f(l_i^b - l_i) + \lambda_1 \sum_{(i,j) \in \mathbf{N}'} r_1(l_i - l_j) \\ E_2(\mathbf{l}) &= \sum_{i \in \mathbf{I}} f(l_i^b - l_i) + \lambda_2 \sum_{(i,j) \in \mathbf{N}'} r_2(l_i - l_j) \end{aligned} \quad (2.6)$$

La solution du modèle E_1 est connue, et nous souhaitons déterminer la valeur λ_2 qui donne la même intensité de régularisation. Nous définissons cette intensité comme étant la somme des termes r de régularisation sur la solution. Soit h_1 et h_2 les histogrammes des écarts entre voisins dans les solutions des deux modèles. Nous proposons alors de fixer λ_2 selon :

$$\lambda_2 = \lambda_1 \frac{\sum_i h_1(i) r_1(i)}{\sum_i h_2(i) r_2(i)} \quad (2.7)$$

Cependant nous ne disposons pas de l'histogramme h_2 . Nous considérons alors l'histogramme de la première solution comme une approximation de h_2 , ce qui conduit à :

$$\hat{\lambda}_2 = \lambda_1 \frac{\sum_i h_1(i) r_1(i)}{\sum_i h_1(i) r_2(i)} \quad (2.8)$$

Par construction, $\hat{\lambda}_2$ sous-estime la valeur de λ_2 . En utilisant (2.8), si les fonctions r_1 et r_2 ne sont pas trop éloignées, cette méthode donne des résultats satisfaisants dans nos tests, comme illustrés dans la Figure 2.6, sur l'image Lena.



Figure 2.6 – Résultats de remise à l'échelle de la régularisation en utilisant (2.8). Sur la première ligne les solutions où r est la valeur absolue avec les paramètres $\lambda = 20$, $\lambda = 50$, $\lambda = 100$, respectivement, et sur la deuxième ligne, les résultats avec régularisation avec la fonction $SEF_{0.5,4}$. Les paramètres calculés sont $\hat{\lambda}_2 = 83$, $\hat{\lambda}_2 = 212$, $\hat{\lambda}_2 = 430$. Visuellement l'intensité de la régularisation paraît similaire dans les trois cas entre le haut et le bas.

1.3 Comparaison entre les schémas d'optimisation approchés et la résolution exacte

Nous souhaitons comparer la solution approchée obtenue par α -Expansion avec la solution minimale quand cela est possible. Le minimum global peut être obtenu par la méthode proposée dans [Ishikawa, 2003]. Bien sûr, il existe des contraintes sur la fonction de régularisation r pour être optimisée par ces deux méthodes. Nous utiliserons donc seulement la fonction valeur absolue et la fonction valeur absolue lissée. Pour cette dernière, il est nécessaire d'opérer une réécriture sous-modulaire de la fusion par expansion, ce qui est possible lorsque r est convexe (cf Section 4.2.1 Chapitre 1).

1.3.1 Cas où r est la fonction valeur absolue

Nous nous plaçons dans le cas où r est la fonction valeur absolue. Cela correspond à l'a priori que l'écart entre des intensités de pixels voisins suit une distribution de Laplace. Le choix de cette loi est motivé par les contraintes des méthodes d'optimisation et ce n'est pas forcément la loi qui modélise le mieux le problème. Les expérimentations sont réalisées pour trois bruits indépendants et identiquement distribués : un bruit gaussien d'écart-type $\sigma = 20$, un bruit laplacien d'échelle $s = 20$ et un bruit de Cauchy d'échelle $s = 20$. Les valeurs en dehors de l'intervalle des données saturent aux valeurs extrêmes. La fonction de régularisation est $r(x) = |x|$ dans les trois cas. Les fonctions d'attache aux données pour les trois types de bruit sont $f(x) = x^2$, $f(x) = |x|$, $f(x) = \ln(1 + \frac{x^2}{s^2})$ et les facteurs de régularisation sont $\lambda = 0.8$, $\lambda = 0.8$, $\lambda = 0.04$.

Les résultats pour les bruits gaussiens et laplaciens sur les cinq images de la base de test sont donnés dans le Tableau 2.1. En pratique α -Expansion converge en deux itérations, la deuxième ne servant qu'à constater que la solution a convergé : l'approximation faite par α -Expansion est suffisante pour atteindre le minimum global. La solution finale est donc optimale, pour un temps d'exécution qui est environs 5 fois inférieur à celui avec la méthode d'Ishikawa. Les résultats pour

Image	Gauss $\sigma = 20$				Laplace $s = 20$			
	Ishikawa		α -Expansion		Ishikawa		α -Expansion	
	énergie	temps	énergie	temps	énergie	temps	énergie	temps
Pepper	13 544 952	55,4 s	+0	10,3 s	12 549 064	52,4 s	+0	9,40 s
Barbara	56 634 892	241 s	+0	40,5 s	48 279 928	231 s	+0	40,5 s
Boat	51 704 102	200 s	+0	39,6 s	51 704 102	186 s	+0	36,5 s
House	12 310 902	47,4 s	+0	9,10 s	11 195 042	42,2 s	+0	8,2 s
Lena	48 999 864	207 s	+0	38,2 s	44 820 374	189 s	+0	35,6 s

Tableau 2.1 – Comparaison entre α -Expansion et Ishikawa de l'énergie finale et du temps d'exécution pour un bruit gaussien et un bruit laplacien. Dans cette expérimentation, on a $\lambda = 0.8$ et les énergies sont données à un facteur 10. On remarque que la solution obtenue par α -Expansion est une solution minimale du problème.

Image	Cauchy $s = 20$			
	Ishikawa		α -Expansion	
	énergie	temps	énergie	temps
Pepper	6 447 042	69,5 s	+5	18,4 s
Barbara	25 462 129	276 s	+19	76,6 s
Boat	24 556 292	225 s	+3	70,6 s
House	5 926 775	50,5 s	+5	15,7 s
Lena	23 585 990	234 s	+29	68,2 s

Tableau 2.2 – Comparaison entre α -Expansion et Ishikawa de l'énergie finale et du temps d'exécution pour un bruit de Cauchy. Dans cette expérimentation, on a $\lambda = 0.04$ et les énergies sont données à un facteur 100. La solution α -Expansion n'est pas minimale en étant toutefois de valeurs très proches de celles du minimum de l'énergie. En pratique, les solutions sont elles-aussi très proches.

le bruit de Cauchy sont donnés dans le Tableau 2.2. α -Expansion converge en trois itérations mais n'atteint pas le minimum global de l'énergie. La valeur du minimum reste toutefois très proche (écart entre 3 et 29). A la différence des deux cas précédents, le troisième modèle n'est pas convexe. En conséquence, il apparaît qu' α -Expansion est susceptible d'atteindre rapidement le minimum global quand le modèle est convexe, mais que ses performances sont réduites pour des attaches aux données non-convexes.

1.3.2 Cas où r est la fonction valeur absolue lissée

Nous nous plaçons maintenant dans le cas d'un bruit gaussien et l'attache aux données f la fonction carré. La fonction de régularisation r est désormais la valeur absolue lissée. En principe la fusion expansion ne peut pas être mise en œuvre avec cette fonction de régularisation. Cependant, comme la fonction est convexe, chaque sous-problème de fusion peut être optimisé globalement moyennant un changement de variable (voir la Section 4.2.1 du Chapitre 1). Nous comparons donc trois approches : celle d'Ishikawa, α -expansion selon différentes variantes, et le schéma par fusion saut +1 et -1. Le graphe d'Ishikawa étant très volumineux, nous faisons les expérimentations sur des sous-images de taille 128 par 128 pixels.

Le schéma α -Expansion a été proposé dans le cas d'une fonction de régularisation métrique (ie. une distance). Or, la fonction de régularisation utilisée est non-métrique. Il en résulte des sous-problèmes binaires pas nécessairement sous-modulaires. Quatre variantes sont utilisées pour résoudre ces sous-problèmes : la relaxation, QPBO, QBPO-P et la réécriture en fusion convexe. Dans le Tableau 2.3, il apparaît que la convergence de la fusion relaxée est deux fois plus rapide que QPBO, lui-même deux fois plus rapide que QPBO-P. Les minima de QPBO et QPBO-P sont égaux et inférieurs au minimum de la fusion relaxée, et sont assez proches du minimum obtenu

Méthode	Pepper		Barbara		Boat	
	Énergie	temps	Énergie	temps	Énergie	temps
Ishiwaka	14 336 480	335 s	13 017 181	347 s	18 077 919	338 s
α-Exp	+251 486	10,6 s	+202 214	11,7 s	+179 278	10,5 s
α-Exp QPBO	+134 795	23,3 s	+126 744	20,6 s	+112 458	22,0 s
α-Exp QPBO-P	+134 795	42,7 s	+126 744	38,1 s	+112 458	38,9 s
α-Exp Convexe	+133 659	16,3 s	+123 607	12,4 s	+104 454	13,5 s
Saut +1/ - 1	+0	2,75 s	+0	2,17 s	+0	2,80 s

Méthode	House		Lena	
	Énergie	temps	Énergie	temps
Ishiwaka	14 321 340	263 s	12 426 375	278 s
α-Exp relaxé	+151 920	10,6 s	+265 218	11,7 s
α-Exp QPBO	+94 446	23,3 s	+158 157	20,6 s
α-Exp QPBO-P	+94 446	42,7 s	+158 157	38,1 s
α-Exp Convexe	+91 732	11,9 s	+154 346	15,4 s
Saut +1/ - 1	+0	2,53 s	+0	2,08 s

Tableau 2.3 – Comparaison des énergies et temps de calcul dans le cas où r est la fonction valeur absolue lissée pour différentes méthodes d’optimisation. Les valeurs d’énergie des schémas sont données en écart à l’énergie minimale. Les autres méthodes sont α -expansion optimisé par fusion relaxée, par QPBO, par QPBO-P, reformulé en fusion convexe et enfin le schéma Saut composé de deux fusions : le saut +1 et le saut -1.

par résolution exacte de chaque sous-problème de fusion (fusion dans le cas convexe). Toutefois, la résolution exacte de chaque sous-problème de fusion ne permet pas d’atteindre le minimum global. Le schéma par saut converge vers le minimum global. Cela s’explique par le fait que ce schéma d’optimisation s’apparente à une descente de gradient et que dans le cas d’une énergie convexe, la descente de gradient converge vers le minimum global. Le schéma Saut est aussi le plus rapide : son temps de calcul est environ un quart de la variante d’ α -Expansion la plus rapide. Il ressort de ces résultats que s’il est possible de traiter des régularisations convexes avec α -Expansion, ce schéma d’optimisation est moins adapté que le schéma par saut pour minimiser l’énergie lorsque r est la fonction valeur absolue lissée.

1.4 Comparaison des performances des différents schémas d’optimisation

Dans cette section la fonction de régularisation est obtenue en observant l’histogramme des écarts d’intensité entre voisins. La forme de la fonction r est déterminée par le $-\log$ de l’histogramme.

$$E(\mathbf{l}) = \sum_{i \in \mathbf{I}} (l_i^b - l_i)^2 + \lambda \sum_{(i,j) \in \mathbf{N}'} SEF_{a,s}(l_i - l_j) \quad (2.9)$$

où $SEF_{a,s}$ est une fonction exponentielle lissée (2.5). Ces fonctions ont la particularité d’approcher une fonction quadratique autour de zéro et une fonction puissance pour les grandes valeurs. Elle correspond au $-\log$ d’une distribution proche d’une gaussienne autour de zéro et d’une autre loi pour les grandes valeurs. Quand $a = 2$, elle correspond à une distribution gaussienne ; quand $a = 1$ à une distribution de Laplace lissée et quand $a \rightarrow 0$, à une distribution de Cauchy. Pour des valeurs de a inférieures à 1, la fonction est ni convexe et ni concave sur \mathbb{R}^+ . La méthode d’Ishikawa ne peut plus être appliquée et les sous-problème de fusion sont donc non-sous-modulaires : des méthodes approchées des fusions doivent être mises en œuvre. Dans cette section nous prendrons $f(x) = x^2$, $g(x) = SEF_{0,5,4}$ et $\lambda = 100$. Nous prenons en compte quatre schémas d’optimisation. Soit $[-T, T]$ l’intervalle où la régularisation est convexe ; les quatre schémas sont :

- α -Expansion,
- Alterné-C : schéma alterné complet avec toutes des expansions et tous les sauts,

- Alterné-L : schéma alterné avec toutes des expansions et les sauts dont le paramètre est limité à l'intervalle $[-T, T]$,
- Alterné-LQ : schéma alterné avec les expansions à intervalle T et les sauts dont le paramètre est limité à l'intervalle $[-T, T]$.

Il s'agit d' α -Expansion et de trois propositions de schémas d'optimisation. Le schéma Alterné-C est présenté dans notre article [Paget *et al.*, 2015]. Le schéma Alterné-LQ est inspirée de [Jeziarska *et al.*, 2011] et adapté aux énergies que nous souhaitons optimiser.

1.4.1 Comparaison entre les schémas d'optimisation

	Fusion relaxée		QPBO		QPBO-P	
	ΔE	τt	ΔE	τt	ΔE	τt
Image	α -Expansion					
Pepper	783 004	20%	184 831	50%	184 501	88%
Barbara	2 779 150	12%	709 860	36%	709 302	62%
Boat	2 938 868	21%	743 708	47%	743 499	81%
House	742 441	22%	194 914	56%	194 884	95%
Lena	3 133 824	15%	764 929	45%	764 497	75%
Image	Alterné-C					
Pepper	1111	39%	6	59%	0	100%
Barbara	5602	36%	40	50%	0	100%
Boat	4025	42%	128	60%	0	100%
House	606	49%	-34	58%	0	100%
Lena	3662	30%	68	59%	0	100%
Image	Alterné-L					
Pepper	1177	14%	205	22%	177	37%
Barbara	5815	69%	224	164%	297	236%
Boat	4190	78%	192	103%	126	153%
House	659	15%	-21	18%	1	31%
Lena	3761	62%	160	89%	97	171%
Image	Alterné-LQ					
Pepper	1183	7%	225	10 %	217	17%
Barbara	5393	25%	-232	50%	-245	85%
Boat	3598	25%	323	37%	310	62%
House	693	5%	88	8 %	83	14%
Lena	3586	26%	85	36%	67	61%

Tableau 2.4 – Comparaison entre les schémas d'optimisation α -Expansion et Alterné-C pour trois méthodes d'optimisation. Les valeurs d'énergie sont données en écart à l'énergie QPBO-P pour le schéma Alterné-C, les temps d'exécution sont donnés en pourcentage du temps d'exécution de QPBO-P pour le schéma Alterné-C.

Le schéma d'optimisation Alterné-C est une extension d' α -Expansion. Il est composé de toutes les expansions et de tous les sauts possibles. Cela a deux conséquences : 1) la solution finale est minimale pour un nombre plus grand de fusions binaires, 2) le temps d'exécution d'un cycle du schéma est plus important. Il en résulte, en principe, une énergie plus basse dans un temps plus important. Cependant, l'énergie possède de nombreux minima locaux qui peuvent empêcher une descente de l'énergie. Inversement, le surcoût en temps de chaque cycle peut être compensé par un nombre de cycles réduit. Nous avons aussi testé trois méthodes d'optimisation pour résoudre les sous-problèmes de fusion : la fusion relaxée, QPBO et QPBO-P. Les résultats pour les cinq images de la base de test sont donnés dans le Tableau 2.4.

Tout d'abord, nous comparons α -Expansion avec Alterné-C. L'introduction des fusions par saut dans Alterné-C permet de faire baisser l'énergie minimale substantiellement (de l'ordre de

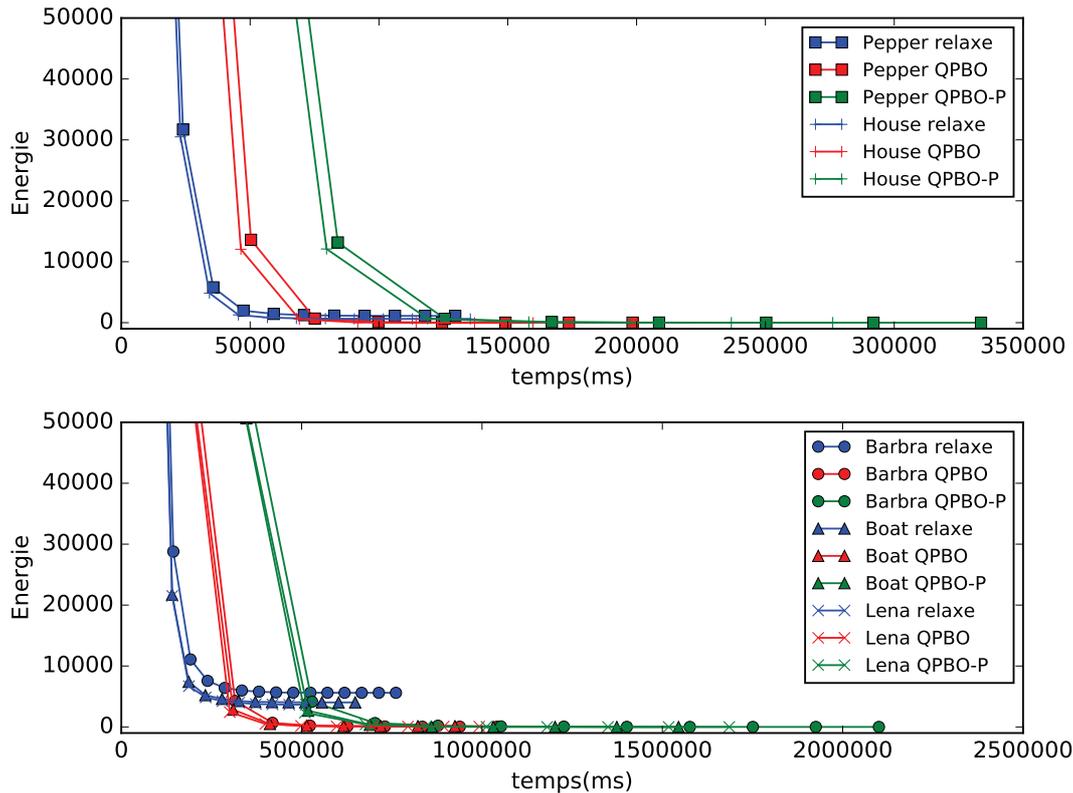


Figure 2.7 – Évolution de la valeur de l'énergie au cours des itérations pour les trois méthodes d'optimisation par GC sur la base de test de dé-bruitage dans le cas du schéma d'optimisation Alterné-C. Les énergies sont données à une constante près et seule la première itération de chaque méthode est en dehors du graphique.

2% de l'énergie) quelle que soit la méthode d'optimisation des fusions. Le temps de convergence est 2 à 3 fois plus long que pour la fusion relaxée, et de +5% à +40% pour les méthodes QPBO et QPBO-P. En effet, la réduction du nombre de cycles d'Alterné-C compense en partie le temps supplémentaire pour l'exécution d'un cycle. Les temps d'exécutions des méthodes QPBO et QPBO-P restent supérieurs à ceux de la fusion relaxée. En conclusion, le schéma α -Expansion est moins bien adapté aux problèmes où la fonction de régularisation possède une cuvette convexe autour de zéro que le schéma Alterné-C qui comprend en plus des fusions par saut, et ce, quelle que soit la méthode d'optimisation utilisée pour les fusions.

Ensuite, nous comparons Alterné-L et Alterné-QL avec Alterné-C. Les schémas Alterné-L et Alterné-QL sont des versions d'Alterné-C avec un ensemble restreint de fusions par itération. Nous remarquons dans les énergies sont très proches pour les trois schémas. Le schéma Alterné-C donne l'énergie la plus basse sauf pour l'image "Barbara". Les temps d'exécutions du schéma Alterné-L est très variable : il peut être réduit de moitié ou doubler par rapport à celui d'Alterné-C. Les temps de convergence d'Alterné-QL sont inférieurs à ceux des deux autres schémas. En conclusion, les trois schémas sont adaptés à la résolution de ce problème. Le schéma Alterné-C donne généralement de plus basses valeurs de l'énergie. Le schéma Alterné-L donne une énergie un peu supérieure et son temps de convergence peut être plus important. Le schéma Alterné-LQ permet un gain important en temps de convergence en impactant peu la valeur de l'énergie finale.

1.4.2 Comparaison des temps de convergence des différents schémas

Nous avons vu précédemment que le temps de convergence de QPBO-P est supérieur à celui de QPBO pour des performances très proches en terme d'énergie. La fusion relaxée converge plus rapidement que les méthodes QPBO. Si les temps de convergence sont très longs, il est très

fréquent que les derniers cycles des schémas ne réduisent que très peu l'énergie. En pratique il est courant d'arrêter les itérations de manière anticipée et les conclusions sur les performances respectives des méthodes peuvent changer. La Figure 2.7 représente l'évolution de la valeur d'énergie au cours des itérations. Les cinq images sont regroupées en deux graphiques en fonction de leur taille. Les comportements de QPBO et QPBO-P sont très proches, si ce n'est que le temps de convergence est plus long pour QPBO-P. La fusion relaxée permet une décroissance de l'énergie moins rapide à chaque cycle mais est exécutée plus rapidement. Elle converge vers une solution d'énergie supérieure à celle des solutions obtenues par les méthodes QPBO.

1.5 Conclusion

Sur le problème de dé-bruitage, dans le cas où l'attache aux données est en carré et la régularisation en valeur absolue, le minimum global de l'énergie peut être atteint par α -Expansion en un temps réduit par rapport à la méthode d'Ishikawa. Cependant, le minimum atteint par α -Expansion s'écarte du minimum global lorsque la fonction f n'est pas convexe. Il en est de même lorsque r est la fonction valeur absolue lissée. Dans ce cas, un schéma d'optimisation par saut permet d'atteindre le minimum global de l'énergie en des temps assez réduits. Il apparaît donc que si α -Expansion présente parfois de bonnes performances d'optimisation, le schéma par saut est mieux adapté aux énergies convexes.

Dans le cas où r est une fonction SEF, ni concave ni convexe sur \mathbb{R}^+ , la valeur d'énergie minimale obtenue par α -Expansion est éloignée de celles obtenues par des schémas d'optimisation qui comprennent des fusions par saut. Peu de changements sont observés lorsqu'on limite les fusions par saut de faibles valeurs. En alternant avec les fusions par expansion échantillonnées avec un pas large, le gain en temps de calculs est intéressant. C'est le schéma Alterné-LQ.

L'optimisation des sous-problèmes de fusions par QPBO et QPBO-P donne des résultats similaires et atteint des valeurs minimales d'énergie inférieures à celles de la fusion relaxée. Toutefois ce gain est limité au vu du temps de calcul nécessaire à la convergence de la solution. En effet, les premières itérations, optimisées par fusion relaxée permettent d'obtenir rapidement des valeurs d'énergie assez basses.

2 Reconstruction stéréoscopique par coupure de graphe

La reconstruction stéréoscopique consiste à reconstruire une scène en trois dimensions, à partir de deux images prises à partir de points de vue différents. La plupart des méthodes reposent sur la recherche de points homologues entre deux images qui correspondent au même élément d'un objet de la scène en trois dimensions. Si la géométrie d'acquisition et la position relative des caméras sont connues, alors les coordonnées relatives en trois dimensions peuvent être déduites des positions dans l'images des deux points homologues. La méthode qui vise à obtenir les points homologues est appelée "appariement" ou "mise en correspondance". Nous rappellerons la géométrie particulière dans laquelle la recherche de points homologues est simplifiée (Section 2.1), puis nous rappellerons quelques mesures de similarité pour la mise en correspondance de pixels (Section 2.2). Toutefois, apparier des points homologues est une tâche difficile à cause des ambiguïtés et elle n'est souvent pas suffisante pour déterminer de façon unique la profondeur de la scène. Aussi un terme d'a priori sur la solution est ajouté au modèle (Section 2.3). Le problème de reconstruction stéréoscopique peut être modélisé de différentes manières et peut intégrer en plus des contraintes géométriques sur la solution (Section 2.4).

2.1 Géométrie rectifiée

Nous ne connaissons de la scène en trois dimensions que nous souhaitons reconstruire seulement deux images, I^1 et I^2 , obtenues à partir de caméras avec des points de vue différents. Ces

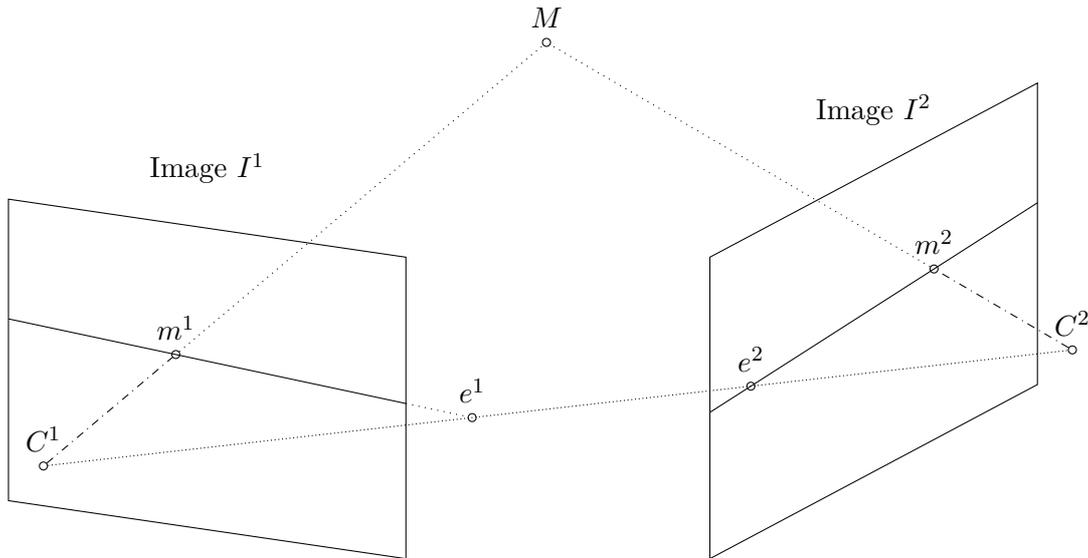


Figure 2.8 – Couple d’images et leurs points épipolaires e^1 et e^2 . Pour une position m^1 de l’image I^1 , l’homologue de ce point, m^2 , est contraint d’être sur une droite dans l’image I^2 , appelée ligne épipolaire. Suivant la profondeur du point M le long de la droite (C^1, m^1) , le point m^2 se déplace sur la ligne épipolaire.

images sont formées par projection centrale des points 3D. C’est un modèle simple appelé le modèle sténopé. Il consiste à considérer que les rayons lumineux convergent en ligne droite vers un point appelé centre optique et que l’image est formée des intersections des rayons avec un plan appelé plan focal. Soit C^1 et C^2 les centres optiques des deux caméras, M un point 3D de la scène et m^1 et m^2 les projetés de M dans l’image I^1 et I^2 , respectivement. Nous pouvons faire les déductions suivantes :

- Les points C^1, m^1, M et C^2, m^2, M sont alignés. Ces cinq points sont co-planaires.
- L’homologue de m^1 se trouve à l’intersection du plan (C^1, C^2, M) avec le plan focal de l’image I^2 , soit sur une droite. Cette droite définit la ligne épipolaire du point m^1 dans l’image I^2 .
- L’intersection e^2 de la droite (C^1, C^2) avec le plan focal de l’image I^2 , appelé épipôle, appartient à toutes lignes épipolaires. Ainsi, les lignes épipolaires d’une image convergent en son épipôle.
- Le raisonnement symétrique est aussi valide.

Ainsi la recherche des points homologues peut être limitée à une droite dans l’autre image. Ces relations géométriques sont illustrées dans la Figure 2.8. Par une homographie, les lignes épipolaires peuvent être rendues parallèles et alignées deux à deux entre les deux images. Cette géométrie est la géométrie rectifiée. Dans cette géométrie, l’appariement est simplifié : les homologues d’une ligne sont dans la ligne à la même hauteur de l’autre image. Il existe une relation simple entre les deux positions horizontales des points appariés dans les images et la profondeur du point 3D dans la scène, comme illustré dans la Figure 2.9. Soit un pixel p de l’image I^1 apparié avec le pixel q de l’image I_2 , la disparité associée à ce couple de points est la différence de position horizontale dans leur image respective, i.e $d = i_p^1 - i_q^2$. On montre alors que la profondeur est proportionnelle à l’inverse de la disparité.

2.2 Modèles d’attache aux données

Comme la profondeur d’un pixel peut être déduite de deux positions image avec des points de vue différents, le problème de reconstruction stéréoscopique revient à appairer les points homologues. Toutefois, les modèles qui se fondent sur les points homologues souffrent de deux problèmes :

- Les points qui se ressemblent ne sont pas nécessairement des images d’un même point.

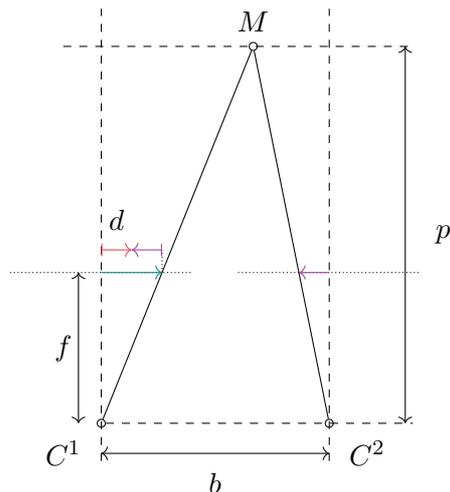


Figure 2.9 – Vue de dessus d’un couple stéréoscopique en géométrie rectifiée. Nous observons qu’il existe une relation géométrique entre la profondeur du pixel et la différence de position horizontale en pixel. Cette différence est appelée la disparité.

- Les positions image correspondant à la profondeur vraie ne sont pas nécessairement des points homologues à cause des occultations de la scène.

Lorsque deux pixels sont appariés, on peut faire deux hypothèses :

- H_0 , ce sont des points homologues, ils correspondent bien au même objet.
- H_1 , ce ne sont pas des points homologues, ils ne correspondent pas au même objet.

Si nous possédons une mesure binaire de ressemblance entre deux positions image, utiliser cette mesure pour estimer l’appariement peut exposer à des erreurs. Les appariements se répartissent alors en quatre classes :

- Les vrais positifs, ils se ressemblent et correspondent à la profondeur vraie, ce sont des points homologues.
- Les faux positifs, ils se ressemblent mais ne correspondent pas à la profondeur vraie. Ce sont des erreurs dues à l’ambiguïté du problème d’appariement.
- Les faux négatifs, ils ne se ressemblent pas mais correspondent à la profondeur vraie. Cela est dû aux occultations ou à des changements d’aspect entre les images comme par exemple, des reflets.
- Les vrais négatifs, ils ne se ressemblent pas et ne correspondent pas à la profondeur vraie.

Les faux positifs sont liés à l’ambiguïté. Cette ambiguïté peut être atténuée par l’introduction d’un terme d’a priori dans le modèle. Les faux négatifs peuvent être pris en compte de deux manières : 1) limiter le coût associé à un tel appariement pour permettre qu’il puisse apparaître dans la solution minimale, 2) introduire des contraintes géométriques dans le modèle pour identifier les pixels occultés et leur appliquer un traitement différent. Une grande variété de fonction de similarité ont été utilisées en reconstruction stéréoscopique [Hirschmuller et Scharstein, 2009; Chambon et Cruzil, 2011]. Nous faisons, dans cette section, un bref aperçu de certaines de ces fonctions.

2.2.1 Méthodes par pixel

En reconstruction, le terme d’attache aux données consiste le plus souvent en une comparaison locale de deux zones de l’image. Si elles se ressemblent, alors le coût d’attache aux données est faible. Inversement, si elles sont dissemblables le coût d’attache aux données sera élevé. Nous faisons tout d’abord l’hypothèse que l’intensité des points homologues est proche. L’attache aux données prend la forme d’une fonction de la différence d’intensité entre le pixel p de la première image I^1 et le pixel q de la deuxième image I^2 :

$$Data(p, q) = f(\Delta Int(p, q)) \quad (2.10)$$

avec f une fonction croissante et

$$\Delta \text{Int}(p, q) = |\text{Int}^1(p) - \text{Int}^2(q)| \quad (2.11)$$

Avec la formulation bayésienne, cette fonction correspond au $-\log$ de la distribution du bruit qu'il peut avoir sur l'intensité de la première image en supposant qu'il n'y a pas de bruit sur la deuxième image. Les images sont soumises au bruit photonique du capteur. Ce bruit suit généralement une loi gaussienne indépendante. Ainsi dans le cas de points homologues (hypothèse H_0), f est une fonction carrée de la différence d'intensité. Lorsque les points ne sont pas homologues (hypothèse H_1), il n'a pas de lien entre les intensités et nous considérons que la différence d'intensité suit une loi uniforme. Ainsi le modèle d'attache aux données dépend de l'hypothèse. Une possibilité est de prendre le maximum de vraisemblance de ces deux hypothèses, il en résulte une attache aux données de la forme :

$$f(x) = \min(kx^2, T) \quad (2.12)$$

soit une fonction carrée tronquée. Une autre possibilité est de faire la somme des fonctions pondérées par la probabilité que les points correspondent à des points homologues, ce qui donne un modèle de la forme :

$$f(x) = -\log(e^{-\frac{x^2}{2s^2}} + K) \quad (2.13)$$

avec s un paramètre d'échelle et K une constante positive. Plus généralement, la fonction f est souvent une fonction convexe autour de zéro et concave pour des fortes valeurs, comme avec les fonction SEF. Une autre justification de la partie concave de la fonction est que l'hypothèse H_1 (ce ne sont pas des points homologues) n'indique pas nécessairement que la profondeur sélectionnée est fautive. C'est le cas en présence d'occultations. Il ne faut donc pas trop pénaliser cet appariement dans l'énergie. Ainsi, il est souvent préférable de limiter les coûts d'appariement de pixels dissemblables pour ne pas complètement interdire l'apparition de ces appariements dans la solution.

Si maintenant nous disposons d'image en couleurs, nous pouvons exploiter cette information de deux manières : soit pour calculer une distance couleur entre les pixels, soit pour corriger certaines perturbations. Dans le premier cas, il s'agit le plus souvent d'une distance dans un espace colorimétrique spécifique. Dans le deuxième cas, il peut s'agir d'un rapport de différences de canaux pour rendre l'attache aux données invariante à une transformation affine qui affecte chaque canal de la même manière, les uns par rapport aux autres. Toutefois cette exploitation de la couleur n'est possible que lorsque les canaux sont bien étalonnés.

2.2.2 Méthodes sur un voisinage

Du fait de l'ambiguïté du problème de reconstruction, il est souvent nécessaire de prendre en compte un voisinage autour des deux positions pixel pour caractériser leur ressemblance. L'attache aux données peut être la somme des attaches aux données calculées sur chaque pixel. De plus, l'approche par voisinage permet de rendre l'attache aux données invariante à certaines perturbations, comme par exemple l'ajout d'une constante entre les intensités des deux images. Pour ce faire, l'intensité moyenne de chaque sous-image est soustraite. Par exemple, en normalisant les intensités des sous-images par l'écart-type, on obtient l'invariance à un facteur entre l'intensité des deux images.

Une mesure de similarité très utilisée est le "Census" [Zabih et Woodfill, 1994]. Cette mesure est non-paramétrique, invariante à une transformation qui conserve l'ordre entre les valeurs d'intensité. Le "Census" est calculé en deux étapes, les sous-images sont binarisées puis la distance de Hamming est calculée entre les deux sous-images binaires :

$$\text{Census}(p, q) = \sum_{p+dp \in \mathbf{W}_p} \mathbb{1}(\text{CT}^1(p, dp) \neq \text{CT}^2(q, dp)) \quad (2.14)$$

avec

$$CT^i(p, dp) = \mathbb{1}(\text{Int}^i(p+dp) > \text{Int}^i(p)) \quad (2.15)$$

où $\mathbb{1}$ est la fonction indicatrice qui donne 0 si l'argument est faux et 1 si l'argument est vrai.

Fenêtres fixes Le voisinage pris en compte autour de chaque pixel apparié peut être une fenêtre de taille fixe dans les deux images. L'inconvénient de cette méthode est qu'elle suppose une profondeur uniforme sur la fenêtre. Or, si la profondeur varie dans la fenêtre, la fenêtre re-projetée dans l'autre image ne sera pas de la même forme. Une image est souvent composée d'objets qui sont relativement homogènes en terme d'intensité et de profondeur. Cela limite l'impact de ce problème, mais les contours des objets sont souvent dégradés. Une alternative est de prendre en compte plusieurs fenêtres décentrées comme dans [Fusiello *et al.*, 1997].

Fenêtres adaptatives La taille de la fenêtre peut être adaptée selon des critères calculés localement. En effet, il est préférable de réduire la taille de la fenêtre en présence de changements brusques de profondeur. En faisant l'hypothèse que les gradients d'intensité correspondent à des contours de disparité, la taille de la fenêtre peut être ajustée en utilisant un critère sur les intensités comme dans [Veksler, 2003]. D'autres critères peuvent être pris en compte comme dans [Yoon et Kweon, 2006]. Une de ces méthode de fenêtre adaptative est l'agrégation des coûts fondée sur des croix (*Cross Based Cost Agregation*, CBCA, [Zhang *et al.*, 2009]). Elle consiste à moyennner les coûts à profondeur constante sur un support. Un "bras" depuis le pixel p dans la direction r est l'ensemble des pixels q se succédant et possédant une intensité proche jusqu'à une certaine distance maximale selon :

$$\begin{aligned} |\text{Int}(p) - \text{Int}(q)| &\leq T_I \\ \text{Dist}(p, q) &\leq T_D \end{aligned} \quad (2.16)$$

où T_I est le seuil sur l'intensité et T_D le seuil sur la distance. Le support \mathbf{W}_p^1 est composé des deux "bras" horizontaux calculés à partir des pixels des deux "bras" verticaux du pixel p de la première I^1 . Le support dans la deuxième image I^2 est défini de la même manière. Le support pour la mise en correspondance du pixel p avec le pixel q est l'intersection de ces deux supports,

$$\mathbf{U}_{p,q} = \{k | p + k \in \mathbf{W}_p^1, q + k \in \mathbf{W}_q^2\} \quad (2.17)$$

où k est un vecteur de décalage de position dans l'image. La CBCA peut être répétée i fois de manière itérative.

Fenêtres inclinées Lorsque la profondeur varie, la projection de la fenêtre change de forme du fait de la perspective. Dans le cas où les objets de la scène sont des plans, le correspondant d'une fenêtre est une fenêtre inclinée. Si cette déformation n'est pas prise en compte, la plupart des pixels de la fenêtre ne seront pas appariés avec le bon pixel de l'autre image. Certaines approches consistent à faire plusieurs hypothèses de plan et sélectionner l'une d'entre elles, comme dans [Ranft et Strauß, 2014].

2.3 Modèles de régularisation

Le terme de régularisation doit correspondre aux caractéristiques attendues de la solution du problème. La régularisation possède un ordre qui correspond au nombre de paramètres de la solution mis en jeu simultanément dans un terme de régularisation, moins 1. Le modèle de régularisation le plus fidèle au problème est souvent d'ordre assez élevé. Or, l'optimisation de tels modèles est difficile comme on l'a vu dans le Chapitre 1. C'est pourquoi la plupart des modèles mettent en jeu des régularisations d'ordre 1 et parfois d'ordre 2. Ainsi, la régularisation est souvent une somme de critères locaux entre 2 ou 3 paramètres de la solution.

2.3.1 Régularisation du premier ordre

L'hypothèse la plus fréquente est l'hypothèse fronto-parallèle : les pixels voisins sont à la même profondeur. Il est plus pratique d'exprimer la distribution des écarts de profondeur en fonction des différences de disparités. Alors la régularisation s'écrit comme :

$$Reg_{pq}(d_p, d_q) = r(|d_p - d_q|) \quad (2.18)$$

L'hypothèse fronto-parallèle peut être prise en compte de manière plus ou moins forte dans le modèle. Il peut s'agir d'une régularisation par la variation totale qui correspond à choisir la valeur absolue pour la fonction r . Lorsque la scène est composée d'objets fronto-parallèles à des profondeurs distinctes, il est préférable de prendre une fonction r concave. De telles fonctions pénalisent plus les petites variations de profondeur que les grandes variations, par rapport à une fonction linéaire. Un cas limite de fonction concave est la fonction de Potts où toute variation de profondeur à le même coût. Souvent les objets sont planaires mais non-frontaux, de sorte que des sauts de disparité entre voisins est assez fréquents. Ainsi, de façon analogue au problème du dé-bruitage, la fonction r concave est relaxée pour ne pas trop pénaliser ces variations. La fonction résultante est concave pour des fortes valeurs et présente une cuvette convexe en zéro. Une fonction très utilisée est la fonction que nous nommons Potts lissée :

$$Potts_l(x) = \begin{cases} 0, & \text{si } x = 0 \\ P1, & \text{si } x = 1 \\ P2, & \text{sinon} \end{cases} \quad (2.19)$$

avec $P1 < P2$. Le choix d'une méthode d'optimisation peut aussi fortement contraindre le choix de la fonction de régularisation, de sorte que certaines fonctions sont très souvent utilisées, comme par exemple, la fonction valeur absolue qui est à la fois concave et convexe sur \mathbb{R}^+ et la fonction valeur absolue tronquée qui est concave sur \mathbb{R}^+ .

L'hypothèse que les contours des objets correspondent à la fois à des contours d'intensité et des contours de profondeur est souvent utilisée pour pondérer localement la régularisation. Les valeurs des coefficients de régularisation sont alors plus élevés dans les zones homogènes et plus faibles en présence de gradients d'intensité.

2.3.2 Régularisation du deuxième ordre

Optimiser globalement une énergie du deuxième ordre est une tâche difficile. Une telle énergie est utilisée lorsque l'on souhaite pénaliser la valeur absolue de la dérivée seconde. Une approche dans le domaine continu est d'introduire des variables auxiliaires que l'on veut proches des valeurs des dérivées et de régulariser des variables auxiliaires [Bredies *et al.*, 2010]. L'énergie prend alors la forme :

$$E(\mathbf{l}, \mathbf{Aux}) = \sum_{p \in \mathbf{I}} f_p(l_p) + \sum_{(p,q) \in \mathbf{N}'} h_{pq}(l_p - l_q - \text{Aux}_{pq}) + \sum_{(a,b) \in \mathbf{N}_A} r_{ab}(\text{Aux}_a - \text{Aux}_b) \quad (2.20)$$

avec \mathbf{N}_A l'ensemble des couples de variables auxiliaires pris en compte dans la régularisation. La fonction h doit maintenir la correspondance entre les différences de label et la variable auxiliaire associée. Le problème est optimisé alternativement en fonction des labels et en fonction des variables auxiliaires. Pour que l'optimisation soit efficace, la contrainte introduite par la fonction h doit être suffisamment relaxée, ce qui nous éloigne du problème initial. Une telle approche peut être mise en œuvre par GC, mais il nous a été assez difficile d'obtenir des résultats satisfaisants. D'une part, dans le cas discret, il est difficile de représenter les pentes faibles : elles correspondent à des plateaux de disparité séparés par des variations. D'autre part, l'optimisation alternée converge souvent vers une solution très éloignée de la solution recherchée. Woodford *et al.* [2009] proposent une méthode fondée sur la coupure de graphe pour traiter le problème sans relaxation.

L'énergie est minimisée par un schéma d'optimisation approchée et les fusions binaires sont résolues par réduction d'ordre et minimisation approchée de FPB non-sous-modulaires.

Une des conséquences de l'augmentation d'ordre de la régularisation est l'augmentation de la taille du problème. Souvent le problème n'est pas traité par pixel mais selon des régions obtenues par une sur-segmentation de l'image. Cela permet de réduire le nombre de paramètres du modèle en estimant un vecteur de paramètres par segment. Il peut s'agir d'estimer un plan par région et de pénaliser les angles entre les plans de régions voisines comme par exemple dans [Tao *et al.*, 2001]. L'inconvénient de cette méthode est que la solution dépend de la qualité de la segmentation. D'autres méthodes optimisent alternativement la segmentation et les profondeurs comme par exemple dans [Bleyer *et al.*, 2011].

Dans le but de compenser le surcoût en calculs nécessaire pour minimiser une énergie d'ordre 2, les propositions de fusion sont obtenues par d'autres méthodes comme par exemple dans [Vogel *et al.*, 2015]. Cette stratégie de fusion permet d'optimiser plus efficacement qu'un schéma d'optimisation classique comme α -Expansion.

2.4 Modèle de reconstruction par GC

Nous considérons un couple d'images en géométrie épipolaire. La profondeur se mesure alors à partir de la disparité horizontale. Les valeurs de disparité pour un pixel sont comprises dans un intervalle $d \in [d_{min}, d_{max}]$. Nous prendrons arbitrairement $d_{min} = 0$ dans la suite. Les disparités qui mettent en jeu des pixels en dehors de l'image peuvent soit être ignorées (alors chaque pixel se voit attribuer une limite d_{max}^p), soit un coût est associé à ces appariements et d_{max} est identique sur toute l'image. Dans cette section, nous présentons deux manières de modéliser le problème de reconstruction. La première, Section 2.4.1, décompose la scène en éléments de volume. La seconde, Section 2.4.2, formule la profondeur comme une fonction 2D d'une image. L'énergie qui décrit le modèle est composée d'un terme d'attache aux données, d'un terme de régularisation et, éventuellement, de termes de contraintes. La principale difficulté de la modélisation est la prise en compte des occultations. Dans la Section 2.4.3, nous verrons une alternative avec une méthode de détection des occultations par post-traitement.

2.4.1 Optimisation d'un volume par GC

Il est possible de formuler le problème de reconstruction stéréoscopique comme l'optimisation d'un volume. Cette représentation est appelée uv-disparité. Chaque appariement possible est représenté par un nœud \mathcal{X}_{pq} . La variable associée au nœud possède deux affectations :

- 0 : le volume autour de cette position est opaque (nous diront abusivement qu'il est plein).
- 1 : le volume autour de cette position est transparent (vide).

Nous appelons ligne de vue du pixel p l'ensemble des appariements qui mettent en jeu cette position. Il est à noter que parcourir la ligne de vue dans le sens croissant des disparités correspond au trajet inverse de la lumière. La variable associée au nœud \mathcal{X}_{pq} possède deux écritures : $x_d^{p,1}$ et $x_d^{q,2}$, selon qu'on parcourt la ligne de vue depuis le pixel p ou le pixel q . L'affectation des nœuds permet de décrire un volume qui peut éventuellement intersecter plusieurs fois certaines lignes de vue. Nous ajoutons un terme pour pénaliser deux appariements voisins à la même disparité qui ne possèdent pas la même affectation.

Nous avons deux fonctions de similarité D_p^1, D_p^2 qui donnent une valeur de coût en fonction de disparité attribué au pixel p dans l'image I^1 et dans l'image I^2 , respectivement. Pour l'homogénéité des équations, nous considérons qu'il existe un appariement au-delà de la disparité maximale, dont la variable est notée $x_{d_{max}+1}^p$, toujours vide et donc assimilable au puits. Nous considérons aussi un appariement en deçà de la disparité minimale, dont la variable est notée x_{d-1}^p , toujours plein et donc assimilable à la source. Cela revient à prendre en compte qu'il peut y avoir des objet plus loin que l'arrière plan. Dans un premier temps, nous considérerons un coût associé à cette affectation $D_p^1(-1)$. La reconstruction peut alors être formulée comme la

minimisation de l'énergie :

$$\begin{aligned}
E(\mathbf{x}) &= \sum_{p \in I^1} \sum_{d=-1}^{d_{max}^p} D_p^1(d) \bar{x}_d^{p,1} x_{d+1}^{p,1} + Occ^1 \bar{x}_{d+1}^{p,1} x_d^{p,1} \\
&+ \sum_{p \in I^2} \sum_{d=-1}^{d_{max}^p} D_p^2(d) \bar{x}_d^{p,2} x_{d+1}^{p,2} + Occ^2 \bar{x}_{d+1}^{p,2} x_d^{p,2} \\
&+ \sum_{(p,p') \in N} \sum_{d=0}^{d_{max}^{p,p'}} \lambda (\bar{x}_d^{p,1} x_d^{p',1} + \bar{x}_d^{p',1} x_d^{p,1})
\end{aligned} \tag{2.21}$$

où Occ^1 et Occ^2 sont des pénalités d'occultation dans les lignes de vue de l'image I^1 et I^2 , et $d_{max}^{p,p'}$ est le minimum de d_{max}^p et $d_{max}^{p'}$. Le troisième terme de l'énergie (2.21) est la régularisation à disparité constante. Cette énergie du premier ordre peut être optimisée globalement par coupure de graphe. La structure de graphe est représentée dans la Figure 2.10 : les coûts relatifs à l'image I^1 sont portés par les arcs horizontaux, les coûts relatifs à l'image I^2 sont portés par les arcs verticaux, les coûts de régularisation sont portés par les arcs diagonaux. Dans le cas où les coûts $D_p^1(-1)$ sont infinis, les nœuds reliés à la source peuvent être assimilés à la source est la construction se simplifie comme présenté dans la Figure 2.11.

Si l'association du coût d'occultation Occ aux transitions 1-0 se justifie par la non-visibilité de cette zone, l'association du coût de ressemblance entre les images D pour les transitions 0-1 est discutable. En effet, cette association est correcte seulement pour la dernière transition visible. Les autres transitions sont masquées par un volume. Ainsi en toute rigueur, un critère de visibilité du volume devrait être ajouté au modèle pour choisir entre un terme de similarité et un terme d'occultation :

$$\left(D_p^1(d) v(x_d^{p,1}) + Occ^1(1 - v(x_d^{p,1})) \right) \bar{x}_d^{p,1} x_{d+1}^{p,1} \tag{2.22}$$

avec

$$v(x_d^{p,1}) = \prod_{k=d+1}^{d_{max}^p} \bar{x}_k^{p,1} \tag{2.23}$$

Malheureusement, l'énergie résultante est d'ordre élevé et donc difficile à optimiser par GC. Une réduction d'ordre est toujours possible mais une telle opération ajoute de nombreux termes non-sous-modulaires à l'énergie de sorte que cette approche est peu utilisée. Une alternative au critère de visibilité est d'introduire une contrainte d'unicité de la transition 1-0. Ainsi les transitions 1-0 correspondent toujours à une position visible, mais les volumes qui peuvent être représentés par l'énergie sont alors limités. La contrainte d'unicité peut aussi être relaxée. En effet, les coupes qui intersectent une seule fois une ligne de visée mettent en jeu moins de capacités d'arc et sont donc plus susceptibles d'être sélectionnées lors de l'optimisation.

2.4.2 Optimisation d'une carte de disparité

Cette approche consiste à considérer une seule valeur de profondeur par pixel. Ainsi un label de disparité est associé à chaque pixel de l'image choisie et le problème revient à optimiser une image appelée la carte de disparité. En se plaçant dans la géométrie de l'image I^1 , l'énergie est de forme :

$$E(\mathbf{l}) = \sum_{p \in I^1} D_p(l_p) + \sum_{(p,q) \in N^1} Reg(l_p, l_q) \tag{2.24}$$

Dans le cas où $Reg(l_p, l_q)$ est la valeur absolue de la différence, avec les notations précédentes, appliquer la méthode d'Ishikawa pour résoudre la carte de disparité est équivalent à résoudre globalement le volume présenté dans la Section 2.4.1 sous la contrainte d'unicité. Le principal avantage de l'introduction de la carte de disparité est de simplifier le problème en faisant de la

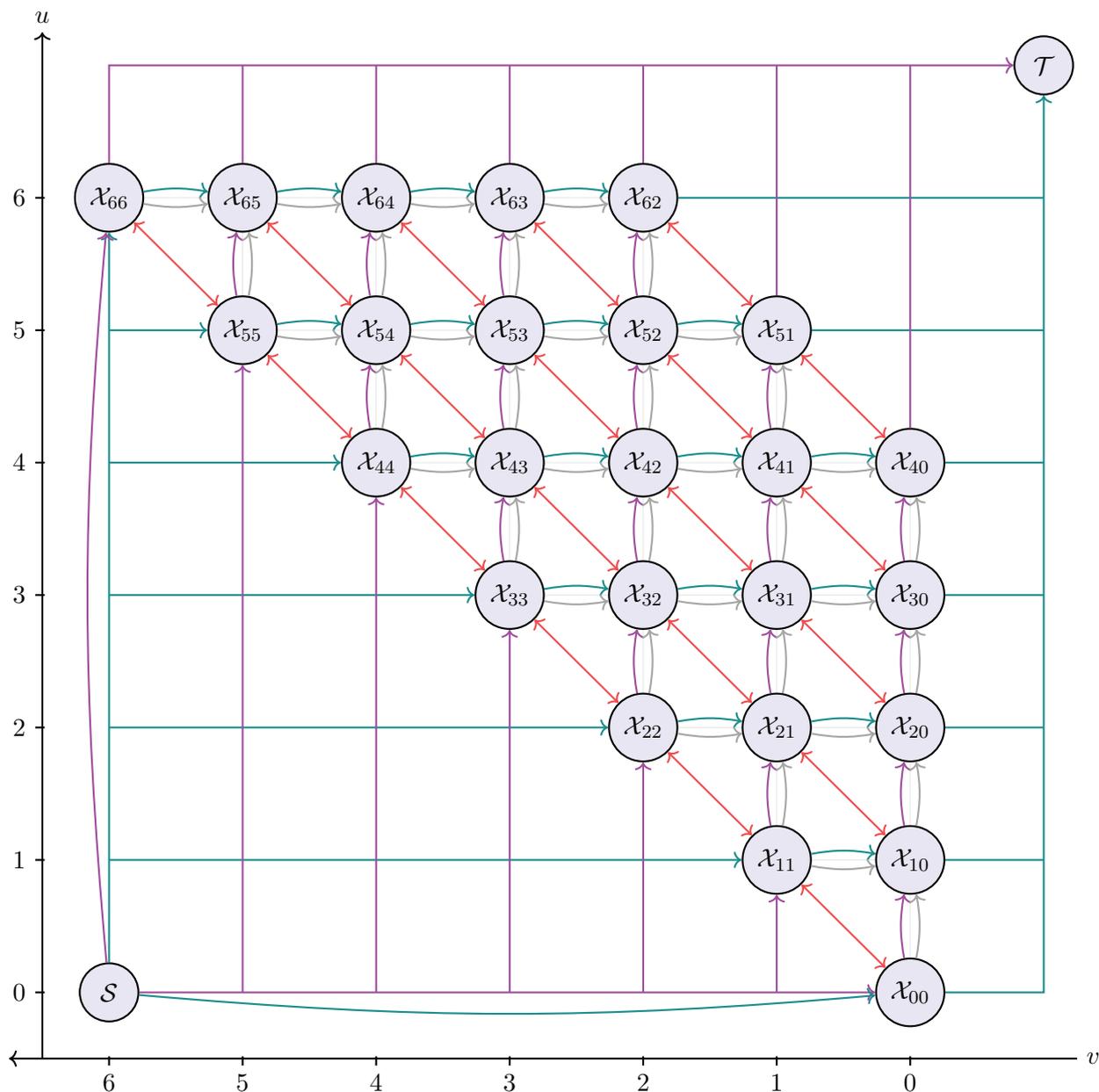


Figure 2.10 – Structure de graphe pour la reconstruction stéréoscopique de deux lignes en correspondance épipolaire dans les images I^1 et I^2 rectifiées. Les disparités croissent selon la première diagonale. Les lignes de vue de l'image 1 correspondent aux lignes horizontales dans le graphe (en vert) et les lignes de vue de l'image 2 correspondent aux lignes verticales (en violet), les contraintes le long des lignes de visée sont représentées en gris. Les liens de régularisation entre les lignes de vues correspondent aux diagonales (en rouge).

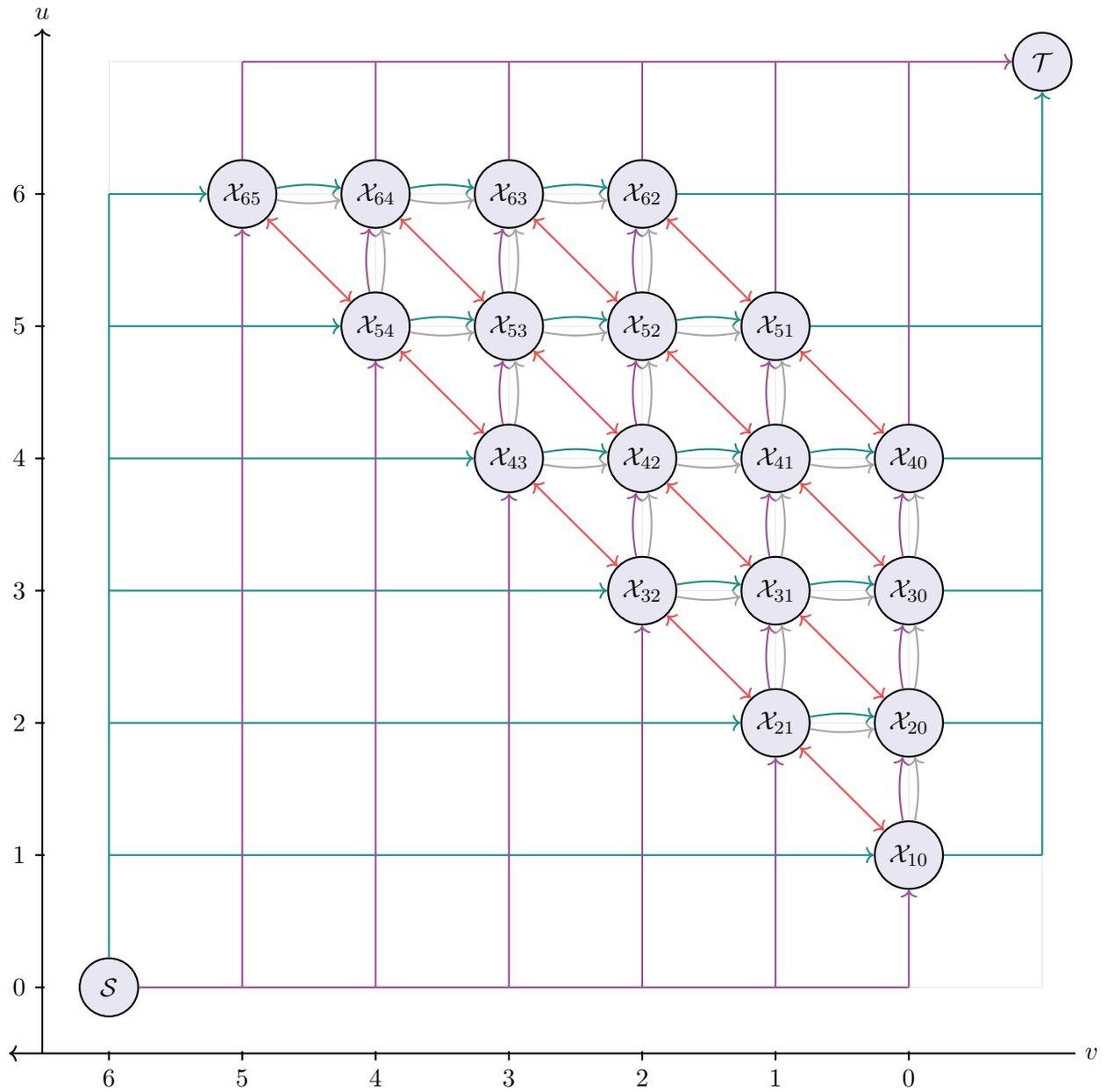


Figure 2.11 – Structure de graphe présentée dans la Figure 2.10 dans le cas où les coûts $D_p(-1)$ sont infinis et après simplification.

profondeur une fonction de l'image. L'autre avantage est qu'il existe de nombreuses méthodes approchées qui permettent d'optimiser un problème multi-labels pour une classe étendue de fonctions de régularisation. L'inconvénient est que la géométrie des scènes qui peuvent être reconstruites a été limitée. Cela pose problème pour la prise en compte des occultations. Une possibilité est d'ajouter un label "occulté". L'attache aux données pour un pixel occulté est fixée à la constante Occ et un coût de régularisation spécifique est associé à la transition "occulté"/"non-occulté". Cependant cette approche ne détectera l'occultation que lorsque l'attache aux données a une valeur élevée, ce qui conduit à une sous-détection des occultations. Dans [Kolmogorov *et al.*, 2014] le problème est formulé différemment ce qui permet de prendre en compte les occultations en imposant une contrainte d'ordre, ce qui est plus satisfaisant. Dans [Wei et Quan, 2005], un critère de visibilité est pris en compte, et des approximations sont réalisées pour que le problème puisse être traité par GC. Enfin, notons que si le modèle est suffisamment robuste pour donner une carte de disparité satisfaisante dans les zones non-occultées, les zones occultées de la carte de disparité peuvent aussi être identifiées a posteriori par validation croisée, comme cela est présenté dans la section suivante.

2.4.3 Validation croisée entre les cartes de disparité

La validation croisée est une méthode d'identification des zones occultées par post-traitement. La carte de disparité est calculée selon deux points de vue (image I^1 et image I^2). La validation croisée se fonde sur le principe que les points homologues sont à la même profondeur dans les deux cartes de disparité. Dans les cartes de disparité vraie, les points homologues peuvent être distingués des occultations en comparant la disparité des points appariés. Pour prendre en compte les effets de la quantification des disparités et aussi des erreurs de faible amplitude, nous tolérons les écarts de disparité de 1 :

$$|l_p^1 - l_{p-l_p^1}^2| \leq 1 \quad (2.25)$$

Si la condition (2.25) n'est pas remplie, les pixels sont considérés comme occultés. Dans le cas où nous ne disposons que d'estimations des deux cartes de disparité, le même test peut être appliqué, mais il peut y avoir une sous-détection des occultations là où les disparités correspondent de manière fortuite ou une sur-détection là où des erreurs ont été commises. L'efficacité de la validation croisée dépend donc de la qualité de l'estimation des deux cartes de disparité.

3 Expérimentations de reconstruction

Dans cette section, nous écrivons le problème de la reconstruction en fonction d'une image de disparité. L'énergie à optimiser est de la forme :

$$E(\mathbf{l}) = \sum_{p \in \mathbf{I}} D_p(l_p) + \lambda \sum_{(i,j) \in \lambda \mathbf{N}'} r(l_i - l_j) \quad (2.26)$$

où \mathbf{I} est l'image des disparités, l_p le label de la disparité du pixel p de coordonnées (i, j) et \mathbf{N}' l'ensemble des pixels voisins pris dans un sens arbitraire. Il n'y a donc pas de prise en compte explicite des occultations. La formulation de l'énergie est analogue à l'énergie du dé-bruitage (2.1). La fonction d'attache aux données en dé-bruitage est une fonction qui possède un unique minimum. L'attache aux données en reconstruction ne possède pas cette propriété ; elle dépend de la structure de l'image et possède de nombreux minima locaux. L'énergie possède donc aussi plus de minima locaux par rapport à (2.1), ce qui la rend plus difficile à optimiser. De plus le modèle n'est pas valide sur toute l'image : dans les zones d'occultations par exemple, la valeur de l'attache aux données, à la disparité vraie, prend souvent des valeurs équivalentes à un mauvais appariement. Pour ces raisons, le problème de reconstruction est nettement plus difficile que le problème de dé-bruitage présenté précédemment. Ainsi, sélectionner le maximum de vraisemblance par pixel ne permet pas d'obtenir une initialisation correcte. Dans les expérimentations,



Figure 2.12 – Couples d’image de la base de test et la vérité de la carte de disparité dans la géométrie de l’image gauche. De gauche à droite et de haut en bas, le couple Tsukuba et les couples Venus, Cones et Teddy de la base Middlebury.

la carte de disparité est initialisée par bloc de 11 par 11 pixels en minimisant, sur chaque bloc, la somme des attaches aux données. Tous d’abord, nous allons présenter les jeux de données utilisés pour l’évaluation (Section 3.1). Les expérimentations se décomposent en deux parties suivant que la fonction r de la régularisation est une fonction convexe (Section 3.2) ou une fonction de Potts lissée (Section 3.3).

3.1 Les jeux de données

3.1.1 Base de Middlebury

La base de test Middlebury est une collection de couples stéréoscopiques. Elle présente plus particulièrement des scènes d’intérieur dont les conditions d’illumination sont bien maîtrisées. Les couples sont fournis dans la géométrie épipolaire rectifiée avec la disparité zéro pour les objets les plus éloignés. Pour chaque couple, une vérité de carte de disparité est aussi fournie dans les deux points de vue du couple. Cette base a été enrichie d’année en année avec des séries d’images de plus en plus complexes.

Le critère d’évaluation est le pourcentage des pixels à plus d’une disparité d’écart à la vérité. Ce critère est calculé dans deux cas : sur tous les pixels ("occ") ou sur les pixels non-occultés seulement ("non-occ"). Pour l’évaluation, nous utilisons les 4 images de la deuxième version de Middlebury : Tsukuba, Venus, Cones, Teddy Figure 2.12. La fonction d’attache aux données est le carré de la différence d’intensité des points appariés dont les valeurs sont tronquées au delà d’un écart de 18 d’intensité.

3.1.2 Kitti

La base de données KITTI est composée de données issues d’un véhicule d’acquisition mobile. Nous nous intéressons à la partie stéréoscopique de cette base. Il s’agit de couples stéréoscopiques de scènes routières en géométrie épipolaire rectifiée. La disparité zéro correspond aux objets à l’infini. Il existe trois réalisations de la base millésimées 2009, 2012 et 2015, que nous appellerons donc KITTI09, KITTI12 et KITTI15.

La base KITTI12 est divisée en un jeu de test et un jeu d’évaluation. Le jeu de test est composé des images en couleur gauche et droite, des vérités de cartes de disparité dans la géométrie gauche et la géométrie droite selon deux versions : la carte "occ" dite "occultée"

composée de tous les points et la carte "non-oc" dite "non occultée" limitée aux points qui sont visibles dans l'autre image. Ces cartes sont obtenues avec un lidar et sont assez peu denses. Le jeu d'évaluation est lui aussi composé des couples d'images en couleurs sans les vérités. Le critère d'évaluation proposé est le pourcentage de disparités à plus de 3 de la disparité vraie. Les images souffrent d'un certain nombre de défauts qui rend la couleur difficilement exploitable. En particulier, il y a des aberrations chromatiques dues aux décalages entre canaux qui rendent la correspondance colorée difficile. Ces défauts sont illustrés dans la Figure 2.13.

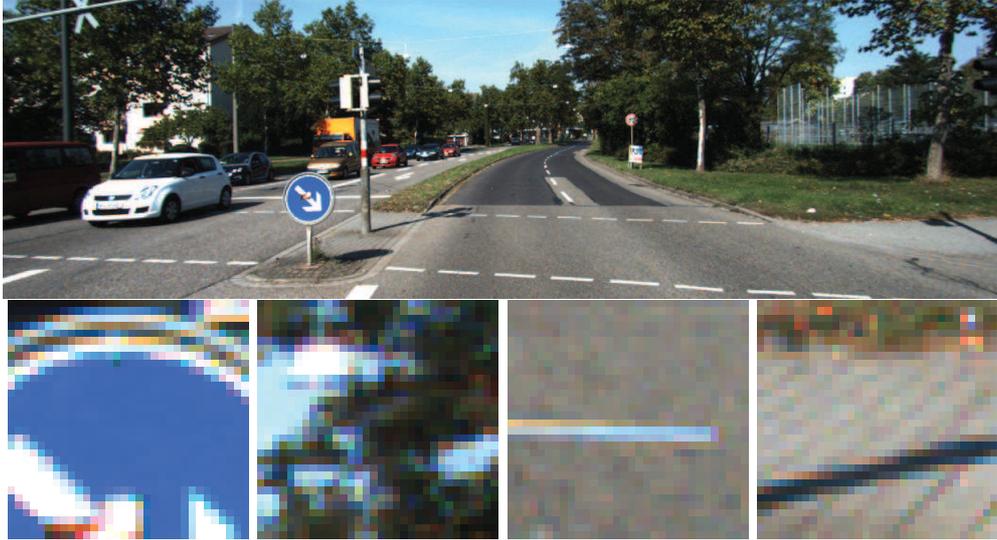


Figure 2.13 – Image gauche du couple de la base KITTI15 et agrandissements de détails de l'image. Les aberrations chromatiques sont particulièrement visibles à la frontière de zones très claires ou très foncées.

Plus généralement, les prises de vue en extérieur sont moins facilement maîtrisables et sont soumises à plus de sources de variabilité. Ainsi la plage des intensités lumineuses peut être très variable de sorte que de nombreuses valeurs peuvent saturer. On observe une bonne correspondance des intensités dans ces zones qui s'explique en partie par la faiblesse de la dynamique de l'image. Certaines surfaces ont des comportements partiellement spéculaires, ainsi le bitume a tendance à refléter le soleil. Les voitures sont elles-aussi des objets difficiles à traiter : les vitres sont à la fois transparentes et spéculaires et leur peinture est elle-aussi très spéculaire. Cela peut conduire à des mauvaises correspondances en intensité entre les points homologues. Cette mauvaise correspondance est illustrée Figure 2.14. Dans le couple d'images choisi, le soleil se reflète différemment sur la route entre les deux images. Il en résulte une mauvaise correspondance entre les intensités des deux images (de plus de 10 niveaux de gris).

3.2 Cas où la fonction r de la régularisation est la valeur absolue

Les différents schémas d'optimisation, présentés dans la Section 4 du Chapitre 1, permettent d'approcher la solution globale du problème de la reconstruction. Dans le cas où la fonction r est la valeur absolue, la solution globale peut être trouvée par la méthode d'Ishikawa. Dans ce cas il est possible de calculer l'écart d'énergie entre les solutions approchées et le minimum global. Cinq schémas d'optimisation sont testés (α -Expansion, Alterné-C, Alterné-L, Alterné-LQ et Saut). Alterné-L est limité à l'intervalle $[-3; 3]$ et Alterné-LQ est en plus quantifié d'un facteur 3. Le Saut est lui aussi limité à l'intervalle $[-3; 3]$. Les sous-problèmes de fusion sont optimisés par GC de manière exacte.

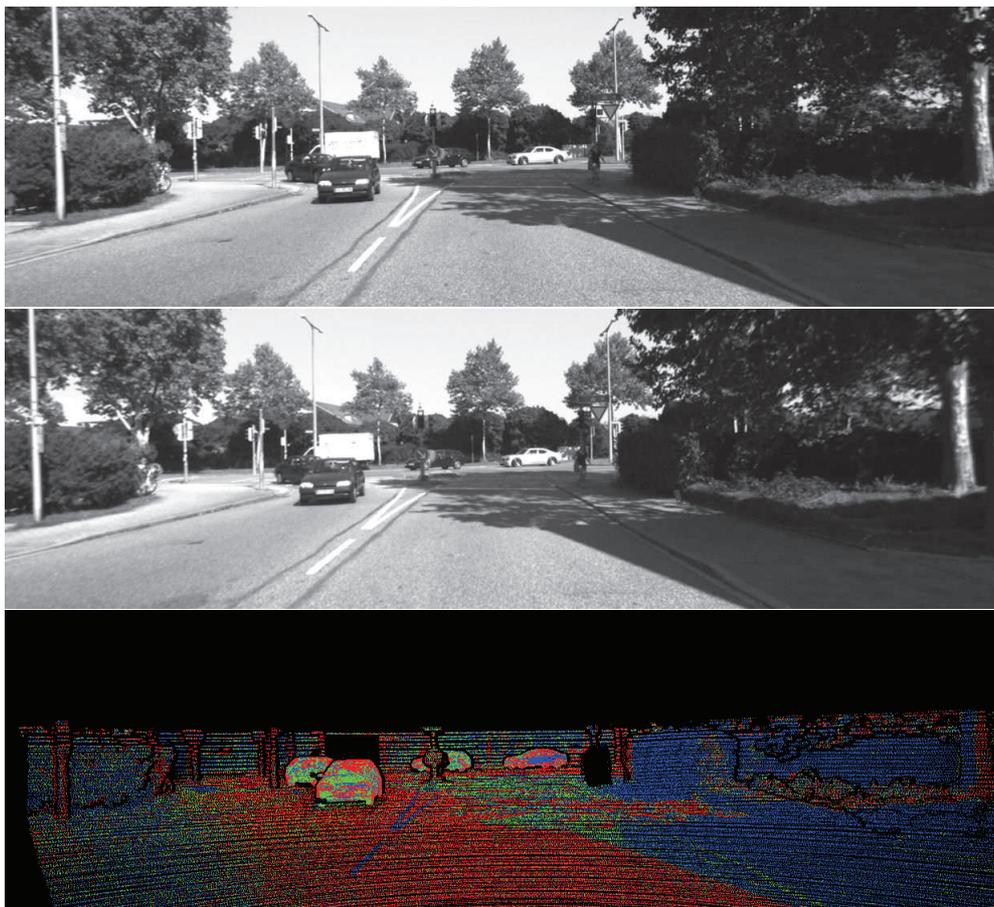


Figure 2.14 – Exemple d’un couple d’image de la base KITTI et de l’écart d’intensité entre les pixels de l’image gauche et de l’image droite pour un appariement à la disparité gauche vraie. Les écarts inférieurs à 5, inférieurs à 10 et supérieurs à 10 sont représentés respectivement en bleu, vert et rouge. Une large proportion des pixels est dans la dernière catégorie. Cela rend difficile une approche d’appariement par pixels. Dans cette image, cette zone correspond principalement à une différence de réflexion spéculaire sur la route. Nous remarquons aussi que les écarts inférieurs à 5 correspondent à des pixels saturés ou proches de la saturation. Ainsi, ces bonnes correspondances sont dues en partie à la faiblesse de la dynamique de l’image dans ces zones.

3.2.1 Sur des scènes d’intérieur de la base Middlebury

La fonction d’attache aux données est la somme, selon les trois canaux couleur, des écarts après un carré tronqué au-delà d’un écart de 18. Les résultats pour quatre couples d’images test sont données dans le Tableau 2.5. En terme de valeur d’énergie finale, le schéma Saut est le plus éloigné du minimum global de l’énergie, par rapport aux autres schémas. Cela s’explique par le fait que l’attache aux données, en reconstruction, est peu régulière et qu’une recherche de solution de proche en proche rencontre rapidement un minimum local. Les autres schémas atteignent des valeurs minimales proches, toutefois le schéma Alterné-LQ minimise un peu moins l’énergie et Alterné-C atteint l’énergie la plus basse sauf pour l’image Teddy.

En termes de temps de convergence, le schéma Alterné-C est entre deux et trois fois plus lent qu’ α -Expansion. Le temps de convergence du schéma Alterné-L est un peu plus lent que celui d’ α -Expansion (de quelques dixièmes de seconde et est même plus rapide pour l’image Cones), pour une énergie finale légèrement plus basse. Il est à noter que la méthode optimale d’Ishikawa s’exécute dans un temps relativement court (de l’ordre de la dizaine de seconde). Cela vient du fait que la taille des images et le nombre de disparités est relativement faible. En termes de taux d’erreur, les performances des différents schémas sont proches pour α -Expansion,

Méthode	Énergie	temps	err.M	Énergie	temps	err.M
	Tsukuba			Venus		
Ishikawa	7 369 287	3,20 s	2.97%	16 506 323	12,84 s	3.39%
α -Expansion	+25 007	3,50 s	2.82%	+37 184	16,58 s	3.41%
Alterné-C	+19 890	6,98 s	2.86%	+34 198	42,54 s	3.42%
Alterné-L	+20 882	3,86 s	2.86%	+35 197	19,19 s	3.41%
Alterné-LQ	+124 773	3,02 s	3.11%	+57 260	7,83 s	3.54%
Saut	+505 753	2,42 s	4.24%	+661 202	7,38 s	4.84%
Méthode	Cones			Teddy		
Ishikawa	38 868 511	33,60 s	5.77%	29 580 531	29,39 s	8.68%
α -Expansion	+172 520	29,51 s	5.81%	+396 782	37,09 s	10.48%
Alterné-C	+152 830	50,92 s	5.82%	+400 489	52,83 s	10.20%
Alterné-L	+164 898	20,62 s	5.82%	+397 795	37,50 s	10.19%
Alterné-LQ	+326 529	9,26 s	6.13%	+465 300	12,35 s	11.77%
Saut	+5 267 073	7,51 s	9.68%	+4 196 985	5,84 s	12.47%

Tableau 2.5 – Comparaison des méthodes d’optimisation sur quatre couples d’images test de la base Middlebury dans le cas d’une régularisation avec la fonction valeur absolue. Les énergies finales des différents schémas d’optimisation sont données en écart à l’énergie minimale.

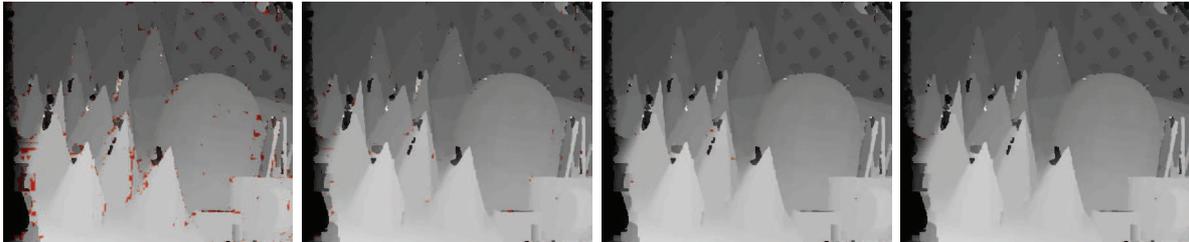


Figure 2.15 – Carte de disparité obtenue sur le couple Cones au cours des itérations 2, 3, 4 et 5 de Alterné-C. Les changements par rapport à l’itération précédente sont en rouge.

Alterné-C et Alterné-L (des différences de quelques centièmes de pourcent jusqu’à 0,3% dans le couple Teddy). Il est plus élevé pour le schéma-LQ de 0.3% à plus de 1.5% dans le cas de l’image Teddy. La solution optimale donne le meilleur score.

Pour compléter l’analyse expérimentale, comme dans la Section 1.4, nous nous intéressons à l’évolution des valeurs d’énergie dans le temps pour les différents schémas. Dans la Figure 2.15, nous pouvons voir que peu de pixels sont remis à jour à chaque itération et que ces derniers sont principalement dans les zones occultées. L’évolution de l’énergie au cours du temps pour le couple Cones est donnée dans le Tableau 2.16. Il est à noter que l’échelle verticale est logarithmique. Dans l’ordre chronologique, les méthodes qui permettent d’obtenir l’énergie la plus basse sont Alterné-LQ avant 4s, Alterné-L jusqu’à 17s et Alterné-C au-delà. Il apparaît que les schémas α -Expansion et Alterné-L sont très proches et que l’addition de quelques fusions par saut permettent d’améliorer les résultats pour un surcoût en temps de calcul très faible.

3.2.2 Sur des images de la base KITTI

Nous estimons la carte de disparité pour quatre couples de la base KITTI : les couples 0, 4, 9 et 20. Le couple 4 est particulièrement difficile du fait de reflets sur la route. L’attache aux données est obtenue en calculant le Census sur une fenêtre 5 par 5 pixels, tronqué au-delà de la valeur 5. Ces coûts sont agrégés par CBCA, décrit dans la Section 2.2.2, réalisé 4 fois. Cette opération lisse localement l’attache aux données à disparité constante. Nous prenons $\lambda = 1.2$, et les énergies sont multipliées par un facteur 10 pour donner des valeurs entières. Nous appliquons les mêmes schémas d’optimisation que dans la section précédente. Les résultats sont donnés dans le Tableau 2.6. Tout d’abord, les différents schémas d’optimisation convergent plus vite que la

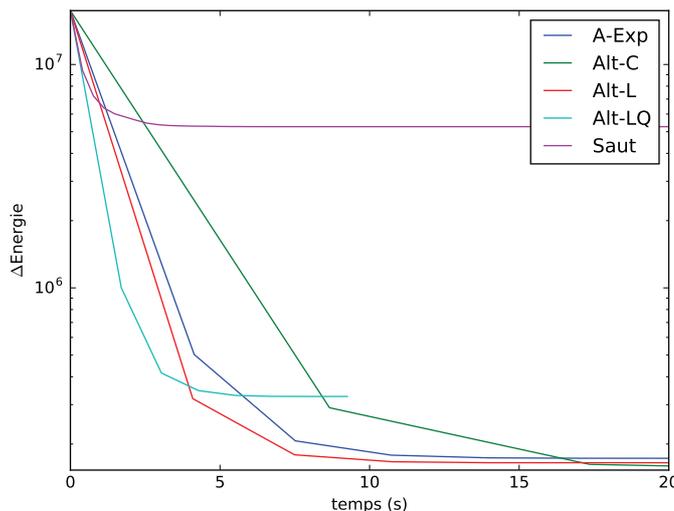


Figure 2.16 – Évolution de l'énergie au cours de l'optimisation pour la reconstruction du couple d'image Cones.

méthode d'Ishikawa, sauf avec le schéma alterné. Ainsi les durées de convergences des schémas d'optimisation sont moins sensibles à l'augmentation de la taille des images et au nombre de disparités par rapport à la méthode optimale d'Ishikawa. Le schéma Saut converge rapidement mais l'énergie est très éloignée du minimum global. Il en résulte un taux d'erreurs plus élevé de 4% à 10% par rapport à la solution minimale. Les schémas α -Expansion, Alterné-C, Alterné-L donnent des énergies finales proches et des taux d'erreur comparables. Le schéma Alterné-LQ donne des énergies plus élevées et un taux d'erreur supérieur de 0,3 à 0,5% par rapport à α -Expansion, pour un gain sur le temps de convergence d'un facteur 2 à 3. Ce facteur est à rapprocher du taux d'échantillonnage des fusions par expansion qui est de 3.

Méthode	Énergie	temps	err.K	Énergie	temps	err.K
	Couple 0			Couple 4		
Ishikawa	159 286 124	270 s	13.5%	161 891 592	395 s	32.4%
α -Expansion	+118 115	131 s	14.7 %	+127 363	171 s	39.4 %
Alterné-C	+113 801	288 s	14.5 %	+130 948	450 s	39.8 %
Alterné-L	+118 521	127 s	14.5 %	+127 213	175 s	39.5 %
Alterné-LQ	+586 055	54 s	14.8 %	+502 379	42 s	38.9 %
Saut	+3 877 381	14 s	23.5 %	+3 876 885	15 s	39.9 %
Méthode	Couple 9			Couple 20		
Ishikawa	175 465 281	235 s	7.4%	174 719 606	622 s	17.6%
α -Expansion	+21 448	95 s	7.5%	+31 764	83 s	18.8%
Alterné-C	+19 551	212 s	7.4%	+30 542	249 s	18.9%
Alterné-L	+21 041	91 s	7.4%	+31 489	102 s	18.8%
Alterné-LQ	+198 785	25 s	8.1%	+238 583	40 s	19.1%
Saut	+2 485 268	9 s	11.6%	+2 711 917	11 s	21.1%

Tableau 2.6 – Comparaison des méthodes d'optimisation sur quatre couples de la base KITTI15 dans le cas d'une régularisation par la valeur absolue. Les énergies finales des différents schémas d'optimisation sont données en écart à l'énergie minimale.

La Figure 2.17 présente l'évolution de l'énergie au cours du temps. Lorsque la première itération d' α -Expansion est achevée, son énergie est inférieure au minimum qu'atteindra le schéma Alterné-LQ. Le schéma Alterné-L n'apporte que peu d'améliorations par rapport à α -Expansion en terme d'énergie, contrairement à l'expérimentation sur la base Middlebury, pour un temps d'exécution un peu plus long. Le temps d'exécution d'un cycle d'Alterné-C est plus du double de

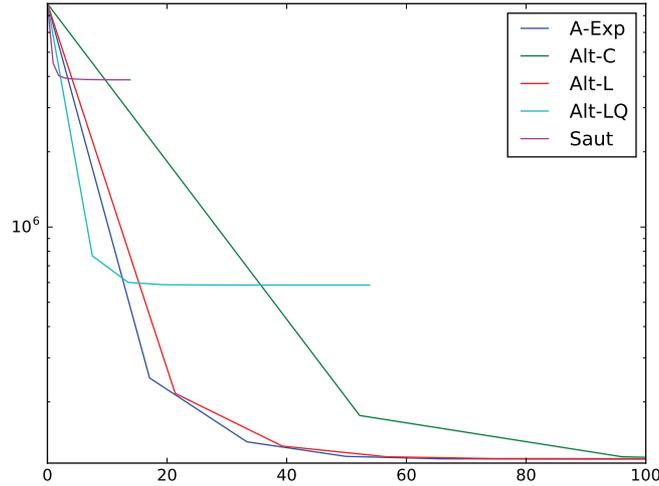


Figure 2.17 – Évolution de l'énergie au cours de l'optimisation pour le couple 0 de la base KITTI15.

celui d'Alterné-L (environ 50s contre 22s). A partir de 100s, les schémas ALterné-C, Alterné-L et α -Expansion atteignent des énergies proches.

En conclusion, le schéma Saut ne permet pas d'obtenir une bonne approximation de la solution pour le problème de reconstruction stéréoscopique. Ainsi, ce schéma ne sera plus utilisé par la suite pour le problème de la reconstruction. Plus généralement, la combinaison de la fusion par saut avec d'autres fusions n'apporte que peu d'améliorations pour la reconstruction contrairement à ce que l'on a pu voir dans le cas du dé-bruitage. En effet, les performances des schémas Alterné-C et Alterné-L sont comparables à celles d' α -Expansion. Le schéma Alterné-LQ permet de converger plus rapidement pour un minimum plus élevé. Ainsi α -Expansion et Alterné-LQ sont des bons compromis entre la minimisation de la valeur de l'énergie et temps de convergence.

3.2.3 Cas de mauvaise reconstruction avec α -Expansion

Dans les résultats précédents, nous avons vu que la solution obtenue par α -Expansion est, en valeur d'énergie, très proche de la solution obtenue par la méthode Ishikawa et que les différences entre les solutions sont localisées dans les zones occultées. Toutefois, il arrive que la solution approchée d' α -Expansion ne soit pas satisfaisante. Prenons un modèle où un filtre de Sobel horizontal est appliqué à l'intensité des images :

$$\mathbf{S}_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \otimes \mathbf{I} \quad (2.27)$$

où \otimes est l'opérateur de convolution. L'attache aux données est une fonction carrée tronquée de la différence de l'intensité après filtrage, \mathbf{S}_x . L'objectif est d'avoir une attache aux données par pixel robuste à l'ajout d'une constante entre l'image I^1 et l'image I^2 sur le support du filtre. Les résultats de reconstruction des deux méthodes sont donnés dans la Figure 2.18. La solution obtenue par α -Expansion consiste en grande partie en un plan à disparité 0 et quelques objets reconstruits. L'explication que nous avançons pour expliquer ce mauvais résultat est que l'attache aux données choisie est moins robuste au changement d'intensité. Le filtre de Sobel a donc tendance à rendre le problème plus ambigu. La régularisation joue alors un rôle plus important du fait du manque d'information donnée par le terme d'attache aux données, de sorte que le problème devient souvent une interpolation dans les zones mal déterminées comme

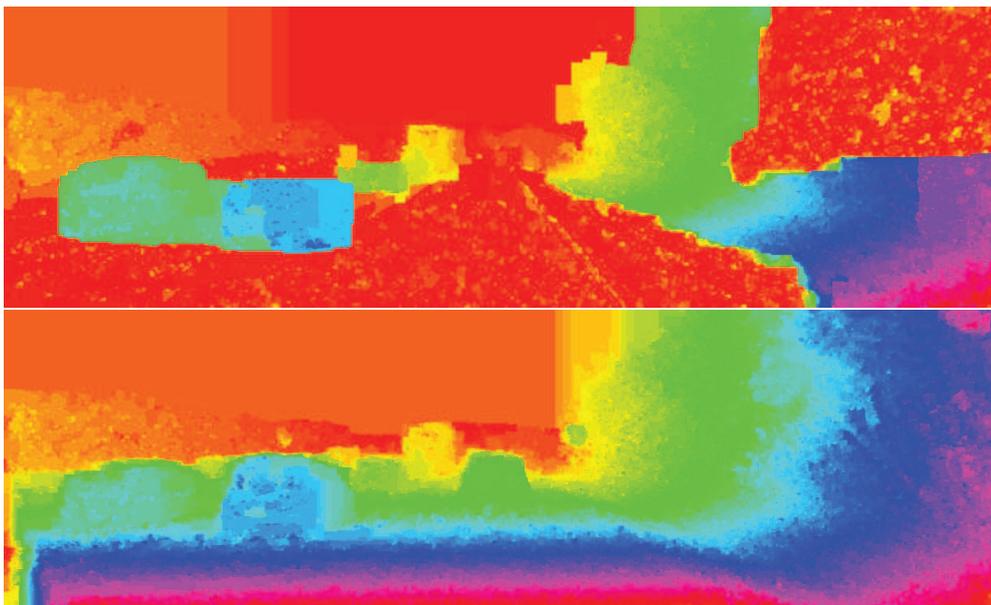


Figure 2.18 – Image de disparité du couple 199 de la base KITTI15 optimisée par α -Expansion et par la méthode d’Ishikawa. La solution α -Expansion est très éloignée de la solution exacte du problème. De grandes portions de la solution sont planes et ne décrivent pas la scène qui a été acquise.

la chaussée. Cela est renforcé par le fait que l’a priori fronto-parallèle n’est pas valide sur ces zones.

3.3 Cas où la fonction r est une fonction de Potts lissée

L’observation des distributions des écarts de disparités entre voisins amène à choisir r égal à une fonction de Potts lissée (2.19), avec le paramètre $2P1 \leq P2$, de sorte que la fonction présente une cuvette convexe. Dans ce cas, ni les fusions par expansion ni les fusions par saut n’ont d’écritures sous-modulaires et des méthodes approchées doivent être mises en œuvre. L’utilisation d’une telle fonction de régularisation est l’occasion de comparer les performances des trois méthodes de résolution approchée des fusions : la fusion relaxée, QPBO et QPBO-P.

3.3.1 Sur des scènes d’intérieur de la base Middlebury

Sur la base Middlebury, l’attache aux données reste la même que dans la Section 3.2.1. Les paramètres de la fonction de régularisation sont $(P1, P2) = (100, 800)$. Les résultats sont montrés dans le Tableau 2.7.

Optimiser les sous-problèmes avec QPBO ou QPBO-P donne la même énergie finale et le même taux d’erreur, sauf dans le cas du couple Cones. Le temps de convergence de QPBO-P est d’au moins 70% supérieur par rapport à celui de QPBO. La méthode QPBO permet d’atteindre des minima inférieurs à ceux de la fusion relaxée pour un temps de convergence 70% à 2 fois plus long.

Les schémas Alterné-C et Alterné-L donnent des résultats similaires. Ce sont ces deux schémas qui atteignent les énergies les plus basses. Le minimum atteint par α -Expansion est plus élevé mais reste inférieur à celui du schéma Alterné-LQ. En termes de taux d’erreur, les schémas α -Expansion, Alterné-C et Alterné-L sont comparables. Les taux d’erreur d’Alterné-Q sont supérieurs de 0.3% à 1%.

En conclusion, le choix de la méthode d’optimisation de fusion binaire n’a pas beaucoup d’influence sur le taux d’erreur. Les énergies les plus basses atteintes par QPBO et QPBO-P

Schéma	relaxé			QPBO			QPBO-P		
	Énergie	t	err.M	Énergie	t	err.M	Énergie	t	err.M
Schéma	Tsukuba								
α -Exp	135 614	4,7 s	2.67%	102 129	9,3 s	2.61%	101 720	17,0 s	2.61%
Alt-C	25 082	67,2 s	2.68%	368	51,0 s	2.70%	0	88,1 s	2.70%
Alt-L	29 498	35,4 s	2.68%	6040	25,5 s	2.70%	6040	46,3 s	2.70%
Alt-LQ	141 580	10,0 s	3.58%	112 940	16,0 s	3.55%	112 940	27,1 s	3.55%
Schéma	Venus								
α -Exp	57 306	9,7 s	2.53%	43 617	16,6 s	2.53%	43 617	31,9 s	2.53%
Alt-C	7421	32,9 s	2.57%	0	51,8 s	2.58%	0	103,1 s	2.58%
Alt-L	7421	14,7 s	2.57%	-26	24,0 s	2.58%	-26	44,2 s	2.58%
Alt-LQ	86 430	12,0 s	2.83%	68 807	17,4 s	2.83%	68 807	29,5 s	2.83%
Schéma	Cones								
α -Exp	249 572	57,6 s	4.55%	176 682	97,6 s	4.56%	175 159	198,4 s	4.53%
Alt-C	33 354	100,3 s	4.53%	7634	116,8 s	4.53%	0	223,6 s	4.50%
Alt-L	37 188	41,1 s	4.53%	8625	95,6 s	4.53%	991	190,6 s	4.51%
Alt-LQ	542 879	16,2 s	5.55%	454 230	23,1 s	5.56%	454 576	47,0 s	5.56%
Schéma	Teddy								
α -Exp	264 286	68,3 s	11.27%	219 753	126,3 s	11.30%	219 753	236,1 s	11.30%
Alt-C	59 527	122,1 s	11.30%	0	201,2 s	11.26%	0	370,3 s	11.26%
Alt-L	75 906	52,6 s	11.30%	36 247	110,1 s	11.27%	36 247	205,9 s	11.27%
Alt-LQ	425 461	18,4 s	11.57%	332 754	24,6 s	11.51%	332 754	47,1 s	11.51%

Tableau 2.7 – Comparaison des énergies, temps de convergence et taux d’erreur pour 4 images, 5 schémas d’optimisation et 3 méthodes d’optimisation. Les valeurs d’énergie sont données en écart à l’énergie Alt-L par QPBO-P. Le taux d’erreur est le pourcentage de pixels non-occultés à plus d’une disparité par rapport la disparité vraie (err.M).

sont obtenues au prix d’un temps de convergence plus long.

3.3.2 Sur la base KITTI

L’attache aux données est obtenue en calculant le Census sur une fenêtre 5 par 5 puis en appliquant 4 fois la CBCA, décrit dans la Section 2.2.2. Les résultats sont montrés dans le Tableau 2.8. Le temps de convergence est 4 fois plus long pour QPBO et 8 fois plus long pour QPBO-P par rapport à la fusion relaxée. Les méthodes QPBO et QPBO-P donnent des énergies très proches. Les taux d’erreur varient très peu selon l’utilisation de la fusion relaxée ou de QPBO.

Contrairement aux couples de la base Middlebury, le taux d’erreur avec les schémas Alterné-C et Alterné-L est plus bas de 0.3% à 1.5% comparé à α -Expansion, sauf dans le cas du couple Teddy. Le taux d’erreur avec le schéma Alterné-LQ est, selon les cas, plus proche de celui d’ α -Expansion ou de celui d’Alterné-C.

4 Conclusion

Nous avons illustré comment la formulation par coupure de graphe peut être appliquée à différents problèmes image. Nous avons présenté, en particulier, un cas simple de dé-bruitage d’image. Lorsque l’attache aux données et la fonction de régularisation est la valeur absolue, nous avons observé que le schéma α -Expansion converge vers le minimum global sur ce problème. Ce n’est plus le cas lorsque la fonction de régularisation est la valeur absolue lissée, même lorsque les fusions expansions sont minimisées exactement. Il en est de même lorsque l’attache aux données n’est pas convexe. Le dé-bruitage nous a permis d’observer l’influence de la forme de la

Schéma	relaxé			QPBO			QPBO-P		
	Énergie	t	err.K	Énergie	t	err.K	Énergie	t	err.K
Schéma	Couple 0								
α -Exp	764 613	106 s	11.35%	693 171	451 s	11.2%	692 936	918 s	10.08%
Alt-C	17 402	1186 s	10.21%	0	4710 s	10.08%	0	9308 s	10.32%
Alt-L	14 145	439 s	10.21%	6988	1476 s	10.32%	6988	2630 s	11.23%
Alt-LQ	882 696	151 s	11.46%	750 282	481 s	11.23%	750 282	740 s	17.33%
Schéma	Couple 4								
α -Exp	656 260	159 s	34.0%	640 843	412 s	34.7%	632 871	956 s	34.7%
Alt-C	44 366	2068 s	32.5%	0	8503 s	32.3%	0	15 879 s	32.3%
Alt-L	42 260	767 s	32.5%	8132	2787 s	32.3%	8132	6537 s	32.3%
Alt-LQ	544 094	208 s	32.7%	458 689	681 s	32.5%	458 689	1875 s	32.5%
Schéma	Couple 9								
α -Exp	690 062	88 s	5.02%	614 827	413 s	5.04%	614 827	828 s	5.0%
Alt-C	9601	1048 s	4.76%	0	4000 s	4.81%	0	6837 s	4.8%
Alt-L	14 478	476 s	4.74%	16 680	1769 s	4.78%	16 680	3113 s	4.8%
Alt-LQ	462 688	151 s	5.87%	366 285	496 s	5.79%	366 516	899 s	5.8%
Schéma	Couple 20								
α -Exp	789 280	122 s	16.31%	691 393	448 s	16.42%	691 325	924 s	16.4%
Alt-C	27 116	2572 s	16.36%	-269	10 947 s	16.37%	0	18 738 s	16.4%
Alt-L	41 002	1021 s	16.51%	17 641	3385 s	16.55%	17 910	5137 s	16.6%
Alt-LQ	544 127	214 s	16.78%	437 513	617 s	16.76%	437 513	545 s	16.8%

Tableau 2.8 – Comparaison des énergies, temps de convergence et taux d’erreur pour 4 images, 5 schémas d’optimisation et 3 méthodes d’optimisation. Les valeurs d’énergie sont données en écart à l’énergie Alt-L par QPBO-P. L’erreur est le pourcentage de pixels à plus de trois disparités de la carte de disparité "non-occ" (err.K).

fonction de régularisation sur le résultat et nous a permis de justifier l’utilisation d’une fonction de régularisation convexe autour de zéro et concave pour des fortes valeurs. Nous avons remarqué aussi que les performances des schémas de fusions dépendent moins de la sous-modularité des sous-problèmes de fusion, qui implique des approximations lors de la résolution, que de la capacité des fusions à proposer des hypothèses cohérentes avec la solution globale.

Nous avons ensuite traité le problème de la reconstruction stéréoscopique par coupure de graphe. Nous avons vu la difficulté d’avoir un modèle valide sur toute l’image du fait de la présence de parties occultées. Parmi les représentations possibles, nous avons choisi l’approche par carte de disparité. L’énergie de ce problème est similaire à l’énergie définie pour le dé-bruitage à la différence de l’attache aux données qui possède de nombreux minima locaux. Ainsi le minimum atteint par les schémas d’optimisation approchés sont plus éloignés du minimum global. Du fait de nombreux minima locaux, les fusions par saut sont moins efficaces pour optimiser le problème que dans le cas du dé-bruitage. Il apparaît qu’utiliser des méthodes d’optimisation raffinées comme QPBO et QPBO-P pour traiter les sous-problèmes de fusion est peu intéressant sur les énergies que nous avons présentées dans ce chapitre, du fait du temps de convergence plus long. En effet, la valeur de l’énergie finale atteinte dépend plus du schéma d’optimisation que de la méthode de résolution des sous-problèmes de fusion.

Chapitre 3

Programmation dynamique appliquée à la reconstruction stéréoscopique

La programmation dynamique (*Dynamic Programming*, DP) est une méthode d'optimisation discrète qui traite des problèmes combinatoires par énumération implicite des solutions. Des ensembles de solutions sont retenues ou rejetés sans que ces solutions soient construites en totalité. La DP exploite le principe d'optimalité de Bellman pour limiter l'exploration des solutions. Ce principe s'énonce simplement comme : un chemin optimal est formé de sous-chemins optimaux. Si (C) est un chemin optimal de A à B et si C appartient à (C) alors les sous-chemin de A à C et de C à B sont optimaux. Ainsi tout chemin contenant un chemin non-optimal peut être retiré de l'ensemble des chemins à explorer. Dans la Section 1, nous verrons comment utiliser la programmation dynamique pour minimiser un champ de Markov 1D du premier ordre. Puis dans la Section 2, nous verrons comment ces méthodes peuvent être appliquées à la reconstruction. Dans la Section 3, nous comparons les résultats des différentes méthodes de reconstruction.

1 Optimiser un champ de Markov du premier ordre par DP

Un champ de Markov 1D peut décrire un problème résultant de l'application du maximum a posteriori (MAP) sur une ligne ou une colonne de l'image. Le problème est formulé soit comme une maximisation d'un produit de probabilité, soit comme la minimisation d'une somme. Dans la Section 1.1, nous présenterons une méthode de résolution d'un tel problème, puis dans la Section 1.2, nous verrons comment cette méthode peut être étendue à des champs qui ne sont pas 1D.

1.1 Optimisation d'un champ de Markov 1D du premier ordre par DP

Les champ de Markov 1D peuvent être représentés par un graphe simple : les nœuds sont ordonnés et chaque nœud possède un arc vers son prédécesseur et son suivant, s'ils existent, comme dans la Figure 3.1. On définit arbitrairement deux sens de parcours : le sens direct dans

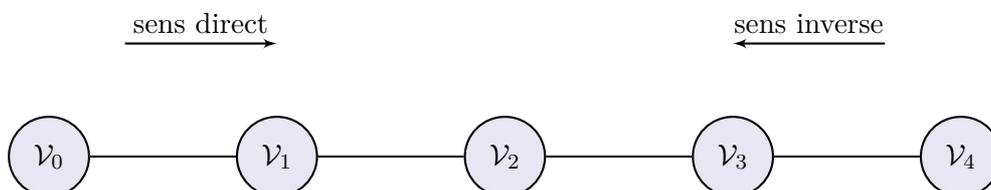


Figure 3.1 – Exemple de chaîne de Markov 1D avec 5 nœuds.

le sens des indices croissants et le sens inverse dans le sens des indices décroissants. Un tel champ peut être optimisé globalement par programmation dynamique.

1.1.1 Maximisation de la probabilité a posteriori

Avec la formulation bayésienne, en appliquant le principe du MAP, on exprime le problème comme la recherche de \mathbf{l}^{MAP} qui minimise la probabilité suivante :

$$\mathbf{l}^{MAP} = \underset{\mathbf{l}}{\operatorname{argmax}} \mathbb{P}(\mathbf{l}) \quad (3.1)$$

avec

$$\mathbb{P}(\mathbf{l}) = \prod_{p=0}^n \mathbb{P}_p(l_p) \prod_{p=0}^{n-1} \mathbb{P}_{p,p+1}(l_p|l_{p+1}) \quad (3.2)$$

où \mathbb{P}_p est la probabilité du label l_p en p et $\mathbb{P}_{p,p+1}$ la probabilité jointe des labels l_p et l_{p+1} en p et $p+1$. Nous réécrivons le problème en fonction de $\mathbb{P}(\mathbf{l}|l_p=k)$ probabilité sachant que la variable l_p possède la valeur k et $\mathbb{P}(l_p=k)$,

$$l_p^{MAP} = \underset{k}{\operatorname{argmax}} \mathbb{P}(l_p=k) \max_{l_0, \dots, l_{p-1}, l_{p+1}, \dots, l_n} \mathbb{P}(\mathbf{l}|l_p=k) \quad (3.3)$$

Cette probabilité peut être décomposée en un produit de deux termes :

$$\mathbb{P}(\mathbf{l}|l_p=k) = \mathbb{P}_{p \rightarrow 0}(k) \mathbb{P}_{p \rightarrow n}(k) \quad (3.4)$$

avec

$$\mathbb{P}_{a \rightarrow b}(k) = \begin{cases} \prod_{p=a+1}^b \mathbb{P}(l_p) \mathbb{P}(l_p|l_{p-1}) & |l_a = k \quad \text{si } a < b \\ \prod_{p=b}^{a-1} \mathbb{P}(l_p) \mathbb{P}(l_p|l_{p+1}) & |l_a = k \quad \text{si } a > b \\ 1 & \text{si } a = b \end{cases} \quad (3.5)$$

Le premier terme de (3.4) représente les interactions avec ce qui se passe avant le pixel p et le deuxième terme les interactions avec ce qui se passe après. La maximisation de $\mathbb{P}(\mathbf{l}|l_p=k)$ peut se réécrire comme la maximisation d'une deuxième probabilité, sur un sous-ensemble de variables :

$$l_p^{MAP} = \underset{k}{\operatorname{argmax}} \mathbb{P}(l_p=k) \Phi_{p,0}(k) \Phi_{p,n}(k) \quad (3.6)$$

avec

$$\Phi_{a \rightarrow b}(k) = \begin{cases} \max_{l_i, i \in [a+1, b]} \mathbb{P}_{a \rightarrow b}(k) & \text{si } a < b \\ \max_{l_i, i \in [b, a-1]} \mathbb{P}_{a \rightarrow b}(k) & \text{si } a > b \\ 1 & \text{si } a = b \end{cases} \quad (3.7)$$

Les fonctions $\Phi_{a,b}(k)$ peuvent être calculées récursivement. Si $a < b$,

$$\begin{aligned} \Phi_{a \rightarrow b}(k) &= \max_{l_{a+1}, \dots, l_b} \mathbb{P}_{a \rightarrow b}(k) \\ &= \max_{l_{a+1}} \mathbb{P}(l_{a+1}|k) \mathbb{P}(l_{a+1}) \max_{l_{a+2}, \dots, l_b} \mathbb{P}_{a+1 \rightarrow b}(l_{a+1}) \\ &= \max_{l_{a+1}} \mathbb{P}(l_{a+1}|k) \mathbb{P}(l_{a+1}) \Phi_{a+1 \rightarrow b}(l_{a+1}) \end{aligned} \quad (3.8)$$

Cette opération de récursion est illustrée dans la Figure 3.2. De manière analogue, nous avons si $a > b$:

$$\Phi_{a \rightarrow b}(k) = \max_{l_{a-1}} \mathbb{P}(l_{a-1}|k) \mathbb{P}(l_{a-1}) \Phi_{a-1 \rightarrow b}(l_{a-1}) \quad (3.9)$$

Ainsi la résolution du problème se décompose en trois étapes :

- Calculer les valeurs $\Phi_{0 \rightarrow p}(k)$ pour tout p en appliquant la récursion (3.9) à partir de la position 0.
- Calculer les valeurs $\Phi_{p \rightarrow n}(k)$ pour tout p en appliquant la récursion (3.8) à partir de la position n .
- Sélectionner l'argument du maximum de (3.6) pour chaque pixel.

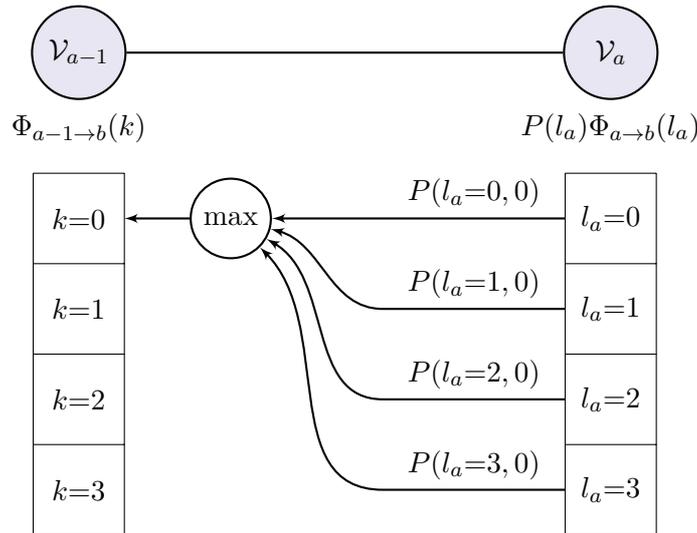


Figure 3.2 – Exemple de calcul d’une valeur Φ . La valeur $\Phi_{a-1 \rightarrow b}(0)$ est obtenue par calcul du maximum selon l_a du produit de $P(l_a)\Phi_{a \rightarrow b}(l_a)$ par $P(l_a, 0)$.

1.1.2 Minimisation d’une énergie

En appliquant le $-\log$ sur la probabilité (3.2), le problème revient à minimiser l’énergie :

$$E(\mathbf{l}) = \sum_{p=0}^n D_p(l_p) + \sum_{p=0}^{n-1} R_{p,p+1}(l_p, l_{p+1}) \quad (3.10)$$

De manière analogue à (3.6), nous réécrivons le problème comme la somme de trois termes :

$$l_p^{MAP} = \underset{k}{\operatorname{argmin}} D_p(k) + Acc_p^{\rightarrow}(k) + Acc_p^{\leftarrow}(k) \quad (3.11)$$

où les valeurs accumulées Acc sont calculées à l’aide de la règle de récursion suivante :

$$\begin{aligned} Acc_0^{\leftarrow}(k) &= 0 \\ Acc_p^{\leftarrow}(k) &= \min_{l_{p-1}} R_{p,p-1}(k, l_{p-1}) + D_{p-1}(l_{p-1}) + Acc_{p-1}^{\leftarrow}(l_{p-1}) \\ Acc_n^{\rightarrow}(k) &= 0 \\ Acc_p^{\rightarrow}(k) &= \min_{l_{p+1}} R_{p,p-1}(k, l_{p+1}) + D_{p+1}(l_{p+1}) + Acc_{p+1}^{\rightarrow}(l_{p+1}) \end{aligned} \quad (3.12)$$

Un exemple de minimisation d’une énergie est donné dans le Tableau 3.1. On note que la somme des termes de D_p et des valeurs accumulées de (3.11) peut être interprétée comme l’énergie minimale sous la contrainte $l_p = k$:

$$l_p^{MAP} = \underset{k}{\operatorname{argmin}} \min_{\mathbf{l}} E(\mathbf{l} | l_p = k) \quad (3.13)$$

1.2 Application à des champs non-1D

Nous considérons un champ de Markov du premier ordre décrit par le graphe $G = (\mathcal{V}, \mathcal{A})$. Pour tout arc du graphe $(\mathcal{V}_p, \mathcal{V}_q)$, son arcs frère $(\mathcal{V}_q, \mathcal{V}_p)$ appartient lui aussi au graphe. Si ce champ est acyclique alors le MAP peut être trouvé en temps polynomial par DP. Dans le cas contraire, des méthodes approchées peuvent être mises en œuvre.

label	Data					Acc \leftarrow				
	l_0	l_1	l_2	l_3	l_5	l_0	l_1	l_2	l_3	l_5
0	5	1	2	0	8	0	3	3	5	5
1	6	0	4	2	4	0	2	2	6	6
2	1	2	5	3	3	0	1	3	5	7
3	0	4	0	5	0	0	0	4	4	8
	Acc \rightarrow					$E(\mathbf{l} l_p = k)$				
0	6	5	3	3	0	11	9	8	8	13
1	6	6	4	2	0	12	8	10	10	10
2	7	6	4	1	0	8	9	12	9	10
3	8	5	5	0	0	8	9	9	9	8

Tableau 3.1 – Minimisation d’une énergie définie par le terme Data soumis à la régularisation par la valeur absolue. Les valeurs $E(\mathbf{l}|l_p = k)$ sont obtenues par la somme des trois autres. Toutes les solutions qui sélectionnent des valeurs $E(\mathbf{l}|l_p = k) = 8$, en rouge, sont des solutions minimales.

1.2.1 Les champs acycliques

L’optimisation d’une probabilité ou d’une énergie peut être étendue à des champs non-1D, mais il est nécessaire que le graphe qui représente le champ soit acyclique. Dans le cas d’un graphe non-orienté, il s’agit d’une forêt, c’est à dire d’un ensemble d’arbres. Un arbre est tel qu’il est possible d’attribuer un numéro d’ordre à chaque pixel, de sorte que tout nœud ne possède qu’un seul lien vers un nœuds d’ordre inférieur, sauf un : ce nœud, appelé racine, n’en possède aucun. Différents graphes non-orientés acycliques ou avec cycles sont présentés dans la Figure 3.3.

1.2.2 Optimisation d’un graphe acyclique par propagation de croyance (BP)

La propagation de croyance (*Belief Propagation*, BP) peut être utilisée pour optimiser un champ de Markov 1D. Un message $m_{p \rightarrow q}$ est associé à tout arc $(\mathcal{V}_p, \mathcal{V}_q)$ du graphe. Les messages sont initialisés et mis à jour à l’aide de la règle de mise à jour suivante :

$$m_{p \rightarrow q}(k) \leftarrow \min_k D_p(k) + R_{pq}(k, d_q) + \sum_{(t,p) \in \mathcal{E}, k \neq q} m_{t \rightarrow p}(k) \quad (3.14)$$

Les message sont initialisés et mis à jour dans un certain ordre. Il existe de nombreuses variantes suivant l’initialisation et selon le choix de l’ordre dans lequel les variables sont mises à jour. Cette méthode converge vers le minimum global, lorsque le graphe est acyclique. Dans le cas d’un champ 1D, appliquer BP en mettant à jour les messages sur tout les nœuds dans le sens direct puis dans le sens inverse est équivalent à la méthode DP présentée dans la Partie 1.1.2. Dans le cas plus général où la chaîne est acyclique, l’arbre peut être optimisé globalement en appliquant deux passes de mise à jour (3.14) : la première passe parcourt les nœuds de la racine vers les feuilles puis la seconde passe parcourt depuis les feuilles vers la racine.

1.2.3 Décomposition en champs acycliques

Dans le cas où le graphe possède des cycles, son optimisation devient *NP*-difficile. Des méthodes approchées ont été appliquées. Certaines consistent à approcher le champ par un arbre en élaguant le graphe. C’est le cas par exemple avec SGM, décrit dans la Partie 2 ci-après. C’est aussi le cas de la méthode *Tree-reweighted Message Passing* introduite dans [Kolmogorov, 2006]. La méthode BP a aussi été étendue à ce cas. Avec la méthode classique, la présence de cycle peut empêcher l’algorithme de converger. Ainsi des heuristiques ont été proposées pour éviter cela, comme dans [Yedidia *et al.*, 2001]. De telles méthodes appliquées à la reconstruction

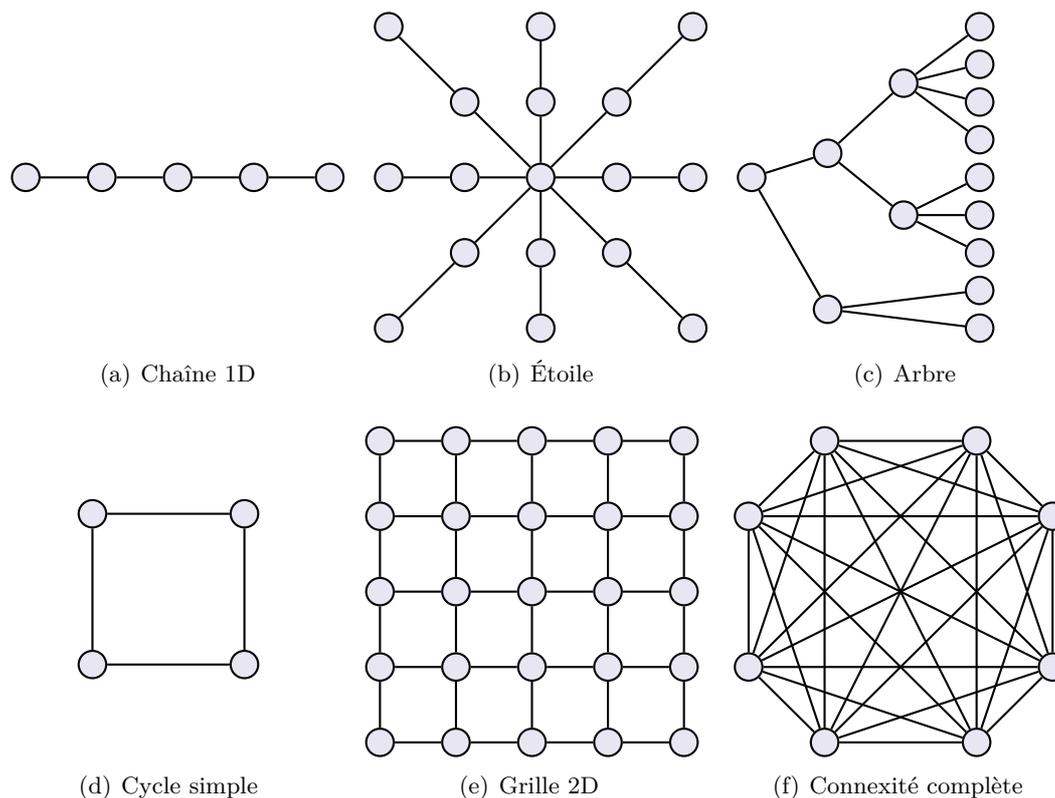


Figure 3.3 – Illustration de graphes acycliques sur la première ligne et de graphes avec cycles sur la seconde ligne. Seuls les problèmes qui s'écrivent sous la forme d'un graphe acyclique peuvent être traités par DP.

stéréoscopique permettent une résolution approchée prenant en compte les occultation comme dans [Sun *et al.*, 2005].

2 Appariement semi-global, *Semi-Global Matching* (SGM)

Les problèmes en image conduisent le plus souvent à une formulation par optimisation sur une grille 2D. Ce champ de Markov possède un grand nombre de cycles. Une approche pour traiter ce problème avec DP est de limiter les interactions entre voisins pour arriver à une optimisation 1D :

$$E^C(\mathbf{l}) = \sum_{p \in \mathbf{I} \cap \mathbf{C}} D_p(l_p) + \sum_{(p,q) \in \mathbf{N}' \cap \mathbf{C}^2} R_{pq}(l_p, l_q) \quad (3.15)$$

où \mathbf{C} est un alignement de pixel, \mathbf{I} est l'ensemble des pixels formant l'image et \mathbf{N}' l'ensemble des couples de pixels voisins en 4-connexité. Nous pouvons décomposer la grille 2D en alignements selon les ligne de l'image. L'inconvénient de cette approche est qu'aucune information n'est partagée entre les lignes. La cohérence de la solution selon les colonnes n'est pas assurée donnant lieu à des artefacts. De la même manière, la décomposition du problème peut être réalisée selon les colonne. Dans la Figure 3.5, nous présentons un exemple de résolution en décomposant selon les lignes et en décomposant selon les colonnes.

2.1 La méthode SGM

La méthode SGM, proposée par Hirschmüller [2008], consiste à ne pas prendre en compte l'information d'un seul alignement 1D, mais de plusieurs alignements 1D formant une étoile autour du pixel considéré. Soit ϱ un ensemble des directions dans l'image tel que si $\varrho \in \varrho$ alors

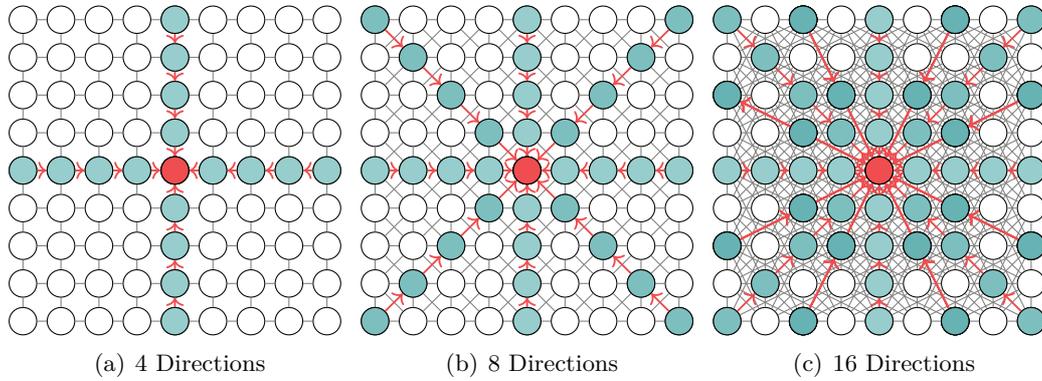


Figure 3.4 – Chemins pris en compte dans l'étape d'agrégation de SGM pour un pixel donné (en rouge) pour 4, 8 et 16 directions. Les pixels pris en compte, l'ensemble noté χ_p^g , sont représentés en bleu-vert. Plus le nombre de directions est grand, plus la couverture de l'image est grande.

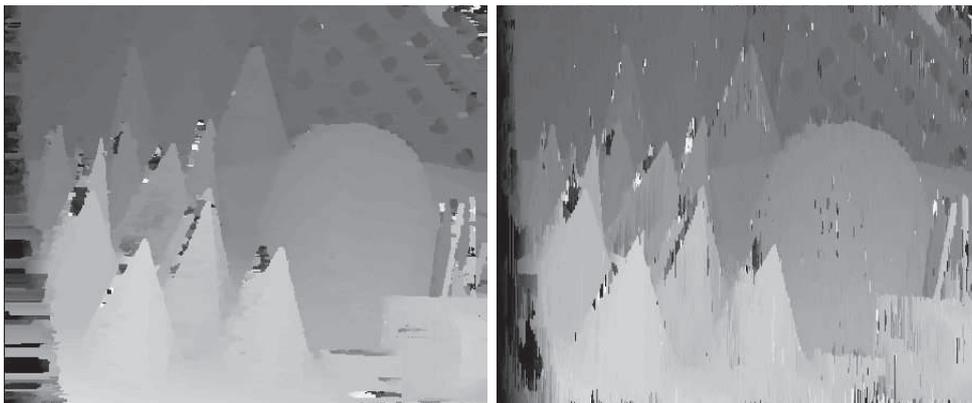


Figure 3.5 – Images de disparité obtenues par optimisation selon les lignes à gauche et selon les colonnes à droite sur la même paire. Le modèle est une attache aux données en carré tronqué au-delà de la valeur 18 et la régularisation est une fonction Potts lissée de paramètre (60, 600).

$-\varrho \in \boldsymbol{\varrho}$, de sorte que toutes les directions viennent par couple. Le problème est posé comme :

$$\begin{aligned} l_p^{SGM} &= \operatorname{argmin}_l SGM_p(l) \\ l_p^{SGM} &= \operatorname{argmin}_l D_p(l) + \sum_{\varrho \in \boldsymbol{\varrho}} Acc_p^\varrho(l) \end{aligned} \quad (3.16)$$

où Acc_p^ϱ sont les valeurs accumulées dans la direction ϱ . Ces valeurs sont calculées de manière analogue à DP dans le cas d'un champ 1D (Section 1.1.2). Soit χ_p^ϱ l'ensemble des pixels dans les directions $\boldsymbol{\varrho}$ par rapport à p , les pixels de χ_p^ϱ sont représentés en bleu dans la Figure 3.4 pour 4, 8 et 16 directions par rapport au pixel central. L'algorithme consiste donc à calculer les valeurs accumulées dans les directions ϱ pour tous les pixels, de les sommer en ajoutant la valeur d'attache aux données, puis de sélectionner le minimum. Comme dans (3.13), les valeurs $SGM_p(l)$ calculées peuvent s'écrire comme le minimum d'une énergie sous la contrainte $l_p = k$:

$$SGM_p(l) = \min_{l|l_p=l} E^{\chi_p^\varrho}(l) \quad (3.17)$$

avec $E^{\chi_p^\varrho}$ l'énergie restreinte aux pixels de χ_p^ϱ ,

$$E^{\chi_p^\varrho}(l) = \sum_{p \in I \cap \chi_p^\varrho} D_p(l_p) + \sum_{(p,q) \in N^\varrho \cap \chi_p^{\varrho^2}} R_{pq}(l_p, l_q) \quad (3.18)$$

où N^ϱ est l'ensemble des couples de pixels voisins dans les directions de $\boldsymbol{\varrho}$.

Dans [Hirschmüller, 2008], les coûts calculés sont légèrement différents. Si la position p est en dehors de l'image $L_p^\varrho(l) = 0$ et sinon $L_p^\varrho(l)$ est défini de manière récursive en fonction des valeurs obtenues à la position suivante dans la direction ϱ , $L_{p+\varrho}^\varrho(l)$:

$$L_p^\varrho(l) = D_p(l) + \min_k \left\{ R_{p,p+\varrho}(l, k) + L_{p+\varrho}^\varrho(k) \right\} - \min_k L_{p+\varrho}^\varrho(k) \quad (3.19)$$

Le terme $\min_k L_{p+\varrho}^\varrho(k)$ est constant, il ne modifie pas la solution du problème mais permet de garder les coûts dans un intervalle : si T_D majore D_p et T_R majore $R_{p,p+\varrho}$, alors

$$L_p^\varrho(l) \leq T_D + T_R \quad (3.20)$$

L'énergie SGM^* est alors définie par pixel comme :

$$SGM_p^*(l) = \sum_{\varrho} L_p^\varrho(l) \quad (3.21)$$

Les coûts L^ϱ sont reliés aux valeurs accumulées Acc^ϱ , présentées précédemment, par la relation $L_p^\varrho(l) = k_p + Acc_p^\varrho(l) + D_p(l)$, k_p étant une constante, ainsi

$$SGM_p^*(l) = K_p + (|\boldsymbol{\varrho}| - 1)D_p(l) + SGM_p(l) \quad (3.22)$$

où K_p est une constante et $|\boldsymbol{\varrho}|$ le nombre de directions prises en compte. Cette relation a été montrée dans [Drory *et al.*, 2014] où le lien est fait entre SGM et différentes méthodes dont BP et TRW-S. Les résultats de la carte de disparité de la paire Cones sont montrés pour SGM dans 4, 8 et 16 directions dans la Figure 3.6. Les méthodes SGM en prenant en compte 4, 8 et 16 directions seront notées SGM_4 , SGM_8 et SGM_{16} . Lorsque le nombre de directions prises en compte est élevé la couverture de l'image augmente et l'agrégation des coûts devient plus isotrope. Cela a pour effet de limiter les artefacts dus à la décomposition en alignements. Toutefois cette augmentation a un coût en termes de calculs. De plus, il peut être intéressant dans certains cas de favoriser certaines directions lors de l'agrégation des coûts.



Figure 3.6 – Images de disparité obtenues par méthode SGM en prenant en compte 4, 8 et 16 directions. Le modèle est le même que dans la Figure 3.5. En comparaison, la solution devient plus lisse à mesure que le nombre de directions prises en compte augmente. Cela permet de réduire les artefacts de la méthode, mais détériore la reconstruction des objets fins.

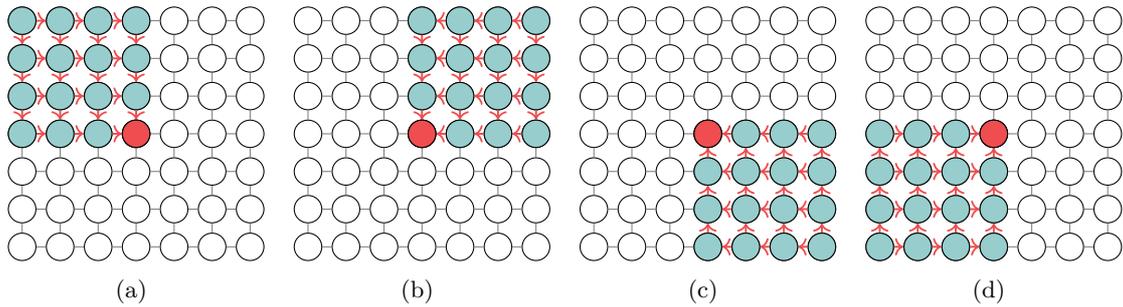


Figure 3.7 – Représentation des parties de l'image mise en jeu lorsque deux directions d'agrégation sont prises en compte simultanément comme dans les méthodes CAT et MGM

2.2 Variantes de la méthode

La méthode SGM consiste à restreindre l'ensemble des pixels mis en jeu et à ignorer les pixels en dehors. Cela a pour conséquence que de nombreux liens entre pixels ne sont pas pris en compte. Il en résulte une solution peu régulière par rapport à une solution qui atteint le minimum global d'une énergie 2D. Prendre en compte plus de liens, pour un même nombre de directions, permet de rendre la solution plus lisse. La règle de récursion peut être modifiée pour prendre en compte l'énergie accumulée dans la direction ϱ et dans la direction ϱ_{\perp} , la direction perpendiculaire à ϱ dans le sens direct. Deux possibilités ont été proposées, nommées CAT et MGM.

2.2.1 Variante CAT

Dans la méthode CAT introduite dans [Ha *et al.*, 2015], le coût provenant des pixels voisins est le minimum du coût dans les deux directions,

$$L_p^{\varrho}(l) = D_p(l) + \min_{\rho \in \{\varrho, \varrho_{\perp}\}} \min_k \left\{ R_{p, p+\rho}(l, k) + L_{p+\rho}^{\varrho}(k) \right\} \quad (3.23)$$

En analysant le coût L^{ϱ} , nous remarquons qu'il correspond au chemin de coût minimal par déplacement de ϱ ou ϱ_{perp} depuis p , de sorte que les plus courts chemins en distance de Manhattan au bord de l'image sont privilégiés. La méthode montre des améliorations sur les bases d'évaluation, toutefois la justification théorique avancée par les auteurs est erronée. Les auteurs constatent que, par construction, les coûts calculés dans CAT sont inférieurs à ceux de SGM et en déduisent la même inégalité sur l'énergie globale. Or l'infériorité des coûts de CAT par rapport à SGM est liée au fait que CAT est une relaxation de SGM. Cette observation ne dit rien sur la valeur de l'énergie globale. Un exemple où la solution issue de SGM possède une énergie globale inférieure à l'énergie de la solution CAT est donné dans la Figure 3.8.

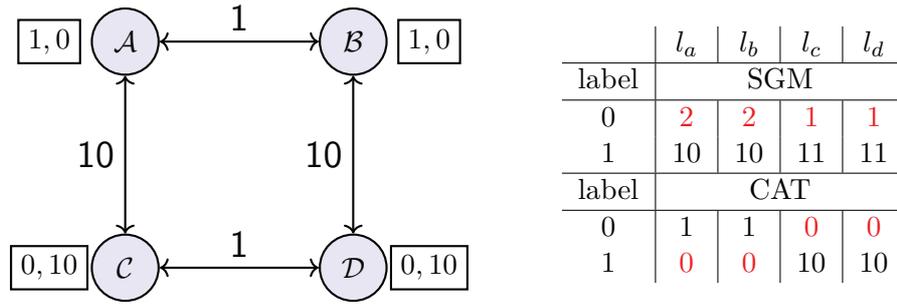


Figure 3.8 – La figure représente un problème à quatre variables binaires représenté par un graphe correspondant à la fonction pseudo-booléenne $P(a, b, c, d) = \bar{a} + \bar{b} + ab + \bar{a}\bar{b} + ab + \bar{c}\bar{d} + 10(c + d + ac + \bar{a}\bar{c} + bd + \bar{b}\bar{d})$. Les valeurs d'attache aux données pour le label 0 et le label 1 sont données à coté de chaque nœud et les coûts de régularisation sont données sur les arcs. Nous prenons en compte quatre directions. Les coûts calculés par la méthode SGM et la méthode CAT pour deux valeurs de label et pour les quatre pixels sont donnés à droite de la figure. Nous pouvons voir que les valeurs calculées par CAT sont bien inférieures à celle calculées par SGM, néanmoins SGM donne l'affectation $P(0, 0, 0, 0) = 2$ et CAT donne l'affectation $P(1, 1, 0, 0) = 20$. La solution de CAT n'atteint pas nécessairement une énergie globale inférieure à celle de la solution de SGM.

Il est à noter que la solution issue de CAT est sensible à une constante ajoutée sur une attache aux données. Cette dernière peut modifier la sélection faite par le minimum selon deux directions. Nous proposons de retrancher une constantes aux coûts accumulés, de manière analogue à Hirschmüller [2008]. L'expression (3.23) devient :

$$L_p^\varrho(l) = D_p(l) + \min_{\rho \in \{\varrho, \varrho_\perp\}} \left\{ \min_k \left\{ R_{p, p+\rho}(l, k) + L_{p+\rho}^\varrho(k) \right\} - \min_k L_{p+\rho}^\varrho(k) \right\} \quad (3.24)$$

Nous appellerons cette méthode CAT_0 . L'interprétation est alors plus difficile, toutefois en considérant que la sélection par le minimum entre les deux directions suit un loi de Bernoulli de paramètre 0.5, la probabilité que le chemin zigzague k fois en n tirages suit une loi binomiale de paramètre $(0.5, n)$. Ainsi, les pixels ont le plus de chance d'être sélectionnés s'il sont dans la direction de la première diagonale par rapport au pixel courant.

2.2.2 Variante MGM

Tout comme dans la méthode CAT, la méthode MGM proposée dans [Facciolo *et al.*, 2015] met en jeu deux directions ϱ et ϱ_\perp dans la règle de récursion, et sélectionne la moyenne des minima dans les deux directions :

$$L_p^\varrho(l) = D_p(l) + \frac{1}{2} \sum_{\rho \in \{\varrho, \varrho_\perp\}} \min_k R_{p, p+\rho}(l, k) + L_{p+\rho}^\varrho(k) \quad (3.25)$$

La valeur de L^ϱ dépend de tous les pixels compris dans le cadran défini par les vecteurs ϱ et ϱ_\perp . Pour comprendre comment se traduit la modification de la récursion de SGM sur l'énergie, nous définissons la contribution d'un pixel à p à la valeur de $L_{p_0}^\varrho$ comme étant la somme des coefficients devant les termes en D_p dans $L_{p_0}^\varrho$. Un pixel contribue pour moitié à valeur de L^ϱ situé dans les directions $-\varrho$ et $-\varrho_\perp$. Ainsi la contribution d'un pixel à la valeur de L^ϱ suit une loi binomiale

$$\text{Contribution}(p_0, p) = \frac{1}{2^{v_i+v_j}} \binom{v_i+v_j}{v_i} \quad (3.26)$$

avec (v_i, v_j) les coordonnées du vecteur entre p_0 et p dans la base (ρ, ρ_\perp) . Ces coordonnées sont positives. Ainsi les contributions à la même distance de Manhattan somment à 1 et décroissent en s'éloignant de la première diagonale. Plus les chemins "zigzaguent", plus ils ont d'importance dans le calcul du coût MGM.

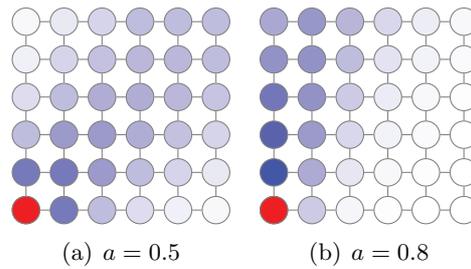


Figure 3.9 – Contributions des pixels au pixel rouge pour deux valeurs de (a) dans (3.27). Une contribution de 1 correspond à un bleu profond. Les valeurs suivant selon la perpendiculaire à la première diagonale suivent une loi binomiale. Dans le cas où le paramètre a est proche de 0.5, les pixels situés sur la première diagonale ont une plus forte contribution. Lorsque le paramètre est proche de 1, ce sont les pixels à la verticale qui ont une plus forte contribution.

2.2.3 Proposition de modification des variantes en exploitant l'anisotropie des méthodes

Nous remarquons que les deux variantes qui tentent d'exploiter deux directions ont tendance à défavoriser la prise en compte des chemins droits. Cet effet s'explique par le fait que les chemins droits sont moins fréquents que ceux qui zigzaguent. Le nombre de chemins de longueur n contenant k zigzagues suit une loi binomiale de paramètres $(0.5, n)$. Dans la méthode MGM, le paramètre 0.5 vient de l'opérateur moyenne, mais il est possible d'influer sur ce paramètre en modifiant la pondération selon les directions ϱ et ϱ_{\perp} . Ainsi la règle de récursion peut être modifiée en :

$$L_p^{\varrho}(l) = D_p(l) + \sum \begin{cases} (1-a) \left(\min_k R_{p,p+\varrho}(l,k) + L_{p+\varrho}^{\varrho}(k) \right) \\ a \left(\min_k R_{p,p+\varrho_{\perp}}(l,k) + L_{p+\varrho_{\perp}}^{\varrho}(k) \right) \end{cases} \quad (3.27)$$

La Figure 3.9 illustre la modification des contributions en fonction du paramètre a . Pour rendre le problème symétrique, pour chaque cadran l'accumulation doit être réalisée avec le paramètre a et le paramètre $1-a$. Ainsi pour modifier le MGM, le nombre d'agrégations à réaliser est double par rapport à SGM. Cette méthode est désignée par la suite comme MGM_a . Il est à noter que MGM_1 est la méthode SGM et que $\text{MGM}_{0.5}$ est le MGM original.

La méthode CAT_0 peut elle-aussi être modifiée pour valoriser la direction ϱ par rapport à la direction ϱ_{\perp} en ajoutant une constante K aux coûts issus de la direction perpendiculaire :

$$L_p^{\varrho}(l) = D_p(l) + \min \begin{cases} \min_k \left\{ R_{p,p+\varrho}(l,k) + L_{p+\varrho}^{\varrho}(k) \right\} - \min_k L_{p+\varrho}^{\varrho}(k) \\ K + \min_k \left\{ R_{p,p+\varrho_{\perp}}(l,k) + L_{p+\varrho_{\perp}}^{\varrho}(k) \right\} - \min_k L_{p+\varrho_{\perp}}^{\varrho}(k) \end{cases} \quad (3.28)$$

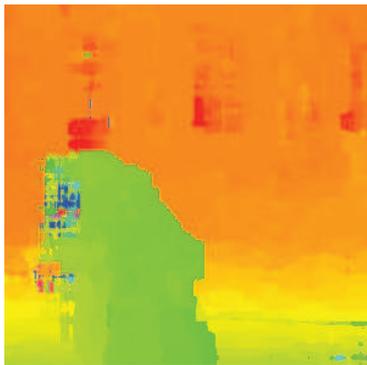
Cette méthode est désignée par la suite comme CAT_K . Il est à noter que CAT_{∞} correspond à la méthode SGM et nous retrouvons la méthode CAT_0 lorsque $K = 0$.

2.3 Raffinement de la solution par DP

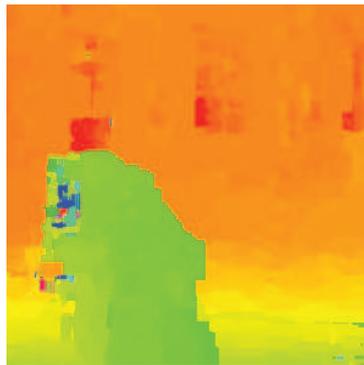
Dans la méthode SGM et ses variantes, il n'y a pas de garantie de régularité de la solution du fait d'une sélection d'un minimum par pixel. Il est donc fréquent qu'apparaissent les artefacts sur la solution qui sont liés à la trame de l'image. Un lissage fréquemment utilisé est d'appliquer un filtre médian sur la solution. Ce traitement permet de corriger des valeurs aberrantes isolées tout en conservant les contours. Nous proposons plutôt de rendre la solution plus lisse en optimisant un sous-ensemble de variable vis à vis de l'énergie prise en 4-connexité. L'idée est de minimiser l'énergie en optimisant alternativement chaque ligne et chaque colonne, les autres étant figées à la valeur de la solution précédente. Seul un alignement de pixel est mis en jeu ce qui permet une



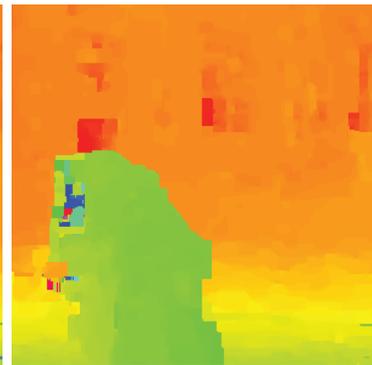
(a) Image gauche



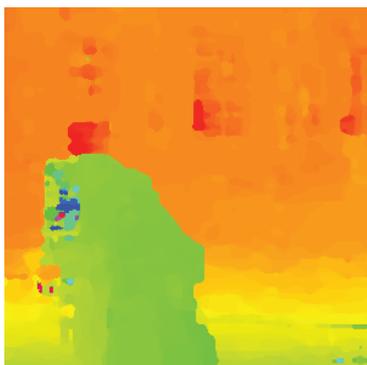
(b) Sans Filtrage



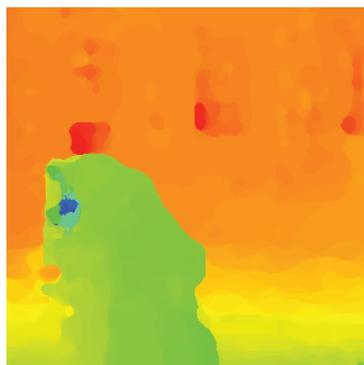
(c) 1 Opt.HV



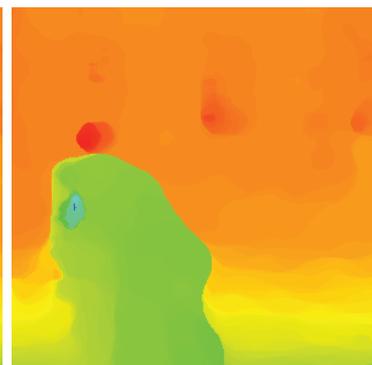
(d) 2 Opt.HV



(e) Médian 5x5



(f) Médian 11x11



(g) Médian 21x21

Figure 3.10 – Image gauche et extrait de carte de disparité (cycliste) obtenue pour le couple 2 de la base KITTI15 par la méthode SGM_4 après différents raffinements : (b) et (c) pour Opt.HV, (e), (f) et (g) par filtrage médian.

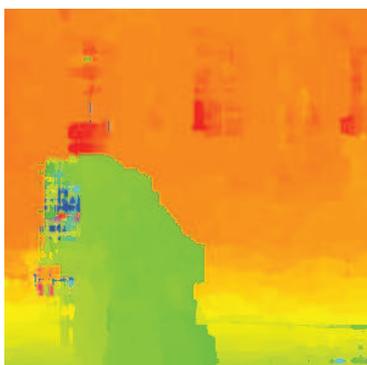
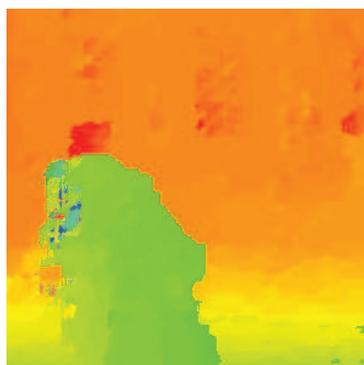
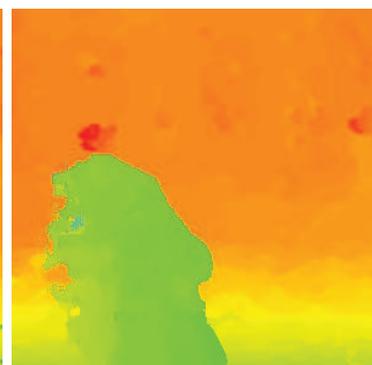
(a) SGM_4 (b) SGM_8 (c) SGM_{16}

Figure 3.11 – Extrait de l'image de disparité obtenue pour le couple 7 de la base KITTI par les méthodes SGM_4 SGM_8 et SGM_{16} . L'extrait correspond à un cycliste prenant un virage.

résolution de manière exacte par DP. Cette méthode converge vers une valeur énergie, souvent assez éloignée du minimum global en 4-connexité, mais elle permet de donner un caractère plus lisse à la solution comme nous pouvons l'observer dans la Figure 3.10. Nous appelons cette méthode de raffinement Opt.HV, pour optimisation horizontale/verticale.

3 Expérimentation

Dans cette section, nous présentons différents résultats de reconstruction stéréoscopique par DP. Nous allons évaluer sur un ensemble de 10 couples pris parmi les 12 couples de la base Middleburry dite "de 2014" (Figure 3.13 et 3.13) : huit couples de la base de 2014 [Scharstein *et al.*, 2014b] (nommés Adirondack, Motorcycle, Piano, Pipes, Playroom, Playtable, Recycle, Shelves), un de la base de 2005 [Hirschmüller et Scharstein, 2007] (ArtL) et un de la base de 2003 [Scharstein et Szeliski, 2003] (Teddy). Nous avons retiré les couples possédant plus 256 valeurs de disparité. L'attache aux données du modèle est le Census calculé sur l'intensité d'une fenêtre 5 par 5 pixels et agrégé 2 fois par CBCA (cf. Section 2.2.2 du Chapitre 2). La régularisation est la fonction de Potts lissée des différences de disparité de paramètres $(P1, P2) = (24, 96)$. Le critère d'évaluation, noté err.M, est le pourcentage de pixels dont la disparité est à une différence de plus de un de la référence. L'évaluation est réalisé sur les points non-occultés de la vérité terrain. Pour compléter ces résultats, une évaluation sur les 10 premiers couples de la base de test de KITTI15 [Menze et Geiger, 2015] est réalisée (Figure 3.14). L'attache aux données est le Census calculé sur les intensités d'une fenêtre 5 par 5 pixels et agrégé 2 fois par CBCA. La régularisation est la fonction de Potts lissée des différences de disparité avec un autre réglage des paramètres $(P1, P2) = (1.2, 23)$. L'énergie est multipliée par 10 pour ne manipuler que des valeurs entières. Avec ce jeux de données, le critère d'évaluation, noté err.K, est le pourcentage de pixels dont la disparité est à une différence de plus de 3 de la référence. Nous utilisons pour cela la vérité terrain "non-occ" qui ne contient que les points visibles dans l'autre image. Dans un premier temps nous observerons l'effet du nombre de directions prises en compte dans la méthode (Section 3.1). Dans un second temps, nous évaluerons les variantes de la méthode SGM entre elles (Section 3.2). Et enfin, nous comparerons les méthodes fondées sur la DP et celles fondée sur les coupure de graphe (CG) (Section3.3).

3.1 Effet du nombre de directions prises en compte dans SGM et du raffinement de la solution

Dans [Hirschmüller et Scharstein, 2007], la méthode SGM est mise en œuvre en prenant en compte 16 directions. Cependant, ne prendre en compte que 4 ou 8 directions peut être suffisant pour obtenir une solution (Section 3.1.1). De plus, ces solutions brutes issues de cette méthode sont souvent raffinées par post-traitement (Section 3.1.2).

3.1.1 Comparaison des méthodes SGM₄, SGM₈ et SGM₁₆

Le nombre de directions prises en compte dans l'étape d'accumulation de SGM est généralement de 4, 8 et 16. La prise de compte de directions supplémentaires fait augmenter la quantité de calculs mais permet d'obtenir des solutions plus lisses, comme nous pouvons le voir dans la Figure 3.15. Lorsque le nombre de directions augmente, les artefacts de la méthode, qui consistent en tramage, sont réduits. Nous observons sur ces trois résultats que l'énergie définie en 4-connexité diminue à mesure que le nombre de directions prises en compte augmente (64 463 678, 53 931 033 et 52 424 183 avec SGM₄ SGM₈ et SGM₁₆, respectivement). La diminution de cette énergie s'explique en grande partie par la réduction des artefacts. En effet, quand la solution n'est pas lisse, le coût de régularisation devient prépondérant dans l'énergie. Ainsi l'énergie globale en 4-connexité est donnée ici à titre de mesure de la quantité de valeurs de disparité proches.

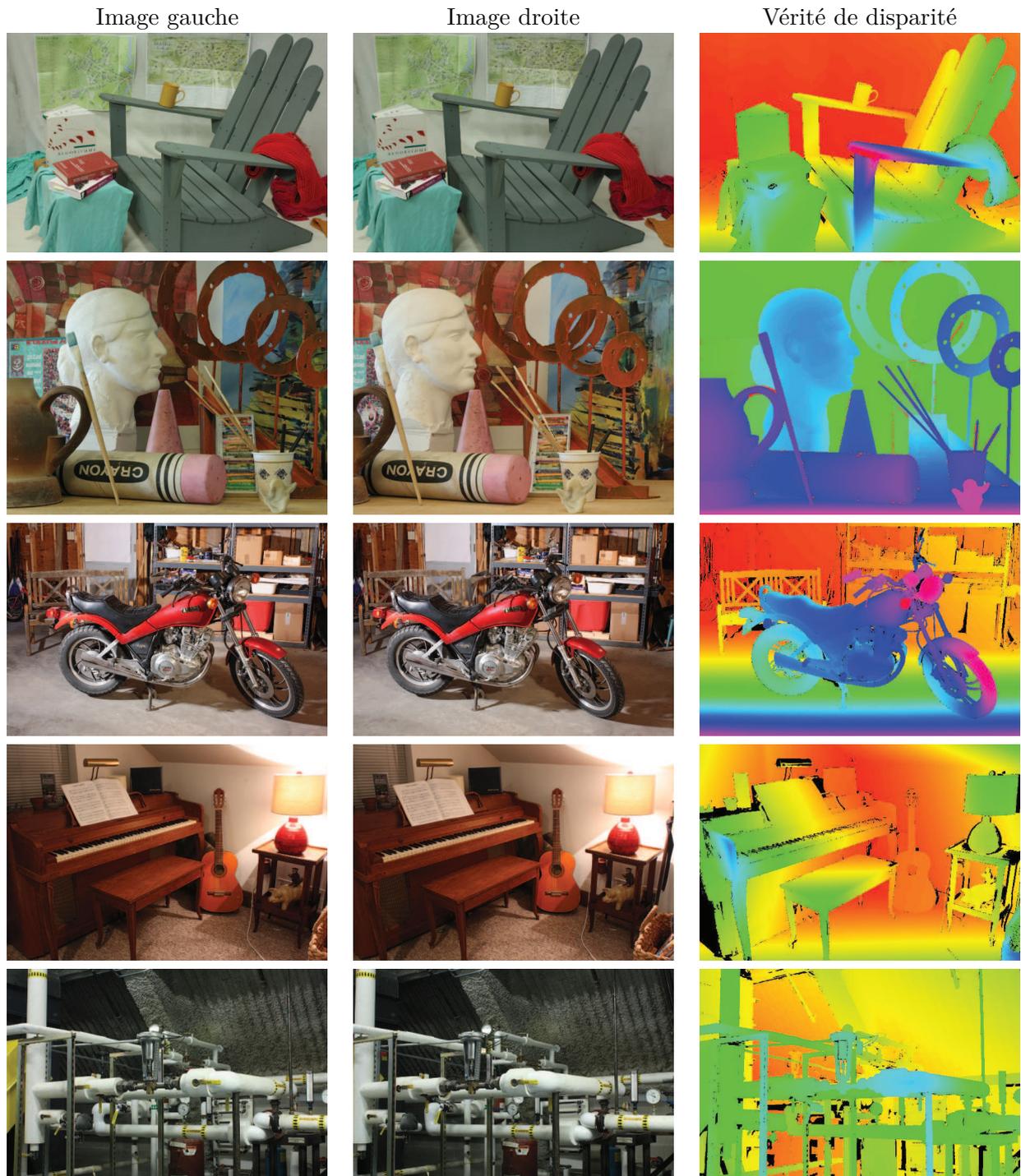


Figure 3.12 – Cinq des dix couples de la base Middlebury ('Adirondack', 'ArtL', 'Motorcycle', 'Piano', 'Pipes', 'Playroom'). Sont présentés l'image gauche, image droite et la vérité de carte de disparité.

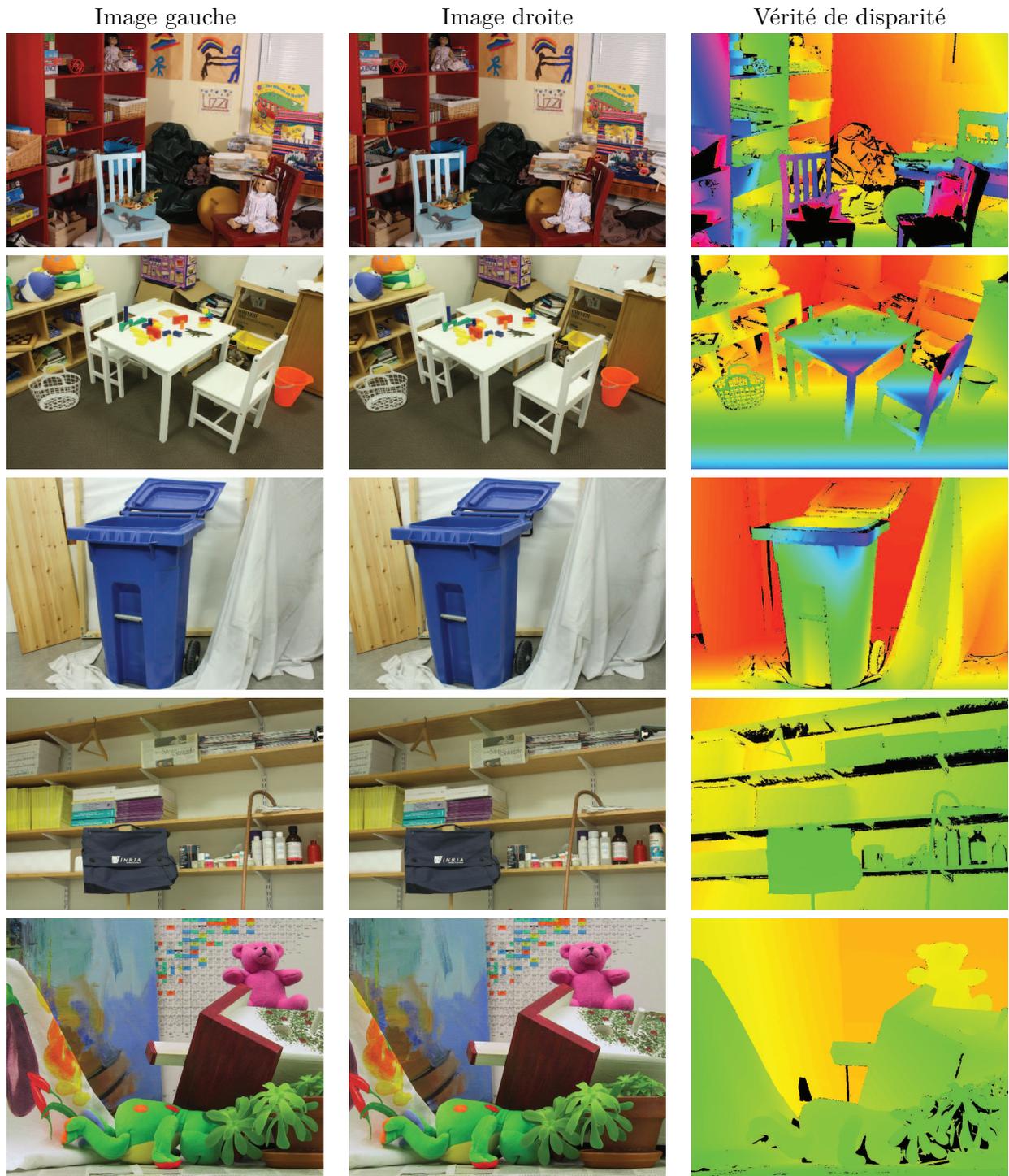


Figure 3.13 – Cinq des dix couples de la base Middlebury ('Playtable', 'Recycle', 'Shelves', 'Teddy'). Sont présentés l'image gauche, image droite et la vérité de carte de disparité.



Figure 3.14 – Dix premiers couples de la base KITTI15. Sont présentés l'image gauche et la carte de disparité "non-occ"

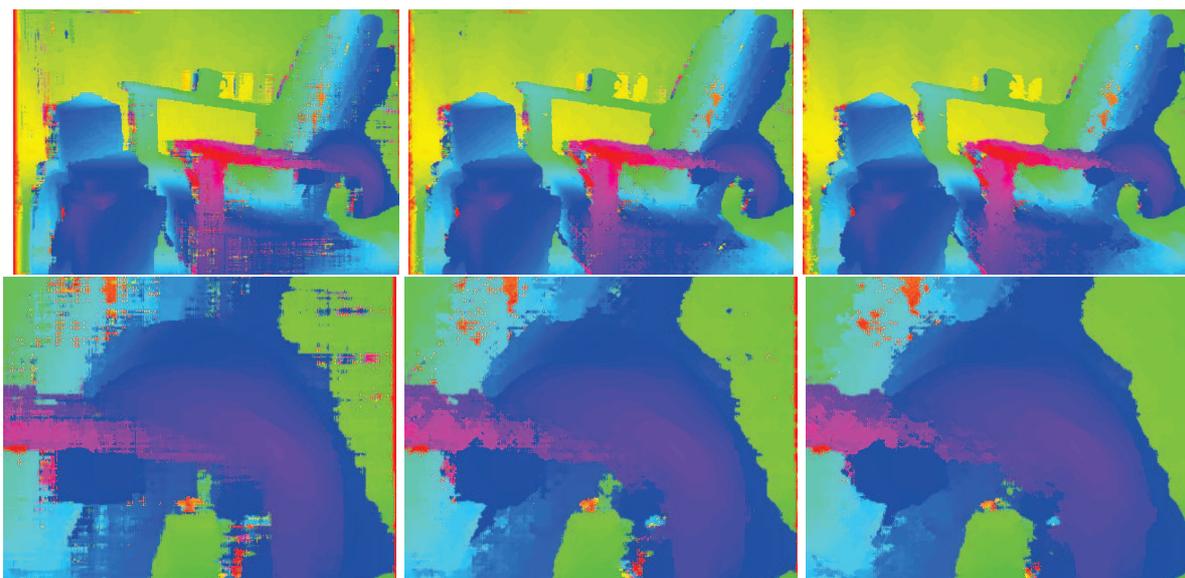


Figure 3.15 – Influence sur nombre de directions prises en compte dans SGM (4, 8 et 16) sur la reconstruction du couple Adirondack de la base Middlebury, ainsi qu'un zoom. Nous observons que la solution est plus lisse et que les artefacts de la méthode SGM sont atténués lorsque le nombre de directions augmente.

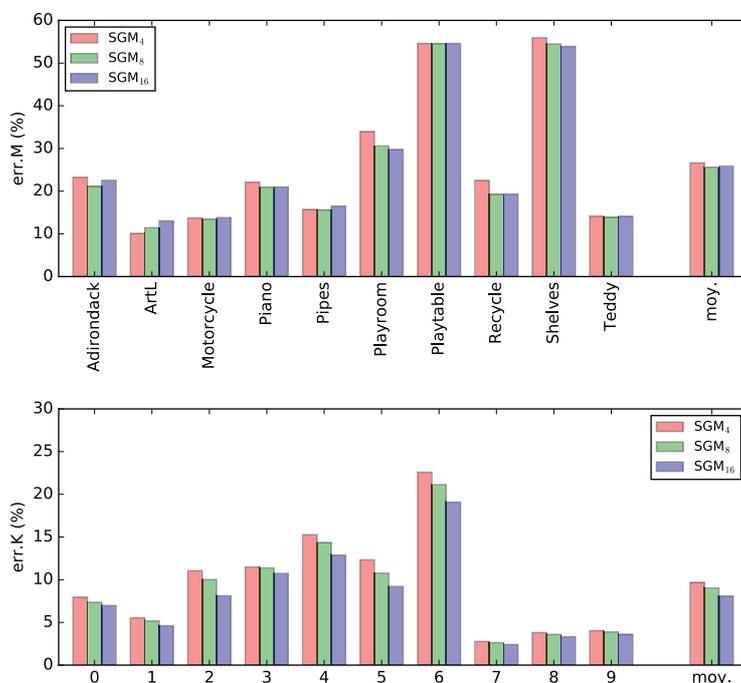


Figure 3.16 – Taux d'erreur des méthodes SGM₄, SGM₈ et SGM₁₆ sur les 10 couples de la base Middlebury en haut, et sur 10 premiers couples de la base KITTI15 en bas. Le taux d'erreur est la proportion de pixel à plus d'une disparité de la vérité de la base Middlebury (err.M) et à plus de 3 disparités de la vérité de la base KITTI (err.K). Le taux d'erreur err.K baisse à mesure que le nombre de directions augmente. Les résultats sont plus contrastés sur la base Middlebury.

Raffinement	SGM ₄	SGM ₈	SGM ₁₆
Aucun	6.90%	6.55%	6.04%
Opt.HV, 1 fois	6.63%	6.42%	6.02%
Opt.HV, 2 fois	6.54%	6.37%	6.03%
Filtrage médian 5x5	6.70%	6.47%	6.04%
Filtrage médian 11x11	6.34%	6.22%	5.91%

Tableau 3.2 – Taux d’erreur KITTI calculés sur toute la base KITTI15. Les raffinements sont l’optimisation par ligne puis les colonnes de la solution réalisé 1 et 2 fois (1 Opt. HV et 2 Opt. HV), et un filtrage médian de la solution avec deux tailles de fenêtre. Les raffinements sont réalisées sur les solutions des méthodes SGM₄, SGM₈ et SGM₁₆. Opt.HV permet d’améliorer le taux d’erreur, mieux qu’un filtrage médian 5x5, mais moins bien qu’un filtrage médian 11x11

Dans la Figure 3.16 nous présentons les taux d’erreur $err.M$ des couples de la base Middlebury pour les méthodes SGM₄, SGM₈ et SGM₁₆, ainsi que le taux d’erreur $err.K$ sur les couples de la base KITTI15. Sur le couple de la base Middlebury, passer de quatre à huit directions fait augmenter le nombre de pixels faux, pour le couple ArtT de +1.32% et le fait baisser jusqu’à -3.39%, avec une moyenne -1.11%, sur les autres couples d’images. Il en est de même lorsqu’on passe de quatre à seize directions : l’augmentation est de +1.32% pour le couple ArtL et la baisse jusqu’à -4.17%, avec une moyenne -0.75% sur les autres couples de la base. Cette amélioration s’explique en partie par le fait que les solutions sont plus lisses et donc présentent moins de tramage, comme nous avons pu l’observer sur le couple Adirondack dans la Figure 3.15. Nous pouvons comparer l’effet du nombre de directions dans la Figure 3.11 sur l’exemple données pour les raffinements de la Figure 3.10. Ce lissage de la solution ne se traduit pas forcément par une amélioration du taux d’erreur dans le cas du couple ArtL. Ce couple contient des objets fins qui sont alors mal reconstruits. Sur les couples de la base KITTI, nous observons que la prise en compte de directions supplémentaires permet de diminuer le nombre de pixels faux entre -0.12% et -1.56%, en moyenne -0.36%, entre SGM₄ et SGM₈ directions, et de -0.36% à -3.50%, en moyenne -1.59% entre SGM₄ et SGM₁₆. En conclusion, l’intérêt de la prise en compte de directions supplémentaires est assez clair sur la base KITTI, et dépend en pratique du couple considéré sur la base Middlebury.

3.1.2 Effet des raffinements

Dans cette section, nous étudions le gain en terme de taux d’erreur de deux méthodes de raffinements de la solution : Opt.HV et le filtrage médian. Les résultats avant et après raffinement de toute la base KITTI15 sont donnés dans le Tableau 3.2. Le raffinement optimisation horizontale/verticale (Opt.HV) apportent des améliorations pour SGM₄ et SGM₈. Le nombre de pixels faux varie de -0.27% et -0.46%, respectivement après une et deux itérations, pour SGM₄ et de -0.13% et -0.18%, respectivement après une et deux itérations, pour SGM₈. L’évolution est peu significative dans le cas de SGM₁₆. Le filtrage médian par une fenêtre 5x5 permet de réduire le nombre pixel faux -0.20% et -0.12% pour les méthodes SGM₄ et SGM₈ et aucun changement pour SGM₁₆, ce qui est moins qu’avec le raffinement Opt.HV. Le filtrage médian sur une fenêtre 11 par 11 permet de diminuer plus significativement de nombre de pixels faux pour les trois méthodes, soit -0.56%, -0.33% et -0.13%. Ainsi, nous remarquons que différents traitements sont moins efficaces sur SGM₁₆ qui donne déjà de bon résultats avant post-traitement. Le raffinement Opt.HV améliore les résultats mais moins qu’un filtrage médian avec une fenêtre large. Le raffinement Opt.HV a pour avantage d’optimiser la solution, même de manière sous-optimal, vis-à-vis de l’énergie globale, ce qui la rend plus facilement comparable avec des solutions obtenues par GC.

3.2 Variantes des méthodes SGM

Les méthodes CAT et MGM sont des variantes de la méthode SGM qui prennent en compte simultanément deux directions lors du calcul des coûts. Nous proposons de comparer les résultats de ces méthodes dans le cas de la prise en compte de 4 directions, ainsi que les propositions de variantes que nous avons faite pour ces deux méthodes. Nous choisissons de donner les résultats sans raffinements de la solution pour mieux observer les caractéristiques des solutions issues de chaque méthode.

3.2.1 Méthode CAT et ses variantes

Nous comparons la méthode SGM avec les méthodes CAT et les variantes CAT_K pour différents valeurs de K : 0, 4, 16 et 32. Nous voulons observer l'influence du paramètre K sur les résultats. Les cartes des disparités estimées pour le couple Adirondack sont présentées dans la Figure 3.17. Nous observons que CAT réduit une partie des artefacts de tramage de la méthode SGM mais des artefacts en ligne apparaissent perpendiculairement aux contours des objets. La variante CAT_0 ne produit pas de tramage comme SGM ni de ligne comme CAT, mais plutôt des tâches d'erreur. Lorsque le paramètre K augmente, les solutions forment un continuum entre la solution de la méthode CAT_0 et celle de la méthode SGM.

Les taux d'erreur pour les couples de la base KITTI et Middlebury sont présentés Figure 3.18. Sur la base Middlebury, la méthodes CAT donne systématiquement un nombre de pixels faux plus élevé que pour la méthode SGM, supérieur en moyenne de +3.05%. Parmi les variantes, les meilleurs taux sont obtenus pour $K = 16$, avec un nombre de pixels faux en moyenne de -0.98% d'erreur que par rapport à la méthode SGM. Sur la base KITTI, la méthode CAT détériore les résultats de -0.1% à -4.68%, en moyenne de -1.87%. Les variantes de CAT détériorent elles-aussi les résultats. Cette détérioration est plus forte pour les faibles valeurs du paramètre K que pour les grandes valeurs : le nombre de pixels faux augmente en moyenne de +0.59% entre SGM et CAT_{16} et de +2.08% entre SGM et CAT_0 .

3.2.2 Méthode MGM et ses variantes

Nous comparons la méthode SGM avec les méthodes MGM et les variantes de MGM_a pour différentes valeurs du paramètres a : 0.9, 0.8, 0.7 et 0.6. Nous voulons observer l'influence de la pondération sur les résultats. Les images de disparité pour le couple Adirondack sont montrées dans la Figure 3.19. Nous remarquons que les solutions forment un continuum entre SGM et MGM au fur et à mesure que le paramètre évolue. Les artefacts de tramage de la méthode SGM prennent la forme de taches dans la méthode MGM.

Les taux d'erreur pour les couples de la base KITTI et Middlebury sont présentés Figure 3.20. Sur la base Middlebury, sur les couples ArtL et Motorcycle, les résultats de la méthode MGM sont moins bon que ceux de SGM (+2.64% et +1.10% de pixels faux). Néanmoins sur les dix couples, la méthode MGM donne en moyenne -0.54% de pixel faux que SGM. Si l'on regarde les résultats des variantes de MGM, elles donnent presque les mêmes taux d'erreur que MGM pour les couples Pipes, Playtable et Shelves. Hormis pour le couple ArtL, les variantes de MGM donne de meilleurs taux d'erreur que SGM. Le meilleur taux moyen est obtenu par $MGM_{0.8}$, dont le nombre de pixels faux est en moyenne de -1.56% par rapport à SGM. Sur la base KITTI, MGM et ses variantes dégradent les résultats. Cette dégradation se réduit à mesure que les variantes de MGM s'approchent de la méthode SGM. En moyenne, le nombre de pixels faux de SGM, $MGM_{0.9}$, $MGM_{0.8}$ est très proches (moins de 0.2% de différence).

En conclusion, les méthodes CAT et MGM ainsi que leur variantes présentent un intérêt limité, sinon détériorent les résultats obtenus sur la base KITTI. Sur la base Middlebury, les conclusions sont plus nuancées. La méthode CAT n'apporte pas d'amélioration, mais sa variantes CAT_{16} donne de meilleurs résultats que SGM. La méthode MGM et ses variantes donnent

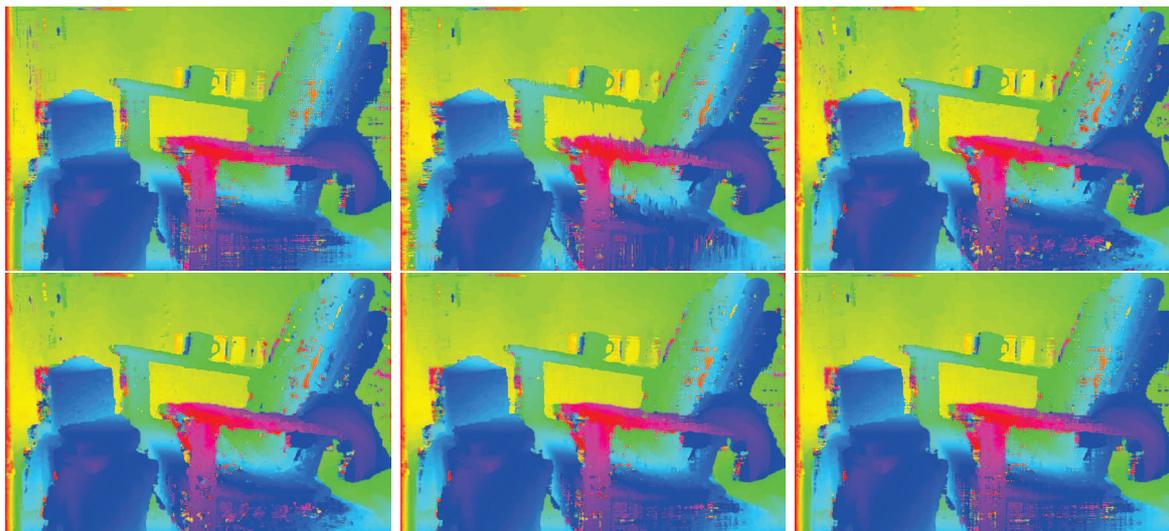


Figure 3.17 – Résultats de reconstruction pour les méthodes SGM, CAT et CAT_0 sur la première ligne ainsi que les variantes CAT_4 , CAT_{16} et CAT_{32} sur la deuxième ligne. La solution CAT comporte des artefacts en ligne suivant la perpendiculaire aux contours. La méthode CAT_0 présente moins d’artefact au niveau des contours, mais des taches d’erreur supplémentaires. À mesure que le paramètre K augmente, la solution tend vers la solution de la méthode SGM.

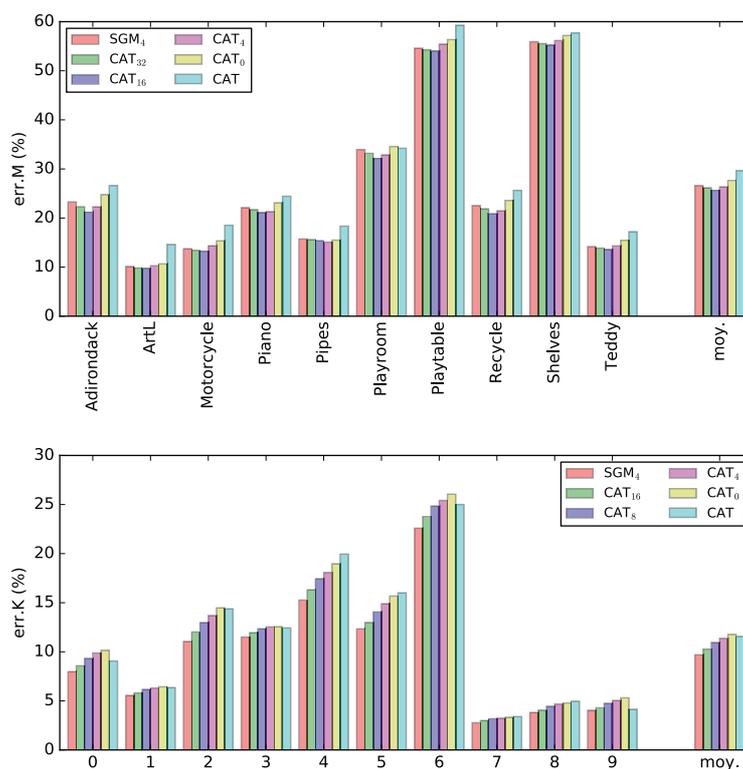


Figure 3.18 – Taux d’erreur des méthodes SGM dans 4 directions, CAT, CAT_0 , CAT_4 , CAT_{16} , CAT_{32} et sur 10 couples de la base Middlebury (err.M) en haut, et sur les 10 premiers couples de la base KITTI15 (err.K) en bas. Sur la base Middlebury, la méthode CAT détériore les résultats par rapport à SGM. Les variantes de CAT donnent les meilleurs résultats pour $K = 8$ et surpasse ceux la méthode SGM. Sur la base KITTI, toutes les méthodes fondées sur CAT donnent de moins bons résultats que SGM. Les variantes de CAT donnent alors de meilleures résultats à mesure qu’elle s’approche de la méthode SGM.

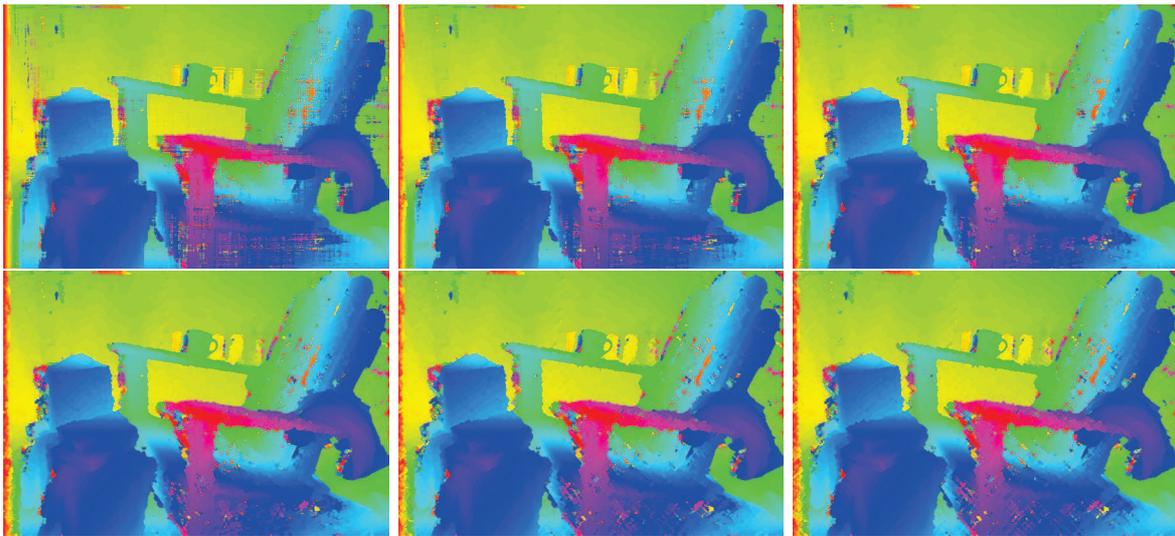


Figure 3.19 – Résultats de reconstruction des méthodes SGM, MGM, $MGM_{0.9}$, $MGM_{0.8}$ et $MGM_{0.7}$. Les différents solutions forment un continuum entre la solution de la méthode SGM de tramage, et MGM qui présente des erreurs plus localisées.

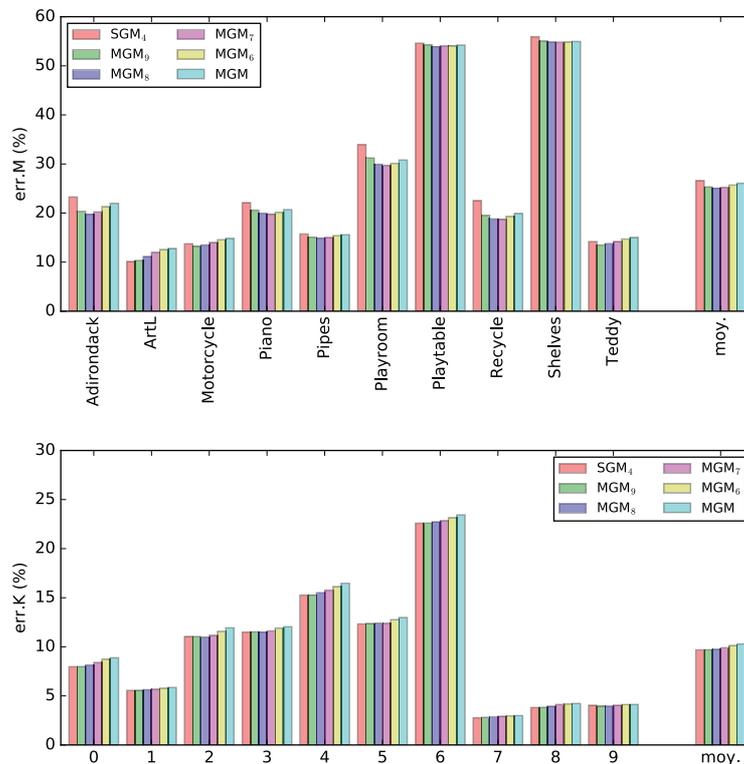


Figure 3.20 – Taux d'erreur des méthodes SGM, MGM, $MGM_{0.9}$, $MGM_{0.8}$, $MGM_{0.7}$ et $MGM_{0.6}$ sur 10 couples de la base Middlebury (err.M) en haut, sur les 10 premiers couples de la base KITTI15 (err.K) en bas. Sur la base Middlebury, la méthode MGM donne de meilleurs résultats que SGM sauf pour les couples ArtL et Motorcycle. Le meilleur résultat moyen est obtenu pour $MGM_{0.8}$.

des taux d'erreur plus bas que SGM, le meilleur taux étant obtenu pour $MGM_{0.8}$. Les deux variantes de CAT et MGM qui donnent les meilleurs résultats sont celles qui, de par leur valeur de paramètre, donnent les solutions à mis chemin entre SGM et la méthode dont elles sont la variante. Dans [Ha *et al.*, 2015], la méthode CAT permet des gains importants sur la base KITTI12, ce que nous n'avons pas pu observer dans cette section. La principale différence entre le modèle utilisé dans cet article et notre modèle est l'étape d'agrégation des coûts par CBCA. Cette opération lisse localement les coûts d'attache aux données à disparité constante. Ainsi, les coûts d'attache aux données de pixels voisins ont tendance à être corrélés ce qui limite les incohérences entre les lignes dans la solution SGM. La solution obtenue étant plus lisse avec moins d'artefacts, il est plus difficile d'observer l'intérêt de la méthode CAT. De même, la CBCA a pu aussi avoir un impact sur les performances de la méthode MGM. En effet, le modèle utilisé dans [Facciolo *et al.*, 2015] sur les couples de la base Middlebury est presque le même que celui que nous avons utilisé à deux différences près : nous avons agrégé les attaches à données par CBCA et nous avons pris en compte quatre directions au lieu de huit. Or, on remarque que le gain apporté par MGM est plus important avec un modèle sans agrégation des coûts. Plus spécifiquement sur la base KITTI, l'absence d'amélioration observée peut s'expliquer par la structure de la scène qui privilégie les directions verticales et horizontales de l'image. Or les variantes de SGM_4 ont tendance à diminuer l'importance des a priori verticaux et horizontaux par rapport aux autres directions.

3.3 Comparaison entre CG et DP

Nous avons présenté deux types de méthode d'optimisation discrète, la programmation dynamique (DP) et la formulation par coupure de graphe (GC). Ces deux méthodes peuvent être employées pour traiter le problème de reconstruction mais l'énergie optimisée, et donc le modèle utilisé, n'est pas le même. Dans le cas où GC est utilisé sur un champ de Markov 2D, l'énergie sera nommée énergie en 4-connexité. Dans le cas où DP est utilisé, l'énergie n'a pas la structure 2D, il en résulte que les solutions issues de ces deux méthodes d'optimisation n'ont pas les mêmes caractéristiques. Nous cherchons dans cette section à illustrer ces différences. Nous verrons que la solution minimale d'une énergie en 4-connexité est parfois mieux approchée avec la méthode SGM qu'avec les méthodes approchées par GC (Section 3.3.1). Dans un second temps, nous montrons que, pour autant, la solution de SGM n'a pas les mêmes propriétés que la solution minimale de l'énergie en 4-connexité (Section 3.3.2).

3.3.1 Solution SGM comme approximation de la solution globale en 4-connexité

Nous avons vu dans la Section 3.2.3 du Chapitre 2 que la méthode d'optimisation α -Expansion peut donner une solution très éloignée de la solution globale. Nous nous sommes donc demandé quelles sont les performances des méthodes par DP pour la même énergie. L'attache aux données est une fonction de la différence des intensités après application d'un filtre de Sobel horizontal de taille 3 par 3 pixels. Cette fonction est un carré tronqué au-delà d'une différence de 18. La fonction de régularisation r est la valeur absolue et le paramètre λ de l'énergie est de 20. Dans la Figure 3.21, les solutions sur la même paire avec différentes méthodes d'optimisation sont montrées. La solution minimale du problème permet de reconstruire la profondeur des principaux objets de la scène. La solution approchée par α -Expansion est très éloignée de la solution minimale : la route et la partie gauche de la scène n'est pas reconstruite. La solution de SGM_4 reconstruit grossièrement la scène mais la solution présente de nombreux artefacts de tramage. La solution de SGM_{16} est plus proche de la solution minimale mis à part dans le ciel. Enfin, même en initialisant α -Expansion avec la solution issue de SGM_{16} la partie basse de la chaussée en bas de l'image n'est pas bien reconstruite. La chaussée est une zone avec de faibles variations d'intensités, de sorte que l'attache aux données est très ambiguë, et la variation du coût est plutôt liée au bruit de l'image. La bonne reconstruction de la chaussée dépend de la capacité de la méthode d'optimisation à interpoler dans ces zones. α -Expansion fait une

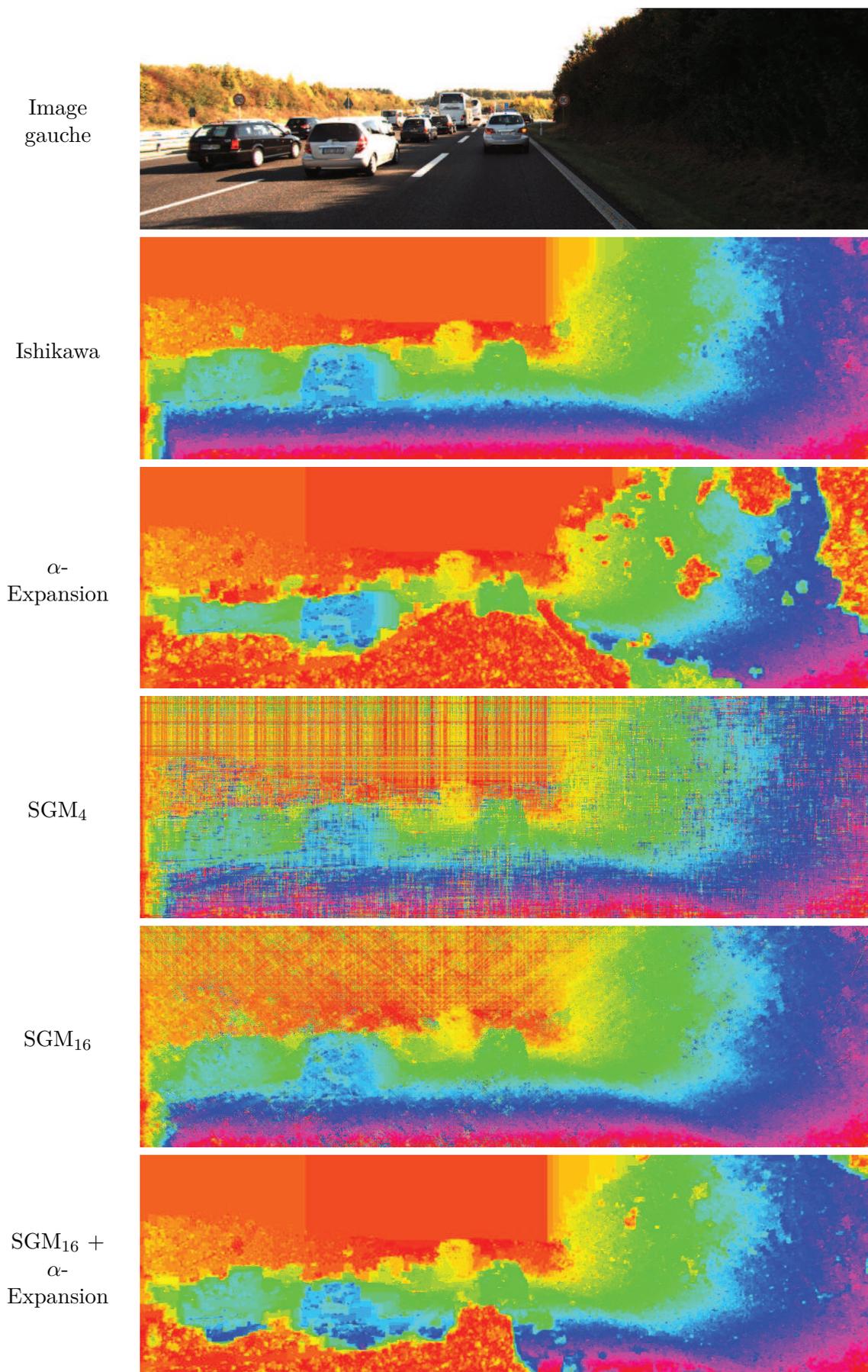


Figure 3.21 – Image gauche du couple 199 de la base KITTI15 et cartes de disparité avec la méthode Ishikawa, α -Expansion, SGM₄ et SGM₁₆. Le dernier résultat est obtenu par α -Expansion, en initialisant par la solution de SGM dans 16 directions.

approximation globale de la solution en proposant un plan fronto-parallèle. Or la chaussée n'est pas fronto-parallèle ce qui limite les performances d' α -Expansion. La méthode SGM, en prenant en compte l'information de la ligne et colonne de manière exacte, permet de prendre en compte correctement les interactions à long distance. Ainsi, la méthode SGM est capable d'interpoler dans les zones homogènes et bruitées. La solution de SGM₄ possède beaucoup d'artefacts, ce qui indique que l'information utilisée dans la ligne et la colonne reste trop partielle. La méthode SGM₁₆, en prenant en compte une plus grande partie de l'image, est en mesure de reconstruire correctement une grande partie de la scène. Il ne semble pas y avoir de lien théorique entre SGM₁₆ et l'énergie globale en 4-connexité, toutefois nous observons que cette méthode peut être plus proche du minimum global de cette énergie que des méthodes qui approximent explicitement la solution minimale.

Le fait qu' α -Expansion, même après initialisation par SGM₁₆, présente des parties quasi fronto-parallèle sur la chaussée, illustre les difficultés d' α -Expansion à interpoler sur de longues distances. Comme la solution recherchée ne respecte pas complètement l'a priori fronto-parallèle, c'est une solution différente qui est sélectionnée mais qui respecte l'a priori.

3.3.2 Différences entre les énergies utilisées dans GC et SGM

Nous avons vu dans la section précédente qu'en augmentant le nombre de directions prise en compte dans SGM, la solution semble approcher de la solution minimale de l'énergie en 4-connexité. La question est de savoir si l'approximation faite par SGM est d'autant meilleure que le nombre de directions prises en compte est grand. Tout d'abord, il ne semble pas y avoir de lien direct entre l'énergie en 4-connexité et avec SGM avec un nombre de directions élevé. Ensuite, le rapport du nombre de termes de régularisation sur le nombre de termes d'attache aux données est d'environ $N/2$ pour une énergie globale en N -connexité et d'environ 1 pour SGM, indépendamment du nombre de directions prises en compte. En outre, selon la direction, les interactions prises en compte se font à des distances différentes, ce qui nécessite probablement des pondérations adaptées pour être cohérent avec un champ de Markov 2D. Ainsi, ajouter des directions à SGM ne permet probablement pas d'approcher le minimum de l'énergie en 4-connexité si des pondérations correctes ne sont pas appliquées à la régularisation.

Nous allons montrer maintenant la différence d'intensité de la régularisation entre SGM et les méthodes par GC, en prenant la même fonction d'attache aux données et la même fonction de régularisation avec ces deux types de méthode. L'attache aux données est le Census sur l'intensité et la régularisation est une fonction de Potts lissée. Dans la Figure 3.23, nous présentons les résultats avec différentes méthodes d'optimisation. Les énergies et taux d'erreur correspondants sont présentés dans le Tableau 3.3. Minimiser la somme des attaches aux données par bloc 11 par 11 donne des résultats intéressants (B11x11). La solution est localement bruitée, mais la structure de la scène est bien reconstruite. Dans les solutions obtenues par les méthodes SGM₄, SGM₈ et SGM₁₆, la scène est bien reconstruite, avec une réduction des artefacts de tramage et des objets fins bien reconstruits, à mesure de la nombre de directions prises en compte augmente. La solution obtenue par GC (schéma Alterné-C) est sur-régularisée et des parties de scènes, comme la chaussée, sont mal reconstruites. Cela illustre qu'une fonction de régularisation identique conduit à une régularisation plus ou moins forte de la solution selon le type de régularisation.

Il est fréquent que les paramètres de la fonction de régularisation qui donnent les meilleurs résultats soient très différents entre SGM et les méthodes par GC. Dans le Tableau 3.4, la fonction r de régularisation est une fonction de Potts lissée où les paramètres $P1, P2$ de la régularisation ont été optimisés pour donner le meilleur taux d'erreur KITTI (err.K). Le taux d'erreur est de 3.0% pour la méthode Alterné-C après une itération, et de 1.94% pour SGM dans 4 directions. Si nous intervertissons les fonctions de régularisation entre les deux modèles, le taux d'erreur d'Alterné-C est alors de 23.17% et la solution est sur-régularisée. Réciproquement, le taux d'erreur de SGM passe à 2.88%. La solution SGM a donc été moins impactée par le changement des paramètres.

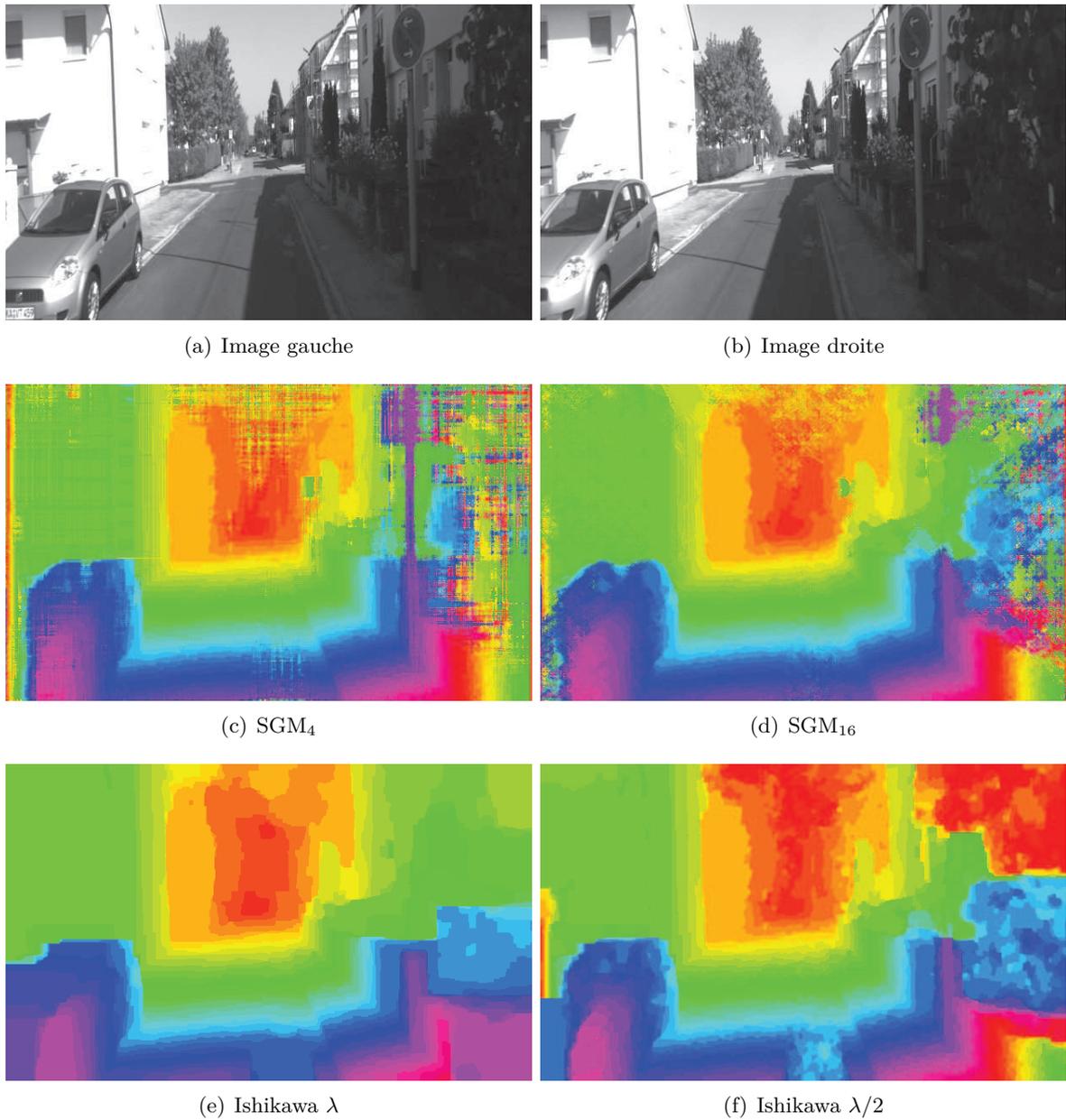


Figure 3.22 – Couple 2 de la base KITTI12 sur la première ligne. Résultat de reconstruction avec SGM₄ et SGM₁₆ sur la deuxième ligne, et par la méthode d’Ishikawa avec le même paramètre de régularisation puis avec le paramètre λ divisé par 2.

Méthode	Énergie	Erreur KITTI
ML	41 105 898	9.57%
SGM4	39 619 052	3.15%
SGM8	37 277 431	3.33%
SGM16	37 263 882	3.21%
Alterné-C	27 030 832	16.16%

Tableau 3.3 – Résultats chiffrés pour la paire 199 de la base KITTI15. Les solutions sont présentées dans la Figure 3.23.

Méthode	Err.K	
	$(P1, P2) = (3, 75)$	$(P1, P2) = (20, 300)$
Alterné-C	3.0%	23.17%
SGM4	2.88%	1.94%

Tableau 3.4 – Résultats chiffrés pour la paire 199 de la base KITTI15. Les paramètres de régularisation sont choisis pour optimiser le critère err.K, respectivement avec Alterné-C et SGM4.

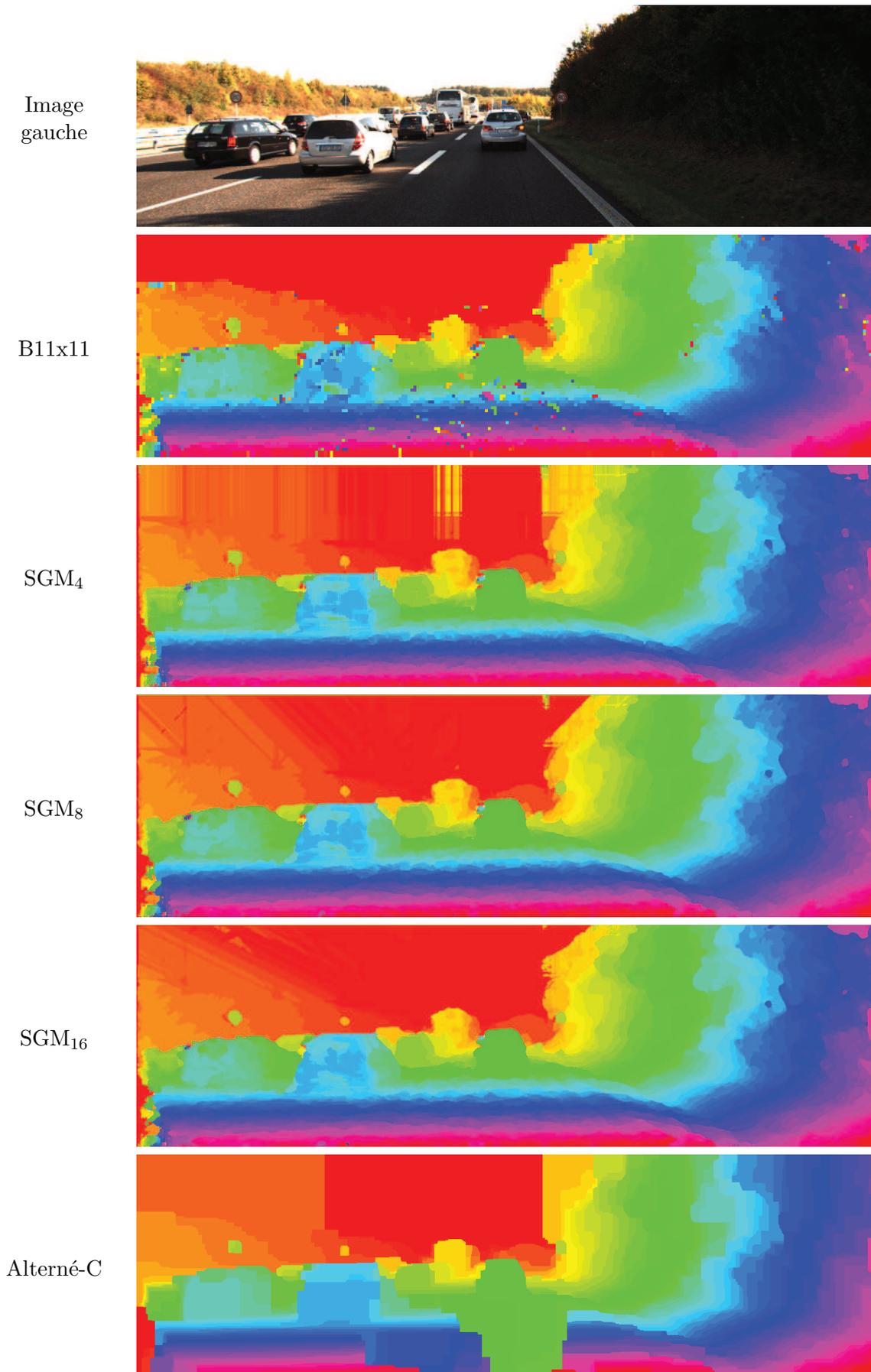


Figure 3.23 – Image gauche et cartes de disparité obtenues sur la paire 199 de la base KITTI15. Les résultats sont donnés de haut en bas par énergie globale en 4-connexité décroissante. Les méthodes sont de haut en bas, B11x11, SGM₄, SGM₈, SGM₁₆ et par GC (schéma Alterné-C).

En dernier lieu, nous montrons dans la Figure 3.22 que la solution obtenue par SGM possède des caractéristiques différentes de la solution minimale de l'énergie 4-connexité. Les images du couple 2 de la base KITTI12 ont été sous échantillonnées horizontalement d'un facteur 2. Le modèle est une fonction d'attache aux données est Census calculé sur une fenêtre 5x5 pixels et la fonction de régularisation est la valeur absolue des différences de disparités. Nous observons que la chaussée est bien reconstruite par SGM₄ et SGM₁₆, sauf dans la partie droite de l'image. La solution minimale de l'énergie globale en 4-connexité est sur-régularisée et la chaussée est partiellement mal reconstruite. Nous avons vu que le rapport entre nombre de termes de régularisation et nombre de termes d'attache aux données est de 2 pour l'énergie 4-connexité et de 1 pour SGM. Ainsi, nous avons généré le résultat en divisant le paramètre λ par 2. La solution minimale de l'énergie globale en 4-connexité parait plus proche de la solutions SGM, mais la chaussée est encore mal reconstruite. En conclusion, dans certaine cas, la solution SGM peut être plus intéressante que le minimum de l'énergie en 4-connexité. En conclusion, ces résultats suggèrent que l'énergie optimisée par SGM est de nature relativement différente de l'énergie en 4-connexité.

3.3.3 Temps de calcul

L'un des principaux avantage de la méthode SGM sur les méthodes par GC est son exécution est rapide. De plus, des implémentations rapides de l'algorithme sont possibles [Arndt *et al.*, 2013]. Le code C++ que nous avons écrit pour la méthode SGM est très peu optimisé : l'image est parcourue pixel par pixel autant de fois qu'il y a de directions et cela sans distribuer le calcul. Enfin, les opérations de sélection du minimum lors de l'accumulation des valeurs sont réalisées par de simples boucles sur les disparités. Pour autant, dans le cas d'images de grande taille avec de nombreuses valeurs de disparité, nous observons un net gain de SGM en termes de temps d'exécution en comparaison avec les autres méthodes. Sur les images de la base Middlebury, l'agrégation dans quatre directions prend en moyenne 48s, quand une itération du schéma Alterné-C prend en moyenne 330s. Dans la pratique, la valeur du paramètre de régularisation λ a un impacte sur le temps d'exécution pour les méthodes fondées sur GC. En effet la zone d'influence d'un pixel de la solution est plus grande ce qui rend plus coûteux l'opération de poussage de flot. Dans le cas de SGM, la modification du paramètre λ impact pas ou peu le temps d'exécution. Pour un couple d'image de la base KITTI12 sous-résolu selon la direction horizontale d'un facteur 2 (soit une taille de 621x376 pixels et 60 valeurs de disparité) le temps d'exécution est de 2,2s et 2,2s pour SGM₄ avec $\lambda = 2$ et $\lambda = 20$, respectivement. Le temps d'exécution est de 12s et 168s pour un cycle du schéma Alterné-C, et il est de 122s et 18394s pour la méthode d'Ishikawa, avec $\lambda = 2$ et $\lambda = 20$ respectivement. Ces temps d'exécution sont obtenus sur un cœur de la carte Intel(R) Xeon(R) CPU E5506 @ 2.13GHz.

4 Conclusion

L'optimisation d'un champ de Markov 2D est un problème difficile. Dans la méthode SGM, un problème alternatif est résolu : optimiser un champ de Markov composé d'alignements autour de chaque pixel. L'optimisation est réalisée efficacement selon les différentes directions par DP. Le nombre de directions prises en compte a un impact sur le résultat. En particulier, lorsque le nombre de directions est grand, la couverture de l'image augmente, et la solution est plus lisse. Des variantes de SGM ont été proposées : CAT et MGM. Ces deux méthodes prennent en compte deux directions lors de l'étape d'agrégation des coûts pour obtenir des solutions plus lisses. Par construction, ces méthodes ont tendance à diminuer l'importance des alignements par rapport aux chemins qui "zigzaguent". Nous avons donc proposé des modifications de ces méthodes pour limiter cet effet, en privilégiant une directions sur une autre.

Lors de la comparaison des résultats, nous avons observé que les résultats sont dépendants du jeu de données. Par exemple, sur la base Middlebury, passer de quatre à huit directions dans

SGM améliore les résultats, mais la solution est détériorée lorsque l'on passe à seize directions, alors que sur la base KITTI, passer à seize directions améliore les résultats. De manière générale, les variantes de SGM donnent lieu à des améliorations sur la base Middlebury, mais pas sur la base KITTI. Cette différence peut venir du fait que les scènes routières sont dominées par des objets horizontaux et verticaux. Comme toutes les variantes de SGM prennent en compte l'information dans d'autres directions, elles diminuent de fait l'importance de la direction horizontale et verticale.

SGM est souvent présenté comme une alternative à GC pour approcher le minimum de l'énergie en 4-connexité. Cependant l'a priori inclus dans le modèle n'est pas le même. Dans les méthodes par GC, la régularisation est selon la 4-connexité. Cela correspond à l'a priori d'une solution contenant des zones fronto-parallèles. Dans la méthode SGM_4 , la régularisation est selon la verticale et l'horizontale. Ainsi l'a priori est que la solution est composée de segments verticaux et horizontaux fronto-parallèles. Il en résulte des solutions dont les caractéristiques sont différentes. Les méthodes par GC obtiendront plus facilement des solutions lisses, et SGM_4 sera capable de reconstruire des objets selon la verticale et l'horizontale. Ainsi, l'énergie optimisée par SGM possède de meilleures capacités d'interpolation à disparité constante selon la verticale et l'horizontale grâce à une optimisation efficace dans ces directions, ce qui permet de reconstruire plus souvent correctement la chaussée.

Chapitre 4

Stabilité de la solution par optimisation discrète et application à reconstruction stéréoscopique

La solution d'un problème image est souvent formulée comme une affectation minimale vis à vis d'une énergie, comme c'est le cas pour la reconstruction stéréoscopique. Toutefois, la seule connaissance de la solution optimale de l'énergie n'est pas suffisante pour une bonne intégration dans la chaîne de traitements. La solution doit alors être complétée par une caractérisation de sa stabilité. Dans la Section 1, nous verrons que le calcul direct de cette caractérisation est trop compliqué dans la pratique. Par contre, des indices plus ou moins partiels peuvent être proposés, qu'il convient de valider empiriquement. Nous proposerons donc, dans la Section 2, des indices de stabilité spécifiques à deux méthodes d'optimisation discrètes : la programmation dynamique et la coupure de graphe. Ces indices ont été conçus et testés sur des problèmes de dé-bruitage et de reconstruction stéréoscopique. Dans la Section 3, nous comparerons les performances des indices et nous décrirons les améliorations permises par la prise en compte de ces indices dans un raffinement de la solution.

1 Indices et méthodes d'évaluation de l'indice

Lorsque le problème est bien posé, la solution est unique et dépend des données de manière continue. Cette solution peut être complétée par des informations de stabilité qui caractérisent l'effet sur la solution d'une perturbation sur les données. La caractérisation de la stabilité ne peut pas prendre toutes les perturbations imaginables en entrée. Par contre pour un modèle de perturbation en entrée, dans certain cas, une caractérisation comme la matrice de covariance de la solution peut être obtenue explicitement. Dans la Section 1.1, nous présenterons quelques cas où des éléments de la matrice de covariance de la solution peuvent être estimés. Mais, ce calcul n'est pas réalisable en général, d'autant plus que le problème n'est pas souvent bien posé. Face à cette difficulté, des indices partiels peuvent être proposés pour évaluer la stabilité de la solution. Comme ces indices sont difficiles à justifier théoriquement, leur performance ne peut être évaluée qu'empiriquement, ce qui est souvent aussi assez difficile. Nous présenterons, dans la Section 1.2, comment procéder pour évaluer les performances de tels indices.

1.1 Estimation locale de la variance

La matrice de covariance de la solution $\hat{\mathbf{l}}$ est définie par la formule suivante :

$$\Sigma_{\hat{\mathbf{l}}} = \mathbb{E} \left(\left(\hat{\mathbf{l}} - \mathbb{E}(\hat{\mathbf{l}}) \right)^t \left(\hat{\mathbf{l}} - \mathbb{E}(\hat{\mathbf{l}}) \right) \right) \quad (4.1)$$

Dans le cas général, cette formule demande de pouvoir évaluer toutes les solutions et leurs probabilités. Sur un problème discret et du fait de la combinatoire, cette approche n'est pas

réalisable en pratique. Dans la Section 1.1.1, nous verrons, sur le problème de dé-bruitage, le cas où le modèle est supposé linéaire avec un bruit gaussien. Le calcul de la matrice de covariance est simplifié et elle peut servir à caractériser la stabilité de la solution. Dans la Section 1.1.2, nous verrons que dans le cas non-linéaire, l'estimation de la matrice de covariance est difficile. De plus, si le problème est mal-posé, l'utilisation de la matrice de covariance est problématique et il vaut mieux travailler avec son inverse. C'est pour cela qu'on lui préfère souvent une caractérisation locale.

1.1.1 Dans le cas du dé-bruitage avec un bruit gaussien

Nous formulons le problème linéaire du dé-bruitage de la même manière que dans la Section 1.1 du Chapitre 2, où \mathbf{l}^b est l'image bruitée et $\hat{\mathbf{l}}$ l'image estimée. Nous considérons \mathbf{B} un bruit additif gaussien centré de matrice de covariance Σ_B que nous supposons inversible. De plus, on suppose un a priori sur la solution lui aussi gaussien de la forme :

$$\mathbb{P}(\mathbf{l}) \propto e^{-\frac{1}{2}\mathbf{l}^t \Pi_l \mathbf{l}} \quad (4.2)$$

En considérant que les résidus suivent la distribution du bruit, après application du principe du MAP, la solution est définie comme l'affectation qui maximise la probabilité suivante :

$$\mathbb{P}(\hat{\mathbf{l}}|\mathbf{l}^b) \propto e^{-\frac{1}{2}(\hat{\mathbf{l}}-\mathbf{l}^b)^t \Sigma_b^{-1}(\hat{\mathbf{l}}-\mathbf{l}^b)} e^{-\frac{1}{2}\hat{\mathbf{l}}^t \Pi_l \hat{\mathbf{l}}} \quad (4.3)$$

Alors, $\hat{\mathbf{l}}$ est la solution du système linéaire suivant :

$$(\Sigma_b^{-1} + \Pi_l)\hat{\mathbf{l}} = \Sigma_b^{-1}\mathbf{l}^b \quad (4.4)$$

La solution du problème dépend linéairement des données. L'effet d'un bruit additif sur les données donne lieu à un bruit additif sur la solution et ces bruits dépendent aussi linéairement l'un de l'autre. Dans nos hypothèse, la variance de \mathbf{l}^b est la variance du bruit, ainsi par propagation de la variance, on obtient :

$$\Sigma_{\hat{\mathbf{l}}} = (Id + \Sigma_b \Pi_l)^{-t} \Sigma_b (Id + \Sigma_b \Pi_l)^{-1} \quad (4.5)$$

La covariance de la solution ne dépend pas de la solution, mais uniquement de la matrice de covariance du bruit et de l'a priori. L'estimation de la variance demande de réaliser seulement une inversion matricielle et des multiplications matricielles ce qui représente une grande simplification par rapport au cas général. Elle ne dépend pas des valeurs de l'image en entrée, ainsi ce calcul peut être réalisé une fois pour toutes, pour un système donné. Géométriquement, cette matrice de covariance est inversement reliée à la courbure de l'énergie. Bien que les matrices manipulées soient de grande taille, elles sont souvent creuses, ce qui peut être exploité pour réduire la quantité des calculs et la taille de la représentation de cette matrice.

1.1.2 Dans le cas général

Lorsque le bruit est Gaussien, la formulation bayésienne conduit à une énergie quadratique et la solution dépend linéairement des données. Dans ce cas, évaluer localement la courbure de l'énergie permet de retrouver l'inverse de la matrice de covariance de la solution. Cependant, la plupart des modèles utilisés en traitement d'image ne sont pas simples. Le calcul explicite de la matrice de covariance est beaucoup trop coûteux et des méthodes alternativement doivent être mises en œuvre.

La distribution de probabilité de la solution peut être estimée par la méthode de Monte-Carlo. La méthode de Monte-Carlo consiste à étudier la sortie d'un système en simulant les perturbations sur les données en entrée. A partir de l'hypothèse de bruit faite dans le modèle, des données simulées sont générées et les solutions sont estimées pour chaque jeu de données. Avec un assez grand tirage de solutions, il est possible d'approcher la distribution de la sortie.

Dans le cas où le problème est bien posé, la densité du tirage tend vers la distribution vraie. Toutefois, cette convergence est lente et la plupart des problèmes que nous traitons sont mal posés. Le nombre de résolutions du problème nécessaire pour obtenir une bonne estimation étant très élevé, cette méthode ne peut pas être mise en œuvre sur des problèmes trop volumineux.

Lorsque la probabilité de la sortie n'est pas gaussienne, les valeurs données par la matrice de covariance sont difficiles à interpréter, notamment dans le cas où la probabilité est multimodale et qu'il existe de nombreux minima locaux dans l'énergie. Dans la pratique, nous obtenons à l'issue de l'optimisation le minimum de l'énergie ou à défaut, un minimum local, éventuellement approché. Une approche consiste à considérer que le problème est localement bien-posé et que pour des petites perturbations des données, l'effet sur la solution est linéaire. Ainsi, en linéarisant l'énergie autour de la solution, si l'énergie est dérivable, la solution $\hat{\mathbf{l}}$ est formulée comme la solution d'un problème linéaire, $A\hat{\mathbf{l}} = B$. En appliquant la propagation de la variance, nous obtenons la matrice de covariance approchée $\Sigma_{\mathbf{l}} = A^{-t}\Sigma_B A^{-1}$, la sorte étant supposée gaussienne.

1.2 Évaluation d'un indice de stabilité

Dans le contexte de la reconstruction, des indices ont été proposés pour caractériser la solution. Ces derniers pouvant être utilisés pour détecter les erreurs dans la solution, pour détecter les occultations. Se pose alors la question de l'évaluation de ces indices. Nous rappelons des méthodes pour évaluer les performances de ces indices, lorsque l'indice cherche à prédire le résultat d'un test binaire (Section 1.2.1) et lorsque l'indice cherche à estimer la variance de la solution (Section 1.2.2). Les indices peuvent aussi être validés indirectement en montrant une amélioration des résultats lorsque qu'ils sont pris en compte lors d'une étape de raffinement de la solution (Section 1.2.3).

1.2.1 Évaluation d'un détecteur binaire

Considérons une hypothèse, par exemple : "le pixel est occulté". Les pixels peuvent être classés en deux catégories : les pixels pour lesquels l'hypothèse est vraie et les pixels pour lesquels l'hypothèse est fausse. On suppose que l'on a un estimateur binaire ou détecteur qui indique dans quelle catégorie se trouve chacun des pixels. On dit que le détecteur est positif s'il indique que le pixel appartient aux pixels où l'hypothèse est vraie et qu'il est négatif s'il indique que le pixel appartient aux pixels où l'hypothèse est fausse. Ce détecteur peut commettre des erreurs, de sorte que les pixels peuvent être classés en quatre catégories :

- Les vrais positifs (VP) : pixels vrais où le détecteur répond positivement, exemple : "Le pixel occulté est indiqué comme occulté".
- Les faux positifs (FP) : pixels faux où le détecteur répond positivement, exemple "Le pixel non-occulté est indiqué comme occulté"
- Les faux négatifs (FN) : pixels vrais où le détecteur répond négativement, exemple "Le pixel occulté est indiqué comme non-occulté"
- Les vrais négatifs (VN) : pixels faux où le détecteur répond négativement, exemple "Le pixel non-occulté est indiqué comme non-occulté".

comme cela est montré dans la Figure 4.1. La matrice de confusion est composée du nombre d'occurrences de ces quatre catégories : VP, FP, FN, VN respectivement (Table 4.1). De nombreuses mesures de la performance du détecteur ont été proposées. Parmi elles, le rappel est la proportion des pixels vrais pour laquelle le détecteur répond positivement :

$$\text{Rappel} = \frac{VP}{VP + FN} \quad (4.6)$$

et la précision est la proportion des pixels vrais parmi les pixels où le détecteur répond positivement :

$$\text{Précision} = \frac{VP}{VP + FP} \quad (4.7)$$

Occurrence		Vérité	
		Vrai	Faux
Détecteur	Positif	VP	FP
	Négatif	FN	VN

Tableau 4.1 – Matrice de confusion

Lorsque le rappel est faible, on dit qu'il y a sous-détection. Lorsque la précision est faible, on dit qu'il y a sur-détection. Dans le pratique, il est difficile d'obtenir un détecteur qui ait à la fois un bon rappel et une bonne précision. Ainsi la recherche d'un bon détecteur est souvent formulée comme un compromis entre ces deux valeurs.

L'indice de stabilité de la solution n'étant pas binaire, il est nécessaire de le binariser pour en faire un détecteur et pouvoir utiliser ce type d'évaluation. Les performances de l'indice vont alors être dépendantes du seuil choisi lors de la binarisation. L'effet de ce seuillage peut être observé sur la courbe Rappel/Précision : la précision est tracée en fonction du rappel en faisant varier la valeur du seuil. Cette courbe peut être résumée en prenant l'aire sous la courbe (*Area Under the Curve*, AUC).

Dans le cas où l'indice est un indice d'erreur, une courbe alternative peut être calculée : le nombre de pixel faux lorsque l'on a retiré les n pixels où l'indice est le plus élevé, en fonction de n . Cette courbe revient à tracer les faux négatifs (FN) en fonction de la somme des vrais et des faux positifs (TP+FP).

Une forte contrainte de ces approches est qu'il est nécessaire de connaître une vérité de ce que mesure le détecteur ou l'indice binarisé. A défaut, une estimée de cette vérité peut être utilisée. Les résultats sont alors donnés à l'erreur de cette estimée près.

1.2.2 Indice de la variance de la solution

Déjà pour les problèmes non-linéaires, l'estimation directe de la variance de la solution est souvent trop difficile et coûteuse en calculs. Ainsi, il peut être intéressant de calculer un indice qui approche cette valeur. Se pose alors la question de l'évaluation d'un tel indice. Dans le contexte de la reconstruction stéréoscopique, contrairement au cas de l'erreur de la solution, il est fort difficile d'avoir une vérité de l'incertitude sur la solution avec laquelle il est possible de comparer l'indice. Nous montrons, dans cette section, quels tests peuvent néanmoins être réalisés sur cet indice.

En reconstruction stéréoscopique, nous disposons d'une solution particulière et d'une carte de disparité vraie. Cette solution est une occurrence de la loi de probabilité dont nous cherchons à estimer la variance. Nous disposons donc de peu d'information pour estimer une distribution en grandes dimensions. Aussi, il est nécessaire de faire des hypothèses sur la distribution de cette loi : nous considérons que cette loi est gaussienne indépendante centrée sur la vérité. Les écarts normalisés suivent alors une loi normale centrée $p(\frac{e_i}{\sigma_i}) \sim N(0, 1)$. On suppose que l'on a un estimateur $\hat{\sigma}$ de l'écart-type σ à un facteur près. Ce facteur peut être estimé par l'estimateur du facteur de variance :

$$\hat{\sigma}_0^2 = \frac{e_i^2}{\hat{\sigma}_i^2} \quad (4.8)$$

Si le nombre de valeurs prises en compte est élevé, nous pouvons considérer que l'indice normalisé suit une loi normale d'écart-type 1. Du fait du grand nombre d'hypothèses et d'effet qui n'ont pas été pris en compte comme la discrétisation, un test statistique rigoureux a peu de chance d'être validé. Nous proposons d'observer la distribution des ratio erreur/indice normalisé par le rapport de l'écart type de l'erreur et de l'indice.

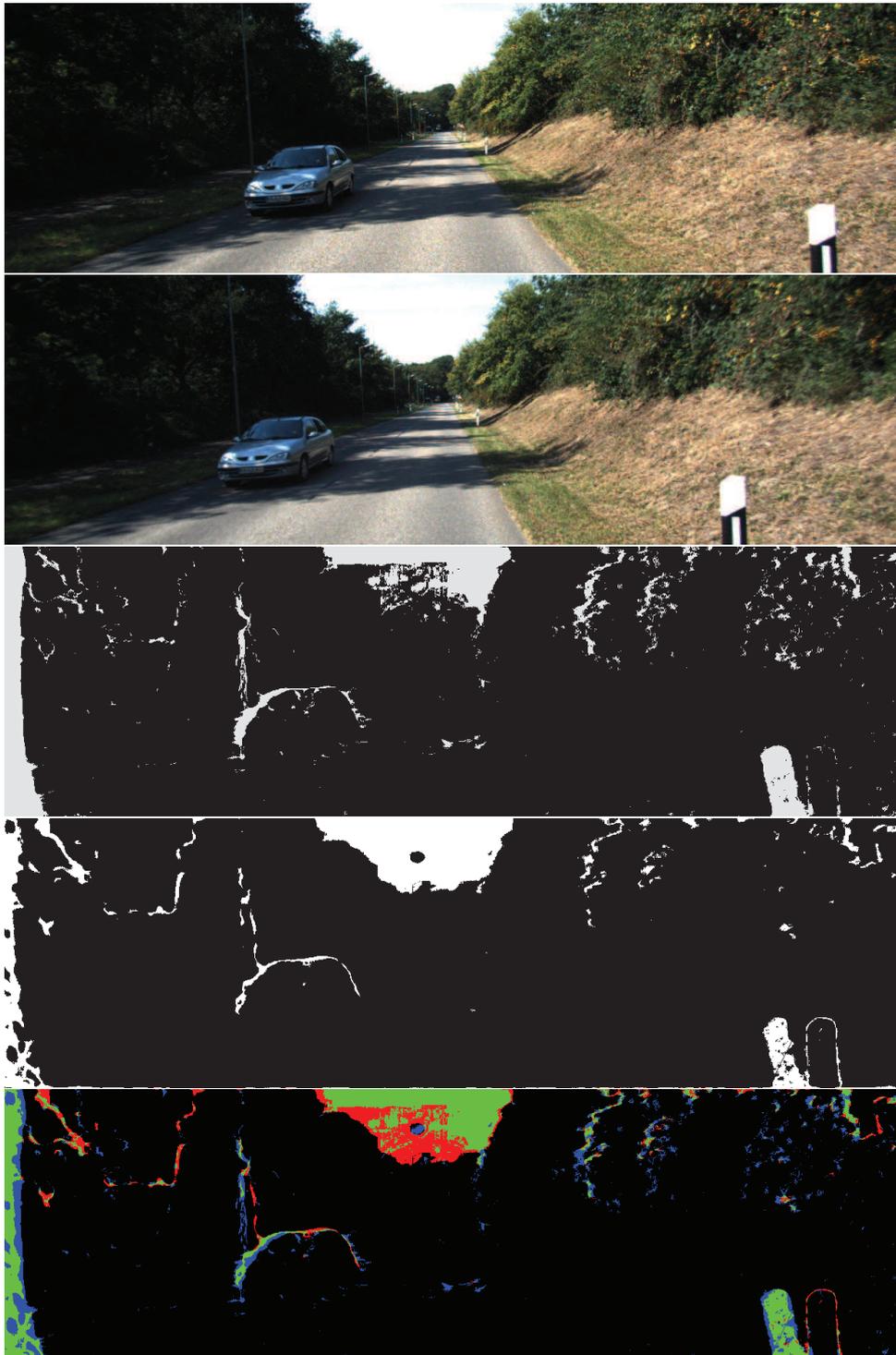


Figure 4.1 – Pour le couple 56 de la base KITTI15 affiché sur les deux premières images, la vérité d'occultation est obtenue par validation croisée (cf. Section 2.4.3 du Chapitre 2) et affichée en troisième position : en blanc, les pixels dont la disparité n'est pas cohérente entre les deux cartes de disparités. Cette vérité est comparée à ce qui est obtenu par un seuillage à 10 de l'indice de stabilité de la solution. Cette carte est affichée en quatrième position. Dans la dernière image, les vrais positifs (VP) sont en **vert**, les faux positifs (FP) en **rouge**, les faux négatifs (FN) en **bleu** et enfin, les vrais négatifs (VN) en noir.

1.2.3 Validation indirecte

Les indices de stabilité cherchent à estimer des informations, qui peuvent être prises en compte dans une étape de raffinement de la solution afin d'améliorer le résultat. Considérons un problème où le modèle n'est pas valide sur toute l'image et où nous disposons d'un indice qui évalue l'adéquation des données avec le modèle. En appliquant un modèle alternatif plus adapté là où l'adéquation est faible, le résultat doit, en principe, être amélioré. L'amélioration observée peut être utilisée pour valider indirectement cet indice. De nombreuses méthodes ont exploité le résultat de la validation croisée (Section 2.4.3 Chapitre 2) soit pour filtrer les résultats [Mei *et al.*, 2011; Sun *et al.*, 2011; Banno et Ikeuchi, 2011], soit pour modifier l'attache aux données [Klaus *et al.*, 2006; Gong et Yang, 2007; Xu et Jia, 2008].

2 Proposition d'indice de confiance par optimisation discrète

De nombreux indices de confiance ont été proposés dans le contexte de la reconstruction stéréoscopique [Hu et Mordohai, 2012]. Chacun de ces indices pris séparément donne souvent des résultats imparfaits. Ainsi, les approches par apprentissage ont cherché à construire des indices qui prédisent l'erreur à partir d'un ensemble de caractéristiques. Par exemple, dans [Haeusler *et al.*, 2013; Spyropoulos *et al.*, 2014; Park et Yoon, 2015], l'indice est appris par forêt aléatoire à partir d'informations extraites des données et d'une carte de profondeur approchée. Dans [Seki et Pollefeys, 2016], l'indice est appris par apprentissage profond à partir d'informations issues des images, sans faire intervenir de solution approchée. Ces approches donnent de bons résultats, cependant, l'étape d'apprentissage est assez contraignante dans la pratique : elle est longue et nécessite des bases de données suffisamment grandes et variées pour lesquelles la vérité de reconstruction est connue. Dans cette section, nous cherchons à exploiter les calculs intermédiaires réalisés au cours de l'étape d'optimisation. Les indices que nous proposons dans cette section sont fondés sur l'énergie issue du modèle. Ils ne sont pas construits pour prédire des erreurs, comme les indices précédemment cités, mais sont construits pour être comme des indicateurs de la stabilité de la solution. Dans la section 2.1, nous montrons comment des indices peuvent être calculés grâce aux propriétés de la programmation dynamique. Dans la section 2.2, nous étudions les indices de stabilité dans le cas des méthodes fondées sur la coupure de graphe.

2.1 Proposition d'indice de confiance par programmation dynamique

L'optimisation d'une énergie par programmation dynamique passe par l'accumulation de coûts à l'issue de laquelle chaque affectation de chaque label est associée à un coût. Cette valeur peut être formulée comme le minimum d'une certaine énergie. Nous calculons des indices à partir de ces valeurs dans la Section 2.1.1. Dans la Section 2.1.2, nous explicitons la valeur de l'indice dans le cas du dé-bruitage avec un bruit gaussien.

2.1.1 Construction d'un indice à partir des messages

Pour rappel, nous avons vu dans le Chapitre 3, que la méthode SGM fait intervenir la programmation dynamique (DP). La résolution d'un problème par DP passe par le calcul de valeurs accumulées Acc qui sont sommées pour donner de nouveaux coûts noté $SGM_p(d)$. La résolution du problème par DP passe par la sélection du minimum de ce coût individuellement pour chaque pixel. Nous avons donc un profil d'énergie par pixel qui peut nous donner une information sur l'ambiguïté de la solution de SGM. Si le profil possède de nombreuses valeurs proches du minimum du profil, alors la solution du problème en ce pixel est ambiguë. Différentes valeurs peuvent être calculées à partir de ce profil. Nous proposons, dans [Paget *et al.*, 2017], comme indice de stabilité, le nombre de valeurs pour lesquelles le coût est en dessous d'un seuil défini par rapport au coût minimal en ce pixel. Soit l_p^{opt} une valeur du label l_p pour laquelle

$SGM_p(l_p)$ est minimal, l'indice est calculé à l'aide de la formule suivante :

$$\text{Stab}_p^{SGM} = \sum_l \mathbf{1}(SGM_p(l) - SGM_p(l_p) \geq T) \quad (4.9)$$

Cet indice est proche de la mesure de perturbation définie dans [Merrell *et al.*, 2007] et appliquée à SGM dans [Haeusler *et al.*, 2013] :

$$\text{Perturbation}_p^{SGM} = \sum_{l \neq l_p^{opt}} e^{-\frac{(SGM_p(l) - SGM_p(l_p^{opt}))^2}{s^2}} \quad (4.10)$$

avec s un paramètre que nous réglerons à la valeur du seuil T utilisé dans l'indice de stabilité (4.9). Une autre mesure, proposé par l'auteur de l'article, est de calculer l'entropie de l'énergie :

$$\text{Entropie}_p^{SGM} = - \sum_l \mathbb{P}_p^{SGM}(l) \log \left(\mathbb{P}_p^{SGM}(l) \right) \quad (4.11)$$

avec $\mathbb{P}_p^{SGM}(l) = e^{-SGM_p(l)} / \sum_{l'} e^{-SGM_p(l')}$.

D'autres indices exploitent la cohérence entre les coûts accumulés sur chaque "bras" de la méthode SGM dans les ϱ directions. Pour rappel, nous avons vu dans le Chapitre 3, que les coûts SGM sont définis dans [Hirschmüller, 2008] comme :

$$SGM^*(p, k) = \sum_{\varrho \in \varrho} L_p^\varrho(k) \quad (4.12)$$

Nous avons préféré une formulation alternative de SGM, pour être cohérent avec la minimisation d'un champ de Markov avec un coût de la forme :

$$SGM_p(k) = D_p(k) + \sum_{\varrho \in \varrho} Acc_p^\varrho(k) \quad (4.13)$$

Il existe une relation entre les valeurs accumulées Acc^ϱ et les valeurs L_p^ϱ

$$L_p^\varrho(l) = k_p + Acc_p^\varrho(l) + D_p(l) \quad (4.14)$$

où k_p est une constante. Dans [Drory *et al.*, 2014], la méthode SGM est reformulée dans le cadre du TWR-S ([Kolmogorov, 2006]) ce qui permet de définir une borne inférieure du coût minimal SGM en chaque pixel, donnée par :

$$\sum_{\varrho \in \varrho} \left\{ \min_{l_p} \left\{ L_p^\varrho(l_p) - \frac{|\varrho| - 1}{|\varrho|} D_p(l_p) \right\} \right\} \leq \min_{l_p} \left\{ \sum_{\varrho \in \varrho} \left\{ L_p^\varrho(l_p) \right\} - (|\varrho| - 1) D_p(l_p) \right\} \quad (4.15)$$

avec, pour rappel, $|\varrho|$ le nombre de direction dans ϱ . L'inégalité peut facilement être vérifiée : il suffit de changer l'ordre des opérations somme et minimum. En substituant les valeurs L_p^ϱ par les valeurs Acc^ϱ , grâce à la relation (4.14), l'inégalité dévient :

$$\sum_{\varrho \in \varrho} \left\{ \min_{l_p} \left\{ Acc_p^\varrho(l_p) + \frac{1}{|\varrho|} D_p(l_p) \right\} \right\} \leq \min_{l_p} SGM_p(l_p) \quad (4.16)$$

Les auteurs utilisent l'écart entre les deux termes de l'inégalité comme indice de confiance. Nous appellerons cette indice : l'indice Drory. Lorsque chaque bras est minimisé simultanément à une même valeur de disparité, alors l'écart entre les deux termes est nul.

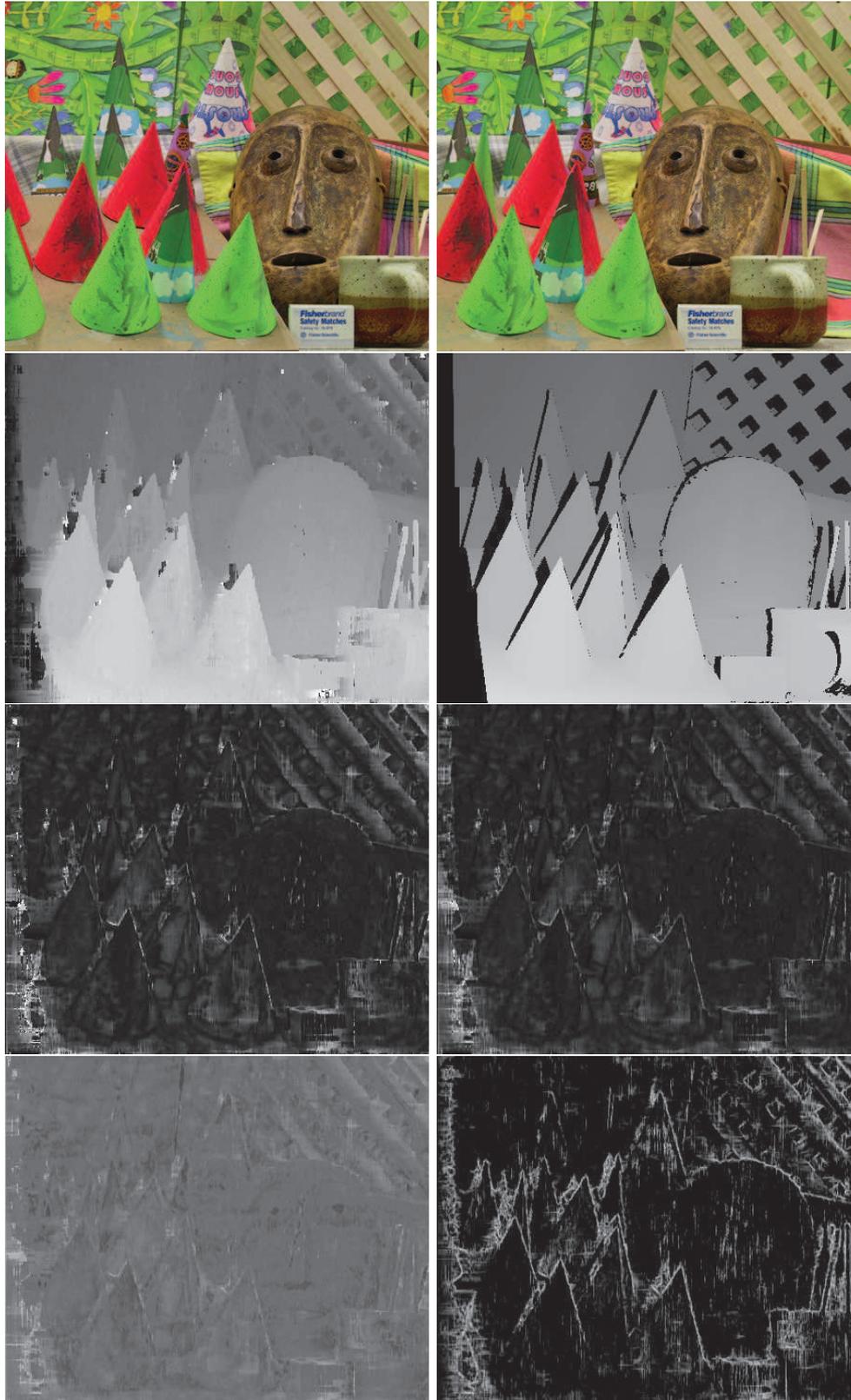


Figure 4.2 – Les différents indices pouvant être calculés au cours de la méthode SGM_4 , sont montrés pour le couple Cones de la base Middlebury qui est affiché sur la première ligne. La solution obtenue avec la méthode SGM_4 et la vérité de disparité sur les pixels non-occultés sont affichées sur la deuxième ligne, puis, l'indice de stabilité proposé et l'indice de perturbation sont sur la troisième ligne et, enfin, l'entropie et l'indice proposé par [Drory *et al.*, 2014] sur la quatrième ligne. Les valeurs d'indice sont ré-étalées pour une meilleure visualisation.

2.1.2 Valeur de l'indice SGM dans le cas gaussien

Nous cherchons à expliciter la valeur de l'indice dans le cas simple pour la méthode SGM dans le cas du dé-bruitage. Nous considérons un champ de Markov, selon différentes directions en étoile autour d'un pixel central de label l_0 constitué de $|\varrho|$ bras :

$$E(\mathbf{l}) = \frac{1}{2}(l_0 - k_0)^2 + \sum_{\varrho \in \boldsymbol{\varrho}} \frac{1}{2} \sum_{i=1}^n (l_i^\varrho - k_i^\varrho)^2 + \frac{\lambda}{2} \sum_{i=0}^{n-1} (l_i^\varrho - l_{i+1}^\varrho)^2 \quad (4.17)$$

où les k sont des constantes qui représentent le signal et l_i^ϱ est le label du i -ième pixel dans la direction ϱ par rapport au centre. Ce modèle n'est pas 2D car les labels sont différents pour chaque pixel central. L'énergie étant quadratique, le profil de l'énergie après optimisation sur les bras sera lui aussi quadratique. En utilisant la définition de l'indice de stabilité (4.9), nous avons la relation :

$$\frac{\partial^2 \min_{\mathbf{l}} E(\mathbf{l}|l_i = k)}{\partial k^2} = \frac{8T}{(\text{Stab}_i^{\text{SGM}})^2} \quad (4.18)$$

Pour évaluer le profil en fonction du label l_0 , il faut annuler toutes les dérivées partielles de l'énergie sauf celle en l_0 . Pour optimiser les l_i^ϱ , nous obtenons un système linéaire sous-déterminé avec un paramètre libre l_0 . Ainsi les différentes valeurs l_i peuvent être exprimées comme une fonction affine de l_0 :

$$l_i^\varrho = a_i^\varrho l_0 + b_i^\varrho \quad (4.19)$$

En injectant ces équations dans (4.17), et après dérivation, nous obtenons :

$$\frac{\partial^2 \min_{\mathbf{l}} E(\mathbf{l}|l_0 = k)}{\partial k^2} = 1 + \sum_{\varrho \in \boldsymbol{\varrho}} \sum_{i=1}^n (a_i^\varrho)^2 + \lambda \sum_{i=0}^{n-1} (a_i^\varrho - a_{i+1}^\varrho)^2 \quad (4.20)$$

Il suffit donc d'explicitier les valeurs a_i pour connaître la valeur de l'indice. Nous écrivons les différents labels en fonction du précédent :

$$l_{i+1} = \alpha_i l_i + \beta_{i+1} \quad (4.21)$$

Annuler la dérivée en l_i avec $i \in [1, n-1]$ s'écrit :

$$l_i - k_i + \lambda(2l_i - l_{i+1} - l_{i-1}) = 0 \quad (4.22)$$

En remplaçant l_{i+1} par son expression dans (4.21), nous obtenons :

$$\begin{cases} \alpha_i = \frac{\lambda}{1 + 2\lambda - \lambda\alpha_{i-1}} \\ \beta_i = \frac{\lambda}{1 + 2\lambda - \lambda\alpha_{i-1}} (\beta_{i+1} + \frac{k_i}{\lambda}) \end{cases} \quad (4.23)$$

Nous modifions l'énergie en l_n de sorte que les valeurs α_i soient stationnaires et égales à une valeur α . Nous avons :

$$\lambda\alpha^2 - (1 + 2\lambda)\alpha + \lambda = 0 \quad (4.24)$$

Nous voulons $\alpha \leq 1$, ainsi :

$$\alpha = \frac{1 + 2\lambda - \sqrt{1 + 4\lambda}}{2\lambda} = 1 + \frac{1 - \sqrt{1 + 4\lambda}}{2\lambda} \quad (4.25)$$

Alors les a_i de l'équation (4.19) suivent une suite géométrique de raison α et de premier terme 1. En remplaçant dans (4.20), nous obtenons :

$$\begin{aligned} \frac{\partial^2 \min_{\mathbf{l}} E(\mathbf{l}|l_i = k)}{\partial k^2} &= 1 + |\varrho| \sum_{i=1}^{n-1} (\alpha^i)^2 + |\varrho|\lambda \sum_{i=0}^{n-1} (\alpha^i - \alpha^{i+1})^2 \\ \frac{\partial^2 \min_{\mathbf{l}} E(\mathbf{l}|l_i = k)}{\partial k^2} &= 1 + |\varrho| \left(\alpha^2 + \lambda(1 - \alpha)^2 \right) \sum_{i=0}^{n-1} (\alpha^2)^i \end{aligned} \quad (4.26)$$

En faisant tendre n vers l'infini, on obtient :

$$\begin{aligned} \frac{\partial^2 \min_{\mathbf{l}} E(\mathbf{l} | l_i = k)}{\partial k^2} &= 1 + |\varrho| \frac{\alpha^2 + \lambda(1 - \alpha)^2}{1 - \alpha^2} \\ &= 1 + |\varrho| \frac{\alpha}{1 - \alpha} \end{aligned} \quad (4.27)$$

L'indice s'écrit alors :

$$\frac{(\text{Stab}_i^{SGM})^2}{8T} = \frac{1 - \alpha}{1 + \alpha(|\varrho| - 1)} = \frac{\sqrt{1 + 4\lambda} - 1}{2\lambda|\varrho| - (|\varrho| - 1)(\sqrt{1 + 4\lambda} - 1)} \quad (4.28)$$

Si λ est petit :

$$\frac{(\text{Stab}_i^{SGM})^2}{8T} \approx 1 - |\varrho|\lambda + (|\varrho|^2 + |\varrho|)\lambda^2 \quad (4.29)$$

et si λ est grand :

$$\frac{(\text{Stab}_i^{SGM})^2}{8T} \approx \frac{1}{|\varrho|\sqrt{\lambda}} \quad (4.30)$$

Nous cherchons maintenant à formuler la variance de l_0 afin de la comparer à la valeur de l'indice. Le calcul de cette variance peut être réalisé en écrivant le label l_0 comme une fonction du signal k_i . Pour cela, il faut expliciter les valeurs β de (4.23) qui suivent la relation :

$$\beta_i = \alpha \left(\frac{k_i}{\lambda} + \beta_{i+1} \right) \quad (4.31)$$

d'où

$$\beta_i = \alpha^{n-1} \beta_n + \frac{1}{\lambda} \sum_{j=i}^{n-1} \alpha^{j-i+1} k_j \quad (4.32)$$

Annuler la dérivée en l_0 demande de vérifier l'équation suivante :

$$l_0 - k_0 + \lambda \sum_{\varrho} l_0 - l_1^{\varrho} = 0 \quad (4.33)$$

En remplaçant l_1 par $\alpha l_0 + \beta_1$, nous obtenons :

$$\left(1 + \sum_{\varrho} \lambda(1 - \alpha) \right) l_0 = k_0 + \sum_{\varrho} \sum_{j=1}^{n-1} \alpha^j k_j^{\varrho} + \lambda \alpha^{n-1} \beta_n^{\varrho} \quad (4.34)$$

En faisant tendre n vers l'infini, et en considérant un variance de 1 pour chaque k , nous obtenons :

$$(1 + |\varrho|\lambda(1 - \alpha))^2 \sigma_{l_0}^2 = 1 + |\varrho| \sum_{j=1}^{\infty} (\alpha^2)^j \quad (4.35)$$

après simplification

$$\sigma_{l_0}^2 = \frac{1 + (|\varrho| - 1)\alpha^2}{((1 + |\varrho|\lambda(1 - \alpha))^2 (1 - \alpha^2))} \quad (4.36)$$

Si λ est petit :

$$\sigma_{l_0}^2 \approx 1 - 2|\varrho|\lambda \quad (4.37)$$

Si λ est grand :

$$\sigma_{l_0}^2 \approx \frac{1}{2|\varrho|\sqrt{\lambda}} \quad (4.38)$$

Pour conclure, le lien entre la valeur d'indice en un pixel et la variance de la solution en ce pixel n'est pas établi. Si la comparaison des expressions littérales n'est pas évidente, les cas limites pour λ petit et λ grand montrent que les deux valeurs que l'on souhaite comparer n'ont pas le même comportement.

2.2 Proposition d'indice de confiance par coupure de graphe

Nous avons vu que le Chapitre 1 que la coupure de graphe permet la résolution d'un problème exprimé en fonction de variables binaires. Nous nous limitons ici aux fonctions d'ordre 2 sous-modulaires de posiforme :

$$P(\mathbf{x}) = k + \sum_i c_{si}x_i + \sum_i c_{it}\bar{x}_i + \sum_{i \neq j} c_{ij}x_i\bar{x}_j. \quad (4.39)$$

Chaque variable booléenne x est représentée par un nœud dans le graphe. A l'issue du poussage de flot, les nœuds peuvent être en trois catégories : les nœuds atteignables depuis la source \mathcal{S}^* , les nœuds atteignables depuis le puits \mathcal{T}^* , et les nœuds non-atteignables \blacklozenge . L'affectation des nœuds de \mathcal{S}^* à la source, des nœuds de \mathcal{T}^* au puits et des nœuds de \blacklozenge à la source ou au puits est minimale. Nous cherchons à caractériser l'ambiguïté de ces affectations. Les nœuds \mathcal{S}^* et \mathcal{T}^* possèdent les mêmes affectations dans tous les minima globaux du problème, contrairement aux nœuds de \blacklozenge . Un indice d'ambiguïté binaire Amb peut être défini comme :

$$\begin{aligned} Amb_i^{GC} &= 1 \text{ si } \mathcal{X}_i \in \mathcal{S}^* \cup \mathcal{T}^* \\ Amb_i^{GC} &= 0 \text{ sinon.} \end{aligned} \quad (4.40)$$

En pratique, cet indice ne donne pas suffisamment d'informations. En effet, la plupart des nœuds n'est pas parfaitement ambiguë, mais reste mal déterminée. Dans la Section 2.2.1, nous verrons un indice qui mesure la stabilité de l'affectation binaire. Dans la Section 2.2.2, nous verrons comment le calcul de cet indice peut être ramené à un poussage de flot particulier. Dans la Section 2.2.3, nous proposerons plusieurs bornes inférieures afin d'approcher cet indice. Le problème traité par GC est un problème binaire, nous verrons la forme que prend cet indice dans le cas d'un problème multi-labels traité par la méthode d'Ishikawa (Section 2.2.4). Enfin, nous proposons un indice lorsque α -Expansion est utilisé pour optimiser l'énergie (Section 2.2.5).

2.2.1 Indice de stabilité

Pour définir un indice qui ne caractérise pas seulement l'ambiguïté de la solution mais sa stabilité, nous définissons un indice de stabilité de la manière suivante :

$$Stab_i^{GC} = \left| \min_{\mathbf{x}|x_i=0} P(\mathbf{x}) - \min_{\mathbf{x}|x_i=1} P(\mathbf{x}) \right| \quad (4.41)$$

Si $\mathcal{X}_i \in \blacklozenge$ alors $Stab_i^{GC} = 0$, comme c'est le cas pour l'indice d'ambiguïté. Une interprétation de l'indice est la capacité minimale nécessaire pour faire changer le nœud d'affectation. Cette indice peut être calculé sur le graphe résiduel à l'issue du poussage de flot. Pour chaque nœud \mathcal{X}_i de \mathcal{S}^* réaliser un poussage de flot avec la contrainte $x_i = 1$ et inversement pour chaque variable \mathcal{X}_i de \mathcal{T}^* réaliser un poussage de flot avec la contrainte $x_i = 0$. Cela implique $1 + \text{card}(\mathcal{S}^* \cup \mathcal{T}^*)$ poussages de flot, soit presque autant que de nœuds dans le graphe. Cependant des simplifications sont possibles. Nous allons montrer que le calcul de l'indice peut être réalisé sur deux sous-graphes. A l'issue du poussage de flot, la posiforme associée au graphe est écrite sous la forme,

$$P(\mathbf{x}) = K + Res(\mathbf{x}) \quad (4.42)$$

où K est le flot maximal et Res la posiforme résiduelle. Soit \mathbf{x}^{opt} une affectation minimale. Ainsi l'indice s'écrit :

$$Stab_i^{GC} = \min_{\mathbf{x}|x_i=\bar{x}_i^{opt}} Res(\mathbf{x}) \quad (4.43)$$

Considérons un nœud \mathcal{X}_i de \mathcal{S}^* , nous forçons l'affectation de ce nœud au puits, en ajoutant un lien infini entre ce nœud et le puits dans le graphe résiduel, et nous poussons à nouveau le flot. Cela aura pour conséquence de rendre non-atteignable depuis la source un certain nombre de nœuds (au minimum le nœud considéré). Le nouvel ensemble des nœuds \mathcal{S}_i^* est donc inclus

dans \mathcal{S}^* . Affecter tous les nœuds en dehors de \mathcal{S}_i^* au puits donne une coupe nulle dans le graphe résiduel du graphe modifié. Nous pouvons donc affecter tous les nœuds en dehors de \mathcal{S}^* avant d'opérer la modification dans le graphe résiduel. Ainsi les indices peuvent être calculés sur deux sous-graphes $Res_{\mathcal{S}}$ et $Res_{\mathcal{T}}$ qui sont le graphe Res restreint aux nœuds de \mathcal{S}^* et \mathcal{T}^* , respectivement :

$$\begin{aligned} Res(\mathbf{x}) &= Res_{\mathcal{S}}(\mathbf{x}) + Res_{\mathcal{T}}(\mathbf{x}) + Reste(\mathbf{x}) \\ Res_{\mathcal{S}}(\mathbf{x}) &= \sum_{\mathcal{X}_i \in \mathcal{S}^*} c_{si}x_i + \sum_{(\mathcal{X}_i, \mathcal{X}_j) \in \mathcal{S}^{*2}} c_{ij}x_i\bar{x}_j \\ Res_{\mathcal{T}}(\mathbf{x}_{\mathcal{T}}) &= \sum_{\mathcal{X}_i \in \mathcal{T}^*} c_{it}\bar{x}_i + \sum_{(\mathcal{X}_i, \mathcal{X}_j) \in \mathcal{T}^{*2}} c_{ij}x_i\bar{x}_j \end{aligned} \quad (4.44)$$

La posiforme $Reste$ contient les termes qui ne sont ni dans $Res_{\mathcal{S}}$ ni dans $Res_{\mathcal{T}}$: il n'intervient pas dans le calcul de l'indice. Pour simplifier, par la suite, la posiforme qui correspond au graphe coté puits est réécrite comme celle d'un graphe coté source. Les capacités vers le puits sont transformées en capacité avec la source. Les capacités des arcs entre deux nœuds sont inversées :

$$\overline{Res}_{\mathcal{T}}(\mathbf{x}_{\mathcal{T}}) = \sum_{\mathcal{X}_i \in \mathcal{T}^*} c_{it}x_i + \sum_{(\mathcal{X}_i, \mathcal{X}_j) \in \mathcal{T}^{*2}} c_{ji}x_i\bar{x}_j \quad (4.45)$$

Ainsi le problème initial devient :

$$\begin{aligned} \text{Stab}_i^{GC} &= \min_{\mathbf{x}|x_i=1} Res_{\mathcal{S}}(\mathbf{x}) \text{ si } \mathcal{X}_i \in \mathcal{S}^* \\ \text{Stab}_i^{GC} &= \min_{\mathbf{x}|x_i=1} \overline{Res}_{\mathcal{T}}(\mathbf{x}) \text{ si } \mathcal{X}_i \in \mathcal{T}^* \\ \text{Stab}_i^{GC} &= 0 \text{ si } \mathcal{X}_i \in \blacklozenge \end{aligned} \quad (4.46)$$

En conclusion, le calcul de l'indice est plus simple sur la posiforme résiduelle que sur la posiforme initiale. A l'issue du poussage flot, les nœuds sont séparés en deux ensembles qui peuvent être traités indépendamment l'un de l'autre, sur deux sous-graphes distincts.

2.2.2 Adaptation du problème de poussage de flot

Le problème du calcul de l'indice possède plusieurs formulations équivalentes. Nous avons vu que le calcul peut être restreint à des graphes où tous les nœuds sont atteignables depuis la source $Res_{\mathcal{S}}$. La première formulation consiste à ajouter un lien infini vers le puits sur le pixel considéré pour le calcul de l'indice. Cette opération est équivalente à une contraction du graphe, le nœud considéré est alors confondu avec le puits :

$$\begin{aligned} \text{Stab}_i^{GC} &= \min_{\mathbf{x}|x_i=1} Res_{\mathcal{S}}(\mathbf{x}) \\ &= \min_{\mathbf{x}|x_i=1} \sum_{\mathcal{X}_j \in \mathcal{S}^*} c_{sj}x_j + \sum_{(\mathcal{X}_j, \mathcal{X}_k) \in \mathcal{S}^{*2}} c_{jk}\bar{x}_jx_k \\ &= \min_{\mathbf{x}} c_{si} + \sum_{\mathcal{X}_j \in \mathcal{S}^* \cap \mathcal{X}_i} c_{sj}x_j + c_{ji}\bar{x}_j + \sum_{(\mathcal{X}_j, \mathcal{X}_k) \in (\mathcal{S}^* \cap \mathcal{X}_i)^2} c_{jk}\bar{x}_jx_k \end{aligned} \quad (4.47)$$

Nous avons une borne supérieure de l'indice égale à la somme des capacités des liens entrants et une borne minimale déduite de l'égalité précédente :

$$c_{si} + \sum_{\mathcal{X}_j \in \mathcal{S}^* \cap \mathcal{X}_i} \min(c_{sj}, c_{ij}) \leq \text{Stab}_i^{GC} \leq c_{si} + \sum_{\mathcal{X}_j \in \mathcal{S}^* \cap \mathcal{X}_i} c_{ji}. \quad (4.48)$$

L'approche présentée nécessite l'ajout d'une constante qui modifie la table de vérité associée à la posiforme. Mais, il est possible de formuler le problème sans modifier la table de vérité. Pour ce faire, nous définissons un nouveau graphe $Res_{\mathcal{S}}^i$ tel que :

$$Res_{\mathcal{S}}(\mathbf{x}) = c_{si}^i x_i + Res_{\mathcal{S}}^i(\mathbf{x}) \quad (4.49)$$

L'indice est alors défini comme :

$$\text{Stab}_i^{GC} = \max c_{si}^i \quad (4.50)$$

Cette valeur peut être maximisée en appliquant la quasi-somme alternée :

$$x_0 + \sum_{i=0}^{n-1} \bar{x}_i x_{i+1} = \sum_{i=0}^{n-1} x_i \bar{x}_{i+1} + x_n \quad (4.51)$$

sur la posiforme du graphe Res_S^i . Dans la pratique, cette formulation du problème revient à ajouter un lien infini entre \mathcal{X}_i et le puits et à pousser le flot. L'indice est alors égal à l'augmentation du flot. La table de vérité initiale peut être rétablie en retirant le lien infini entre \mathcal{X}_i et le puits, en ajoutant un lien entre la source et \mathcal{X}_i dont la capacité est égale à l'indice, et enfin, en retranchant la valeur de l'indice à la valeur du flot.

La procédure pour prendre en compte un changement sur les capacités qui font intervenir la source et le puits est décrite dans [Boykov et Jolly, 2001]. Dans [Kohli et Torr, 2008], la coupure dynamique de graphe permet de résoudre le poussage de flot lorsque les capacités du graphes sont modifiées. Partant du constat que si les changements sont faibles, le nombre de flot à pousser dans le graphe sont peu nombreux, de manière analogue à l'algorithme Maxflow, l'arbre de recherche est maintenu après application des changements dans le graphe. Lorsqu'une nouvelle capacité est introduite dans le graphe, certains nœuds peuvent se retrouver orphelins et sont réintégrés à l'arbre de recherche à l'issue d'une étape d'adoption. Cette technique permet de limiter le nombre d'opération à réaliser pour le calcul d'une valeur d'indice. Néanmoins, l'opération doit être répétée pour chaque calcul d'indice, ce qui limite en pratique l'utilisation de cette méthode lorsque le nombre de variable est grand.

2.2.3 Borne inférieure de l'indice

Malgré la réutilisation des arbres de recherche, le calcul de l'indice peut rester relativement long. Nous cherchons donc une borne inférieure de l'indice qui puisse être calculée rapidement. La borne inférieure donnée dans (4.48) est très locale, ainsi nous proposons deux alternatives mettant en jeu des arbres. Lorsque le graphe est un arbre, l'indice peut être calculé efficacement par programmation dynamique en partant de la racine de l'arbre, qui est soit la source soit le puits, grâce à la formule suivante :

$$\text{Stab}_i^{GC} = \min(c_{ji}, \text{Stab}_j^{GC}) \quad (4.52)$$

où \mathcal{X}_j est le nœud précédent dans l'arbre. Si le graphe n'est pas un arbre, il est possible d'élaguer le graphe afin de se ramener à un arbre. Ce type d'approche amènera à une sous-estimation de l'indice. Il peut s'agir de l'arbre de recherche construit lors du poussage de flot, ce qui nécessite très peu de calculs pour obtenir l'indice. Parmi les arbres qui peuvent être construits, il est possible d'en construire un qui maximise la valeur de l'indice pour tous les nœuds. Nous proposons un algorithme simple pour calculer ces valeurs. Il utilise deux listes de nœuds : la liste des candidats et la listes des élus. Ces nœuds sont tous associés à un score qui correspond à une borne inférieure de l'indice. L'algorithme répète alors deux étapes alternativement : l'élection et la candidature. L'élection consiste à faire entrer le candidat qui possède le plus haut score dans la liste des élus et de le retirer de la liste des candidats. Pour tous les nœuds voisins du nouvel élu, nous appliquons la candidature, il existe trois cas :

- Le voisin est dans la liste des élus. On ne fait rien.
- Le voisin est dans la liste des candidats. Nous calculons le score qui peut être propagé sur ce nœud dans la limite de la capacité du lien correspondant comme dans (4.52). S'il est supérieur à son score courant, alors nous le mettons à jour.
- Sinon, le voisin est intégré à la liste des candidats et son score est calculé de manière analogue à (4.52).

L'algorithme commence avec une liste de candidat vide et une liste d'élu qui comprend seulement la source (ou le puits) avec un score infini. Il s'arrête lorsqu'il n'y a plus de candidat dans la liste lors de l'étape d'élection.

2.2.4 Extension à la méthode d'Ishikawa

Dans le cas d'une énergie multi-labels où la régularisation est une fonction convexe de la différence de label, la construction de graphe, proposée dans [Ishikawa, 2003], permet d'obtenir le minimum global de cette énergie :

$$E(\mathbf{l}) = \sum_{i \in I} D_i(l_i) + \sum_{i < j} R_{ij}(l_i, l_j) \quad (4.53)$$

Chaque variable discrète l_i est représentée par un vecteur de variable binaires \mathbf{x}^i . La valeur d'un label est codée de la manière suivante :

$$l_i = l \Leftrightarrow \begin{cases} x_k^i = 1 & \text{si } k \geq l \\ x_k^i = 0 & \text{sinon} \end{cases} \quad (4.54)$$

L'indice peut alors s'écrire comme une différence d'énergie sous une contrainte d'inégalité sur un des labels :

$$\text{Stab}_{i,k}^{Ish} = \left| \min_{\mathbf{l} | x_k^i = 1} E(\mathbf{l}) - \min_{\mathbf{l} | x_k^i = 0} E(\mathbf{l}) \right| = \left| \min_{\mathbf{l} | l_i < k} E(\mathbf{l}) - \min_{\mathbf{l} | l_i \geq k} E(\mathbf{l}) \right| \quad (4.55)$$

Soit \mathbf{l}^{opt} une affectation optimale

$$\text{Stab}_{i,k}^{Ish} = \begin{cases} \min_{\mathbf{l} | l_i \geq k} E(\mathbf{l}) - E(\mathbf{l}^{opt}) & \text{si } k > l_i^{opt} \\ \min_{\mathbf{l} | l_i \leq k} E(\mathbf{l}) - E(\mathbf{l}^{opt}) & \text{si } k < l_i^{opt} \\ 0 & \text{sinon} \end{cases} \quad (4.56)$$

Dans [Kohli et Torr, 2008], deux contraintes sont ajoutées pour fixer la valeur d'un des labels ; Dans ce cas, les relations d'inégalité dans 4.56 deviennent des relations d'égalité :

$$\text{Kohli}_{i,k}^{Ish} = \min_{\mathbf{l} | l_i = k} E(\mathbf{l}) - E(\mathbf{l}^{opt}) \quad (4.57)$$

L'indice ainsi obtenu se rapproche des valeurs de coût dans le contexte de la programmation dynamique. Nous obtenons alors des profils d'indice pour un label donné sur lequel des indices peuvent être calculés.

2.2.5 Extension à la méthode α -Expansion

Le schéma d'optimisation approchée α -Expansion consiste à approcher successivement la solution à l'aide de fusion par expansion. A chaque fusion par expansion de paramètre α , si le nœud est affecté aux puits ($\mathcal{X}_p \in \mathcal{T}^*$) alors la valeur de label correspondante est remplacée par α . Un cycle du schéma est composé des fusions par expansion avec toutes les valeurs possibles du paramètre. L'algorithme s'arrête lorsque la solution est stable. Cela signifie que pour tout label de l'image courante, et toute valeur de paramètre α , l'affectation du nœud est soit à la source, soit ni à la source ni au puits ($\mathcal{X}_p \in \mathcal{S}^* \cup \blacklozenge$). Soit $E(\mathbf{l}^\alpha)$ l'énergie de la solution lorsque l'algorithme converge, l'indice de stabilité pour une fusion de paramètre a s'écrit :

$$\text{Stab}_i^{\alpha=a} = \min_{\mathbf{x} | x_i = 1} P^{\alpha=a}(\mathbf{x}) - E(\mathbf{l}^\alpha) \quad (4.58)$$

où $P^{\alpha=a}$ est la posiforme qui donne l'énergie des affectations issues de la fusion par expansion de paramètre a . Nous définissons alors un indice de la stabilité de la solution α -Expansion comme :

$$\text{Stab}_i^{\alpha-Exp} = \sum_a \mathbf{1}(\text{Stab}_i^{\alpha=a} \geq T) \quad (4.59)$$

où T est un paramètre de seuil. La perturbation ou l'entropie peuvent elles aussi être calculées sur chaque pixel. Nous avons vu dans le Chapitre 2, dans le cas de la reconstruction qu'une affectation qui respecte l'a priori fronto-parallèle possède souvent une énergie relativement faible du fait de l'ambiguïté des données. Ainsi, analyser la stabilité de la solution vis-à-vis de telles affectations peut donner une information intéressante sur la stabilité de la carte disparité.

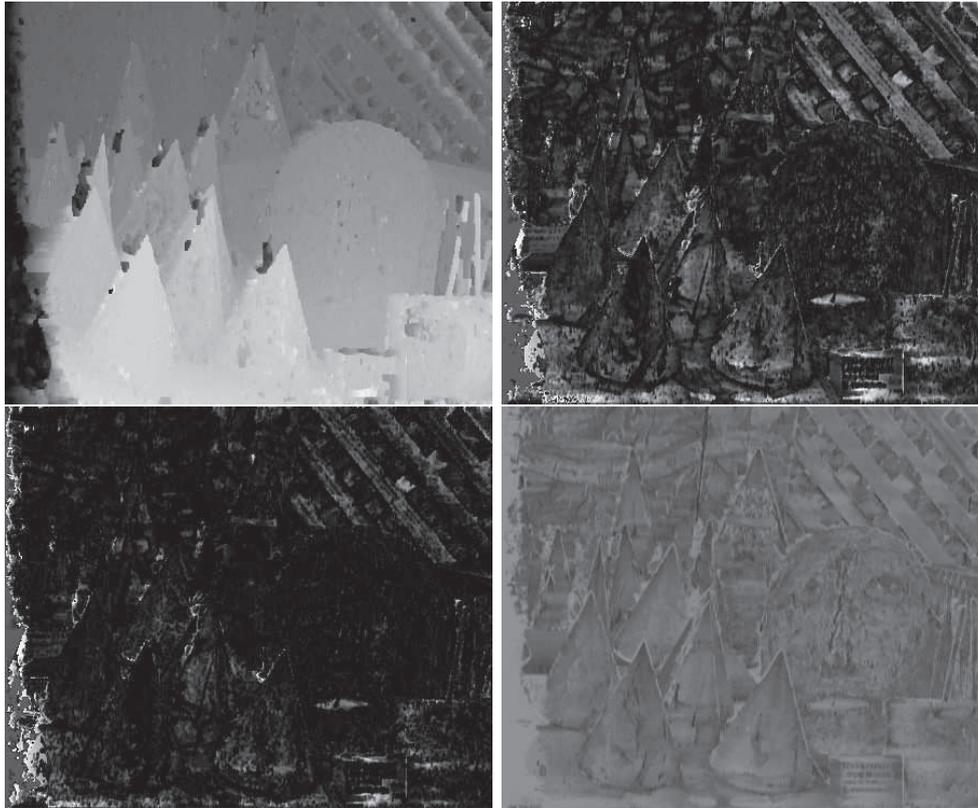


Figure 4.3 – Les différents indices proposés à l’aide de la méthode α -Expansion, sont montrés pour le couple Cones de la base Middlebury. La solution et l’indice de stabilité sont montrés sur la première ligne. L’indice de perturbation et l’entropie sont montrés sur la deuxième ligne. Les valeurs d’indice sont ré-étalées pour une meilleure visualisation.

3 Évaluation des indices

Nous avons formulé deux problèmes de traitement d’image dans le Chapitre 2 : le dé-bruitage et la reconstruction stéréoscopique. Pour l’évaluation des indices, nous écartons le problème de dé-bruitage car les énergies utilisées sont peu ambiguës comparées aux énergies utilisées pour la reconstruction stéréoscopique. Les solutions ont alors tendance à être très stables, et l’indice de stabilité n’est pas très intéressant dans ce cas. De plus, la méthode SGM appliquée au dé-bruitage ne donne pas de résultats satisfaisants. Ainsi, dans cette section, nous évaluons les performances des indices de stabilité sur le problème de reconstruction stéréoscopique. Nous présentons, dans la Section 3.1, les performances des indices issus de la programmation dynamique dans la réalisation de certaines tâches : la détection d’erreurs, la détection d’occultations et l’estimation sommaire de la variance de la solution. Dans la Section 3.2, nous réalisons les mêmes tests avec les indices issus de la coupure de graphe. Enfin, nous montrons, dans la Section 3.3, les améliorations obtenues par raffinement de la solution.

3.1 Indices issus de la programmation dynamique

Nous évaluons les indices sur deux couples de la base Middlebury, appelés Cones et Teddy, et sur les couples de la base KITTI15. Dans le modèle pour les images de la base Middlebury, la fonction d’attache aux données est un carré tronqué au-delà de 18, et la régularisation est la valeur absolue de la différence de labels. Dans le modèle pour les images de la base Middlebury, la fonction d’attache aux données est Census sur une fenêtre 5x5 pixels agrégé deux fois par CBCA, et la régularisation est la valeur absolue lissée de la différence de labels $(P1, P2) = (1.2, 23)$, comme dans le Chapitre 3. Les reconstructions sont obtenues par la méthode SGM_4 .

Dans la Figure 4.4, nous montrons les résultats lorsque l'indice est utilisé pour prédire les erreurs dans la solution. Nous évaluons ici l'indice de stabilité, l'indice de perturbation, l'entropie et l'indice de Drory. Le paramètre de seuil T est fixé à deux fois la valeur du paramètre $P2$ de la régularisation. Un pixel est considéré comme faux si sa disparité est à une différence de plus de un de la disparité vraie sur la base Middlebury et à une différence de plus de trois de la disparité vraie sur la base KITTI. Ces deux seuils sont ceux communément utilisés pour l'évaluation des résultats pour ces deux bases. Sur les couples Cones et Teddy, les différents indices donnent des résultats relativement proches. Pour un rappel de 50%, soit une omission de la moitié des erreurs, la précision est entre 81% et 85% pour le couple Cones et entre 76% et 82% pour le couple Teddy, suivant les indices. L'entropie et l'indice de Drory donnant des résultats un peu meilleurs par rapport à la perturbation et la stabilité. Sur les couples de la base KITTI15, les performances sont moins bonnes, avec, pour un rappel de 50%, une précision entre 40% et 50%. Les résultats des différents indices sont proches, sauf l'entropie qui donne les moins bons résultats, comme on peut l'observer sur la courbe de Décimation/Erreur.

La validation croisée est une méthode pour détecter les zones occultées qui donne des résultats satisfaisants lorsque les reconstructions sont d'une qualité suffisante. En plus d'une simple classification binaire, nous considérons trois catégories de pixels comme dans [Hirschmüller, 2008]. Soit d_p^1 la disparité du pixel p dans l'image I^1 et d_{p-d}^2 la disparité du pixel à la position p décalée de la disparité d dans l'image I^2 . Les classes des pixels sont définies comme :

$$\begin{aligned} & \text{si } |d_p^1 - d_{p-d}^2| \leq 1 \text{ "correct"} \\ & \text{sinon si } \exists d \neq d_p, |d - d_{p-d}^2| = 0 \text{ "non-concordant"} \\ & \text{sinon "occulté"} \end{aligned} \quad (4.60)$$

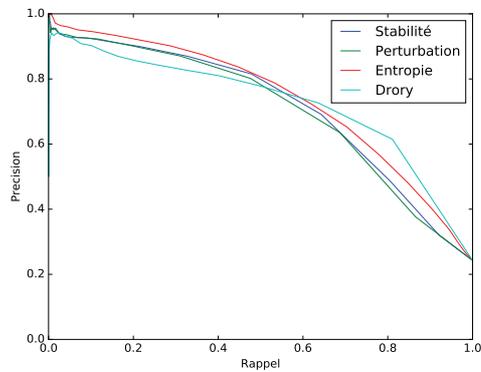
Dans [Mei *et al.*, 2011], un filtrage de la solution est appliqué en fonction de la classe attribuée au pixel :

- Les pixels "corrects" sont conservés à leur valeur de disparité.
- Les pixels "non-concordants" sont fixés à la valeur médiane des plus proches voisins "corrects" dans 8 directions.
- Les pixels "occultés" sont fixés à la valeur du pixel "correct" à gauche (hypothèse d'un appariement à l'arrière plan).

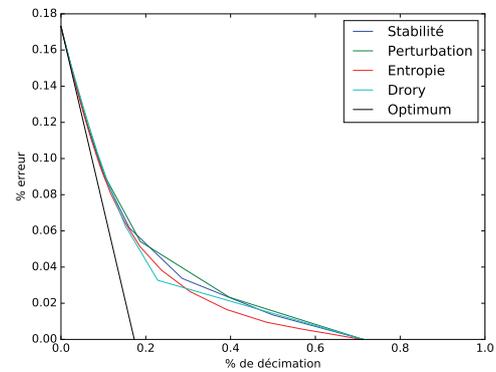
Nous appellerons cette méthode : filtrage LRC, pour *Left-Right Check*. L'inconvénient de la validation croisée est que la reconstruction stéréoscopique doit être réalisée pour les deux points de vue, contrairement aux indices que nous évaluons qui sont calculés à partir d'une seule reconstruction. Dans la base Middlebury, les vérités sont denses et les masques d'occultations sont donnés. Nous comparerons alors les performances des indices à celles de la validation croisée. Dans les courbes Rappel/Précision, nous considérons deux cas intermédiaires : les "non-concordants" sont pris en compte comme occultés ou comme corrects. Les vérités de profondeur de la base KITTI ne sont pas denses, ainsi il n'est pas possible de calculer un masque des occultations à partir de ces vérités. Nous utiliserons alors le résultat de la validation croisée comme vérité d'occultation pour évaluer les indices. Les résultats seront donc donnés, aux erreurs commises lors de la validation croisée près.

Dans la Figure 4.5, nous montrons les résultats lorsqu'un indice est utilisé pour prédire les zones occultées dans la solution. Comparé à la validation croisée, les différents indices ne sont pas en mesure de prédire correctement les occultations. En effet, la validation croisée fait surtout des sous-détections et très peu de sur-détection (rappel de 66% pour une précision de 97%), alors que pour les différents indices, pour un rappel de 50%, la précision est entre 43% et 53% pour le couple Cones et entre 35% et 38%. Dans le cas des couples de base KITTI, les performances sont comparables (précision entre 38% et 50%, pour un rappel de 50%) à la différence que les résultats sont donnés par rapport à validation croisée.

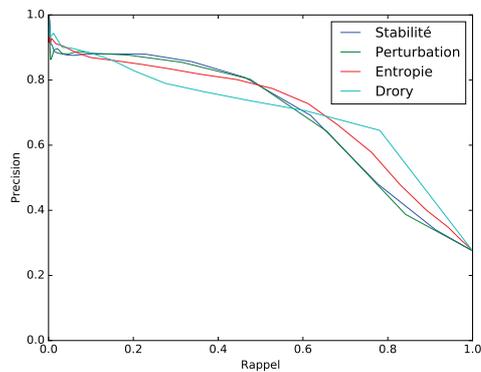
Enfin dans la Figure 4.6, nous montrons la distribution des rapports entre l'erreur normalisée par son écart-type et l'indice lui aussi normalisée par son écart-type. Si les hypothèses faites sont vérifiées, la distribution après prise en compte de l'indice doit tendre vers la loi normale d'écart-



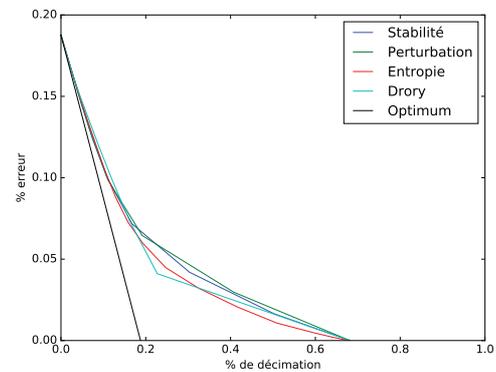
(a) Courbe Rappel/Précision Cones



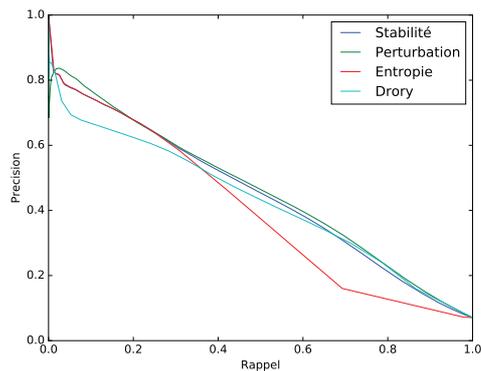
(b) Courbe Décimation/Erreur Cones



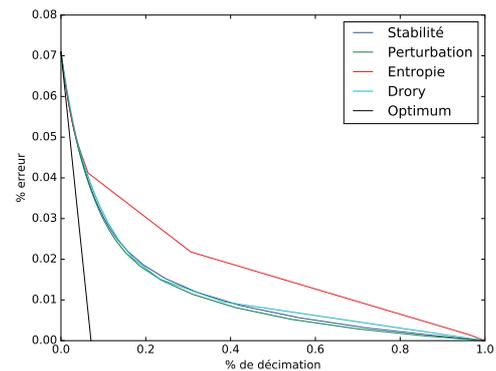
(c) Courbe Rappel/Précision Teddy



(d) Courbe Décimation/Erreur Teddy

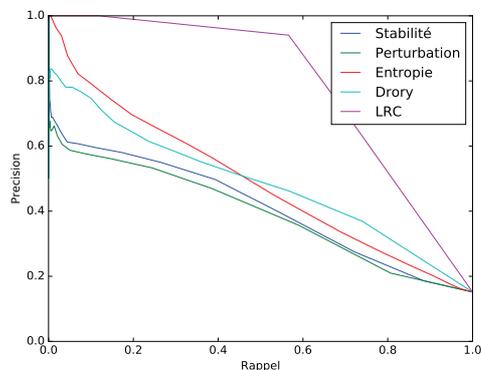


(e) Courbe Rappel/Précision KITTI

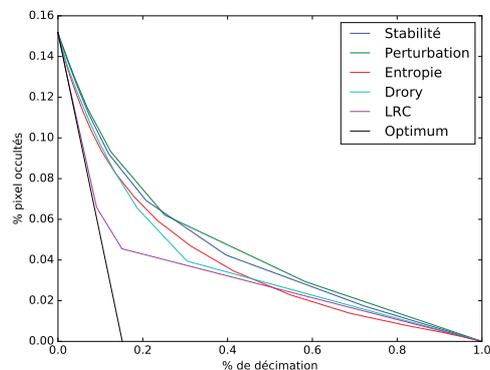


(f) Courbe Décimation/Erreur KITTI

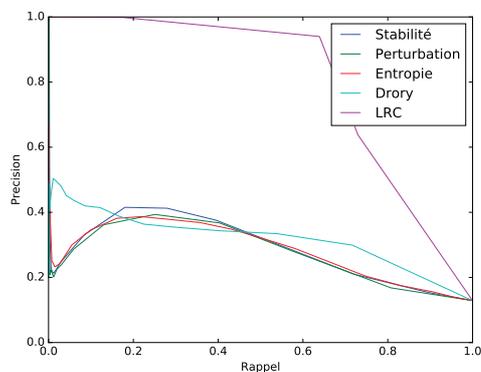
Figure 4.4 – Courbes représentant la capacité des indices à prédire les erreurs. Les reconstructions sont obtenues par la méthode SGM_4 . Nous montrons, sur la première ligne (a), (b), les courbes pour le couple Cones, sur la deuxième ligne (c), (d), les courbes pour le couple Teddy et sur la troisième ligne (e), (f), les courbes correspondantes aux 200 couples de la base KITTI15.



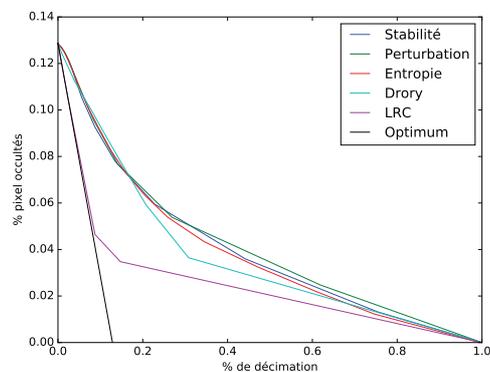
(a) Courbe Rappel-Précision Cones



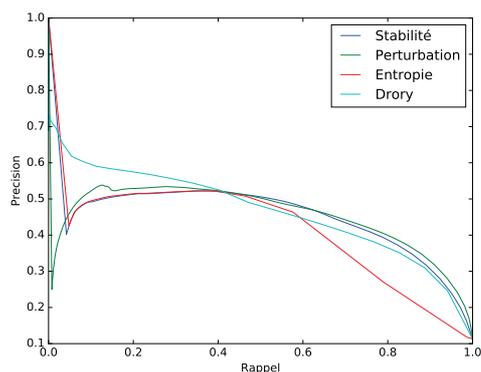
(b) Courbe Erreur-décimation Cones



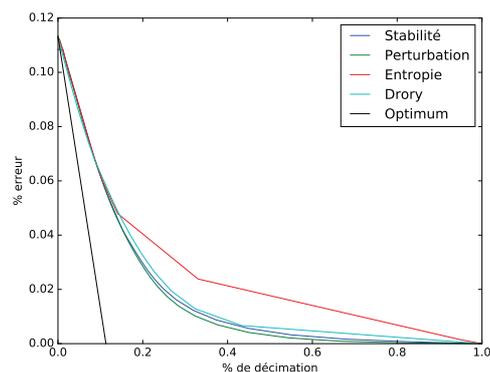
(c) Courbe Rappel-Précision Teddy



(d) Courbe Erreur-décimation Teddy



(e) Courbe Rappel-Précision KITTI



(f) Courbe Erreur-décimation KITTI

Figure 4.5 – Courbes représentant la capacité des indices à prédire les zones occultées. Nous montrons, sur la première ligne (a), (b), les courbes pour le couple Cones, sur la deuxième ligne (c), (d), les courbes pour le couple Teddy et sur la troisième ligne (e), (f), les courbes correspondantes aux 200 couples de la base KITTI15. Les reconstructions sont obtenues par la méthode SGM_4 .

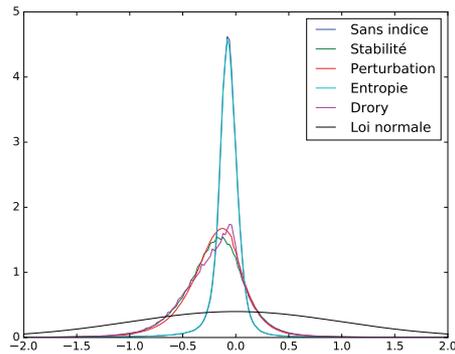


Figure 4.6 – Distribution de l’erreur divisé par un indice. Ce rapport est normalisé par son écart-type empirique.

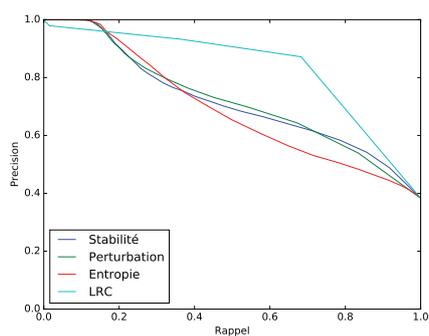
type inférieur à 1. La distribution des erreurs avant prise en compte de l’indice est très piquée et la décroissance de la queue est très rapide. Lorsque l’on utilise l’entropie, la distribution change peu. Cela vient du fait que les valeurs d’entropies varient relativement peu. Avec les indices de stabilité et de perturbations, la distribution s’étale mais la distribution n’est plus centrée. Avec l’indice de Drory, la distribution a son pic en zéro, mais n’est pas symétrique. Ces résultats semblent indiquer que trois derniers indices s’approchent d’un écart-type de la solution. Les moyennes quadratiques sont de 0.74, 0.63, 0.74 et 0.63 pour la stabilité, la perturbation, l’entropie et Drory, respectivement, ce qui seulement indiquer que l’indice de perturbation se rapproche le plus d’une variance.

3.2 Indices issus de la coupure de graphe

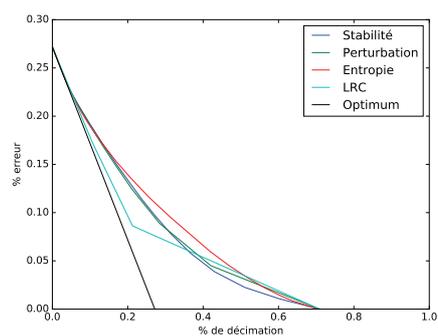
Dans cette section, nous évaluons les indices que nous proposons lorsque la méthode α -expansion est utilisée pour optimiser l’énergie. Nous donnons les résultats sur les deux couples Connes et Teddy de la base Middlebury. Les résultats sur la base KITTI ne sont pas présentés dans cette section car nous n’avons pas réussi à obtenir de reconstructions satisfaisantes avec α -expansion et les résultats obtenus sont trop dégradés pour pouvoir réellement évaluer les indices. En effet, la valeur de l’indice est dépendant du modèle, et le modèle, dans ce cas, n’a pas l’air adapté. La fonction d’attache aux données et la fonction de régularisation est la même que dans la section précédente. Dans la Figure 4.7, nous montrons les courbes de Rappel/Précision et Décimation/Erreur dans le cas où l’indice est utilisé pour prédire les erreurs et dans le cas où l’indice est utilisé pour prédire les zones occultées. Pour un rappel de 50%, la précision de détection des erreurs est de 65% pour l’entropie et de 73% pour la perturbation sur le couple Cones et de 55% pour l’indice de stabilité et de 63% pour la perturbation. Il apparaît que les résultats des trois indices sont relativement proches. Les performances pour la détection des occultations sont bien en deçà de celles de la validation croisée.

3.3 Indices pour le raffinement de la solution

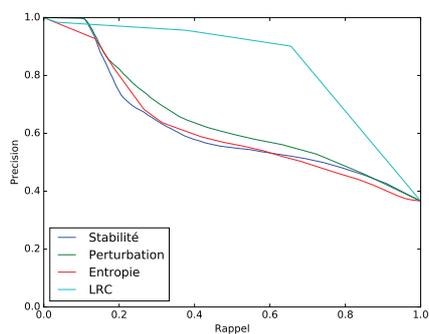
Dans cette section, nous montrons comment l’indice de stabilité peut être utilisé pour améliorer les reconstructions par une étape de raffinement. L’objectif est de valider indirectement le choix des indices en observant ou non une am Dans la Figure 4.8, nous montrons les résultats dans un cas où la géométrie de la scène est assez simple. Les valeurs élevées d’indice correspondent au ciel qui est proche de la saturation dans les images et aux contours d’occultation, par exemple autour du véhicule. Dans la Figures 4.9, nous pouvons observer que les valeurs élevées d’indices correspondent à des zones de l’image où la reconstruction est difficile. Il s’agit par exemple des objets proches et spéculaires comme les voitures et dans une moindre mesure la chaussée. Nous proposons d’utiliser l’indice de stabilité de deux manières. La première est un



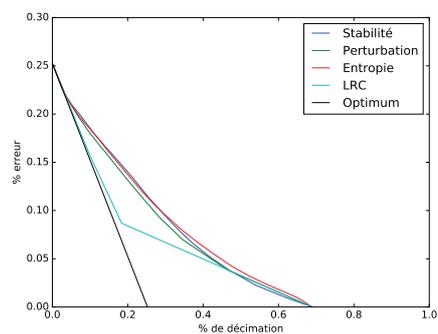
(a) Courbe Rappel-Précision Cones



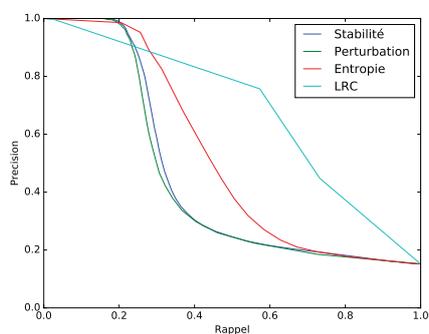
(b) Courbe Erreur-décimation Cones



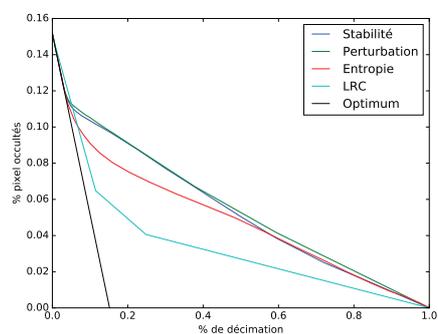
(c) Courbe Rappel-Précision Teddy



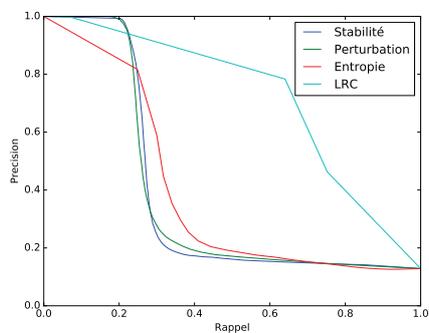
(d) Courbe Erreur-décimation Teddy



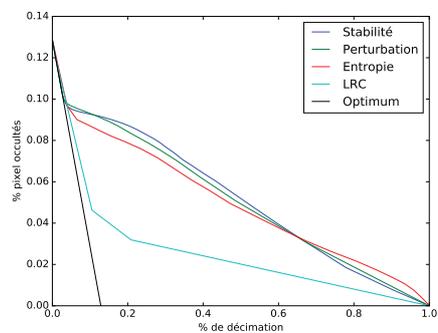
(e) Courbe Rappel-Précision Cones



(f) Courbe Erreur-décimation Cones



(g) Courbe Rappel-Précision Teddy



(h) Courbe Erreur-décimation Teddy

Figure 4.7 – Courbes représentant la capacité des indices à prédire les erreurs dans le couple Cones (a) et (b), et le couple Teddy (c) et (d) ainsi que les occultations dans le couple Cones (e) et (f) et dans le couple Teddy (g) et (h). Les reconstructions sont obtenues par la méthode α -expansion.

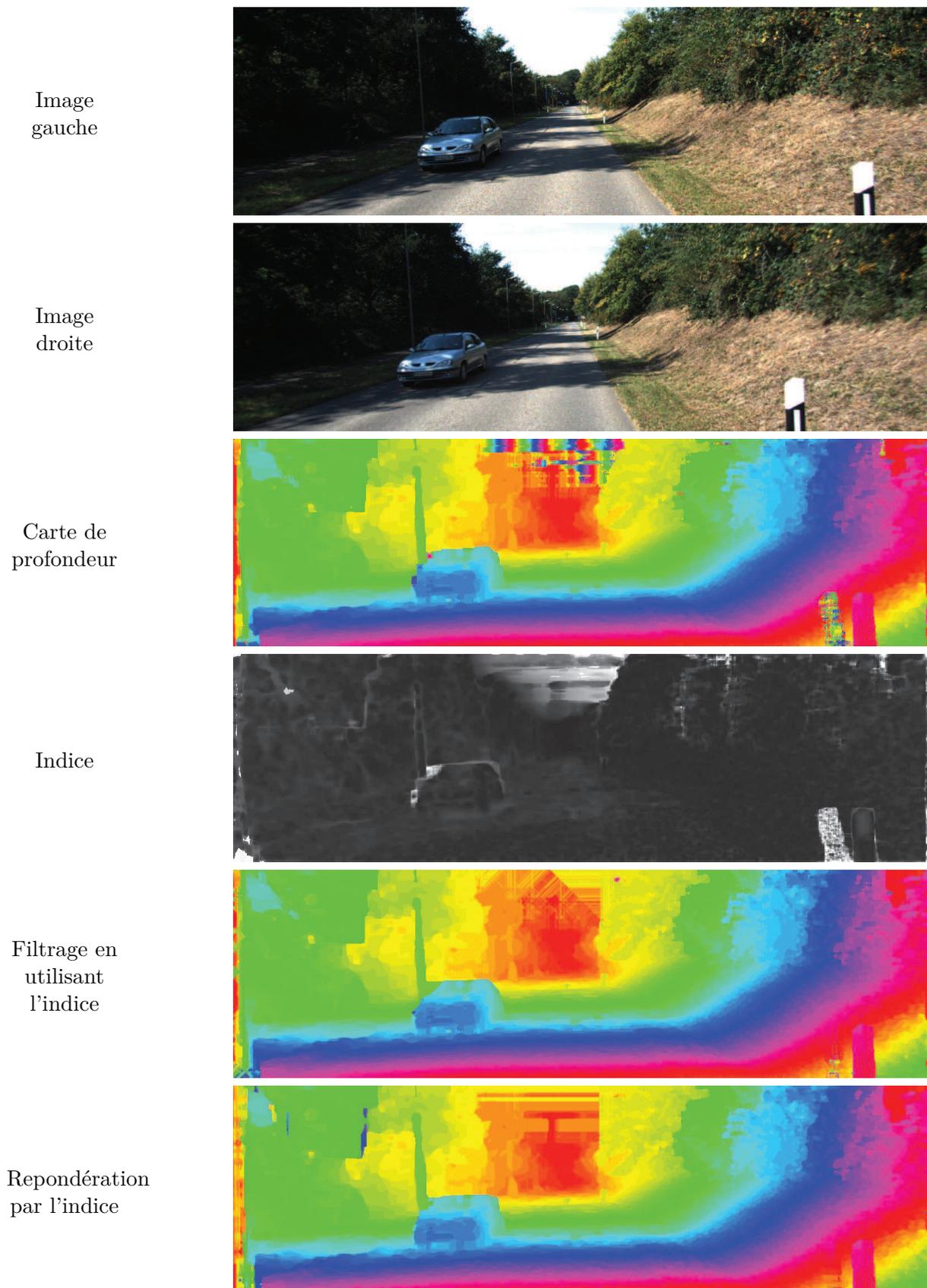


Figure 4.8 – De haut en bas, le couple 56 de la base KITTI15, carte de profondeur, l'indice de stabilité, la carte après filtrage utilisant l'indice, carte obtenue après repondération par l'indice.

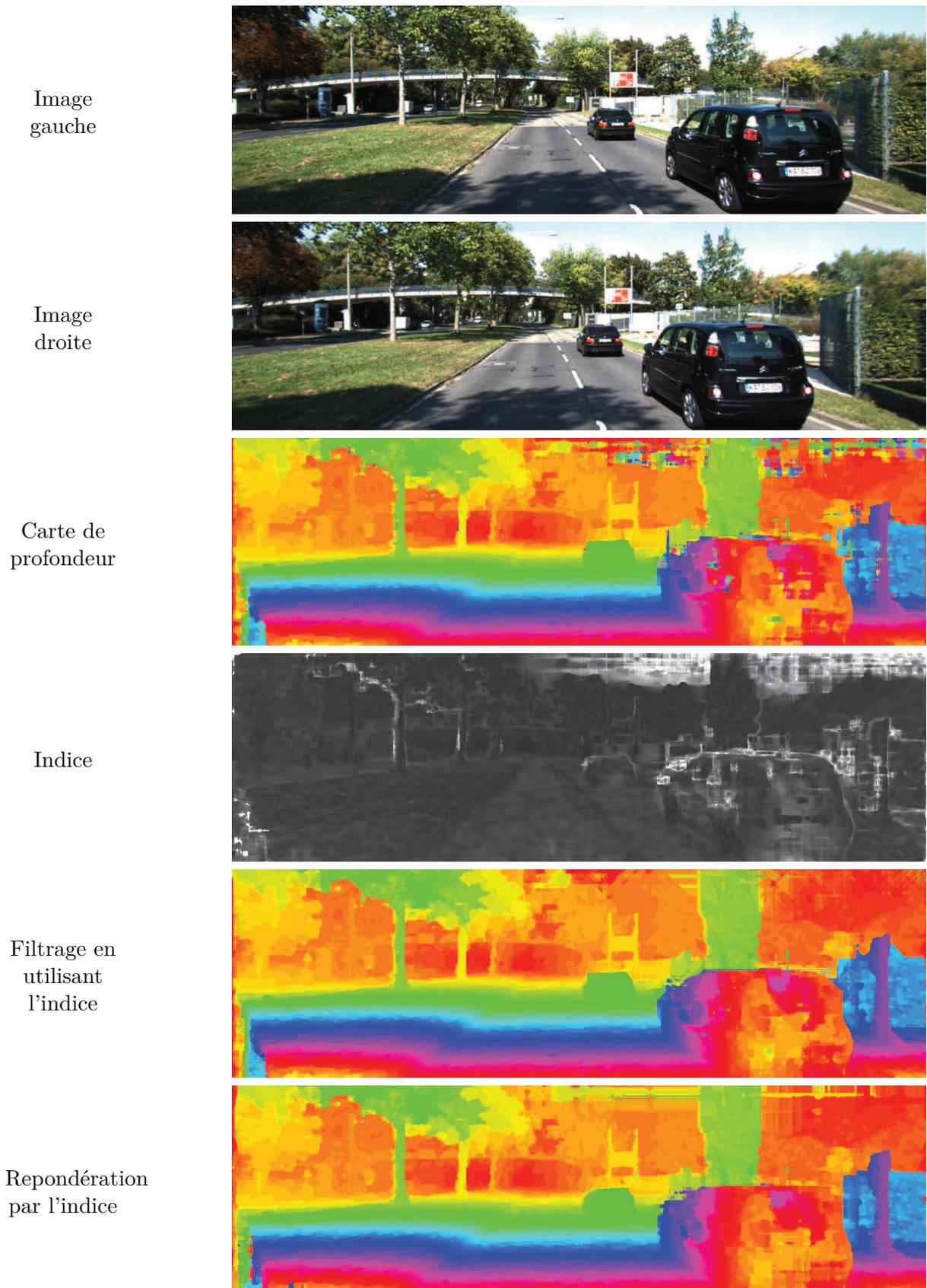


Figure 4.9 – De haut en bas, le couple 20 de la base KITTI15, carte de profondeur, l'indice de stabilité, la carte après filtrage utilisant l'indice, carte obtenue après repondération par l'indice

Filtrage LRC	KITTI12		KITTI15	
	Avant	Après	Avant	Après
Sans indice	7.60%	6.59%	6.90%	6.11%
Avec filtrage par l'indice	6.94%	6.40%	6.59%	6.14%
Avec repondération par l'indice	6.81%	6.01%	6.15%	5.64%

Tableau 4.2 – Pourcentages moyens d'erreur à plus de trois pixels de disparité de la vérité terrain pour les bases KITTI12 et KITTI15 : sur la première ligne, les résultats lorsque l'indice n'est pas pris en dans la reconstruction, sur la deuxième ligne, lorsque la solution est filtrée avec l'indice et sur la troisième ligne, après repondération par l'indice. Chaque résultat est donnée avant et après d'appliquer un filtrage LRC sur la solution.

post-traitement, elle consiste en un filtrage : les pixels qui ont une valeur d'indice supérieure à un seuil T_1 sont fixés à la valeur médiane des plus proches voisins dans huit directions qui ont une valeur d'indice est inférieure au seuil. La deuxième consiste à recalculer la reconstruction après avoir pondéré les attaches aux données par l'inverse de la valeur des indices :

$$D'_p(l_p) = \frac{\kappa}{Stab_p^{SGM}} D_p(l_p) \quad (4.61)$$

avec $D'_p(l_p)$ l'attache aux données repondérée et κ une constante pour maintenir l'équilibre dans l'énergie entre les termes d'attache aux données et les termes de régularisation.

Sur les base KITTI12 et KITTI15, nous appliquons le filtrage au dessus du seuil $T_1 = 2$, et nous prenons $\kappa = 1.5$. Sur la Figure 4.8, nous pouvons observer que la zone occultée par le poteau en bas à gauche de l'image est mieux reconstruite après filtrage. Sur la Figure 4.9, les disparités qui correspondent à la voiture la plus proche sont plus régulières après filtrage, sans toutefois arriver à la reconstruire correctement. Nous pouvons observer sur les Figures 4.8 et 4.9 que ces deux raffinements donnent des résultats proches. En termes quantitatifs, les pourcentages d'erreur à plus de trois disparités de la disparité vraie (vérité dite "non-occultée") sont donnés dans la Table 4.2. Le filtrage par l'indice permet de réduire le nombre de pixels faux de -0.46% et la pondération de -0.79% pour KITTI12. Sur la base KITTI15 la variation alors est de -0.31% et -0.75%. Nous comparons cette amélioration à celle obtenue après application du filtrage LRC qui est -1.01% sur KITTI12 et -0.79% sur KITTI15. Ces résultats sont meilleurs que par les raffinements utilisant l'indice, toutefois le filtrage LRC demande d'estimer la carte de disparité dans les deux points de vues. Nous avons alors appliqué le filtrage LRC sur les résultats après raffinement utilisant l'indice. La combinaison des deux traitements permettent d'obtenir une amélioration supplémentaire des résultats.

4 Conclusion

Nous avons expliqué, dans une première partie, la difficulté de caractériser de manière rigoureuse la stabilité d'une solution. Nous avons ensuite proposé des indices de stabilité pour deux méthodes d'optimisation discrète : la programmation dynamique et la coupure de graphe. Enfin les indices ont été évalués pour réaliser certaines tâches, telles que la détection d'erreur, la détection de zones occultées et l'estimation sommaire de la variance de la solution.

S'ils sont qualitativement intéressants, quantitativement, les résultats sont mitigés. Les indices sont meilleurs pour prédire les erreurs que pour prédire les zones occultées. On remarque que le comportement des différents indices est souvent proche, ce qui n'est pas forcément évident quand on compare la formulation de l'indice de stabilité et l'indice Drory par exemple. Cela peut venir du fait qu'ils sont tous fondés sur les caractéristiques de l'énergie et qu'ils captent approximativement le même type d'information.

Nous avons montré que l'utilisation de ces indices permet une amélioration des résultats sur la base KITTI15. Le filtrage des pixels où l'indice est supérieur à un seuil permet de réduire les

pourcentages d'erreur des reconstruction sur la base KITTI dans le cas de SGM. De même, la repondération de l'attache aux données par l'inverse de l'indice donne lieu à une amélioration des résultats. Cependant ces deux approches ont tendance à lisser la solution là où l'indice est élevé. Or les contours d'occultation donnent souvent des valeurs d'indice élevées. Il en résulte des bords moins bien définis dans les reconstructions après lissage.

Conclusions et perspectives

Les chaînes de traitement qui mettent en jeu des problèmes de traitement d'image sont généralement complexes, et donc découpés en sous-problèmes, de sorte qu'il est difficile de caractériser la stabilité en sortie de chaîne. La résolution de ces sous-problèmes est souvent formulée comme la minimisation d'une énergie. S'il est possible, à l'issue de l'optimisation, de caractériser l'incertitude de la solution obtenue, cette caractérisation peut être utilisée comme une incertitude sur l'entrée de l'étape de traitement suivante. En appliquant ce type d'approche à toute la chaîne, il est possible, en principe, de caractériser la sortie de la chaîne. Nous nous sommes intéressé au problème de dé-bruitage et plus en détails à celui de la reconstruction stéréoscopique. Nous avons choisi une énergie dont les hypothèses peuvent être interprétées dans le cadre bayésien. Les résultats sont obtenus à l'aide de deux méthodes d'optimisation discrète : la coupure de graphe et la programmation dynamique. Dans le Chapitre 1, nous avons présenté les méthodes par coupure de graphe. Dans le Chapitre 2, nous avons appliqué cette méthode à deux problèmes image : la reconstruction stéréoscopique et le dé-bruitage. Par la suite, dans le Chapitre 3, nous avons abordé le problème de la reconstruction à l'aide d'autres méthodes d'optimisation. Enfin, dans le Chapitre 4, nous avons proposé des indices de stabilité fondés sur les deux méthodes d'optimisation discrète.

1 Résultats et contributions

1.1 Extension de la classe de fonctions optimisées par coupure de graphe

Nous nous sommes intéressés à l'optimisation par coupure de graphe. Cette méthode d'optimisation discrète permet l'optimisation de problèmes combinatoires binaires. Un problème peut être traité par coupure de graphe s'il est formulé comme le minimum d'une fonction pseudo-booléenne sous-modulaire. Nous avons présenté un cas dans lequel la minimisation d'une énergie multi-labels est possible avec la méthode proposée dans [Ishikawa, 2003]. Toutefois, des méthodes d'optimisation approchées par fusions binaires sont souvent préférées car elles permettent de traiter une plus grande variété d'énergies. Nous nous sommes intéressés au cas de la régularisation par des fonctions de la différence de labels de pixel voisins qui sont convexes autour de zéro et concaves pour des grandes valeurs. Ce type de fonction peut être justifié grâce à l'interprétation bayésienne dans le cas du dé-bruitage et de la reconstruction stéréoscopique. Nous avons proposé un schéma d'optimisation qui alterne fusion par expansion et fusion par saut [Paget *et al.*, 2015]. Différentes variantes ont été testées : Alt-C, Alt-L et Alt-QL. Ce dernier schéma s'inspire d'un schéma proposé dans [Jeziarska *et al.*, 2011] dans un cas particulier et seulement appliqué au dé-bruitage. Nous avons appliqué ces schémas dans le cas du problème de dé-bruitage, ils permettent d'optimiser des énergies qui préservent les contours et qui reconstruisent correctement les dégradés d'intensité. Dans le cas de la reconstruction, ces schémas permettent d'utiliser la fonction de régularisation Potts lissée qui donne de meilleures reconstructions des plans non-frontaux que la fonction de régularisation Potts.

1.2 Qualité de l'approximation des schémas d'optimisation par fusion binaire

Nous avons comparé les solutions approchées par les schémas d'optimisation par fusions binaires à la solution du problème. Il apparaît, dans nos exemples de dé-bruitage, que lorsque l'énergie est convexe, les schémas faisant intervenir les fusions par expansion ou les fusions par saut atteignent le minimum global de l'énergie. Si l'attache aux données n'est pas convexe, sans être ambiguë, on observe un écart en énergie entre le minimum atteint et le minimum global. Toutefois, quand l'attache aux données est très ambiguë, cet écart augmente et les solutions approchées obtenues peuvent être très éloignées de la solution du problème. Le schéma α -Expansion est pourtant connu pour donner, dans la pratique, une solution proche de la solution minimale de l'énergie. Dans le cas de la reconstruction stéréoscopique, lorsque l'attache aux données est ambiguë et l'a priori n'est pas complètement valide, comme par exemple la chaussée qui est grise homogène et qui ne correspond pas à un plan frontal, α -Expansion a tendance à reconstruire des plans frontaux erronés.

La contrainte de sous-modularité se traduit par une contrainte sur la fonction de régularisation utilisable dans l'énergie. Ainsi, pour un type de fusion, nous pouvons définir le type de fonction de régularisation qui donne un problème sous-modulaire. Dans le cas où la fonction de régularisation ne respecte pas cette contrainte, une fusion binaire approchée doit être mise en œuvre. Nous avons utilisé la fusion relaxée, la méthode QPBO [Boros et Hammer, 2002] et une variante QPBO-P [Rother *et al.*, 2007]. Nous montrons que l'approximation faite lors de la résolution des fusions binaires impacte moins le résultat que le choix d'un schéma d'optimisation adapté au problème.

1.3 Interprétation de SGM par rapport à un champ de Markov formulé sur toute l'image

Dans [Hirschmüller, 2008], la méthode SGM est présentée comme une résolution approchée d'un champ de Markov formulé sur toute l'image, comme c'est le cas dans les méthodes par coupure de graphe. Dans cette logique, les variantes CAT et MGM présentées dans [Ha *et al.*, 2015] et dans [Facciolo *et al.*, 2015] définissent comme objectif de réduire l'écart entre les deux formulations. Plutôt qu'une approximation, nous préférons interpréter la méthode SGM comme une nouvelle formulation de l'a priori sur la solution. Dans le cas de SGM dans quatre directions, l'a priori est que le label pris par le pixel doit être cohérent avec la ligne et la colonne de l'image où il se trouve. Cette formulation permet de privilégier des directions particulières, ce qui n'est pas possible dans le cas d'un champ de Markov sur toute l'image.

1.4 Modification des directions privilégiées dans la méthode SGM

Nous avons étudié deux variantes de la méthode SGM, CAT [Ha *et al.*, 2015] et MGM [Facciolo *et al.*, 2015]. Nous avons donné une interprétation de l'a priori sur la solution de ces deux méthodes et remarqué qu'elles ont tendance à diminuer l'importance des chemins droits par rapport à ceux qui "zigzaguent". Nous avons observé que lorsque les coûts d'attache aux données ont été lissés localement, ces deux méthodes n'apportent pas beaucoup d'améliorations. Cela semble indiquer que ces méthodes permettent de lisser localement la solution. Nous avons proposé deux variantes de ces méthodes qui modifient la direction favorisée. Ces variantes ont donné des améliorations sur la base Middlebury, mais pas ou peu sur la base KITTI.

1.5 Indice de confiance fondé sur SGM

Lors des calculs réalisés par la méthode SGM, des coûts sont associés à chaque valeur que peut prendre le label d'un pixel. Ces coûts peuvent être utilisés pour calculer un indice par pixel. Nous avons proposé un indice qui mesure la stabilité de la solution [Paget *et al.*, 2017].

Dans le cas d'un modèle de dé-bruitage avec bruit gaussien, nous avons explicité la valeur de l'indice en fonction de la variance du signal et de l'a priori. Nous avons comparé cet indice à d'autres indices fondés sur l'énergie proposés dans [Haeusler *et al.*, 2013] et [Drory *et al.*, 2014]. L'indice de stabilité donne des résultats intéressants pour prédire les erreurs dans la solution, mais sa capacité à prédire les zones occultées est limitée. Nous avons montré que les résultats obtenus avec les différents indices sont très proches ce qui indique que ces indices captent la même information. En revanche, nous avons des difficultés à évaluer la capacité des indices à estimer la variance de la solution. Nous avons montré que l'indice peut être utilisé pour raffiner la solution obtenue de deux manières : par un filtrage de la solution et par un nouveau calcul avec une pondération des attaches aux données par l'indice.

1.6 Indice de confiance fondé sur α -expansion

Nous nous sommes intéressés à un indice de la stabilité fondé sur la méthode d'optimisation la coupure de graphe. Cet indice mesure la stabilité de l'affectation binaire d'un nœud à l'issue du poussage de flot. Cette mesure, proposée par [Kohli et Torr, 2008] a été appliquée dans le cas de la méthode d'Ishikawa [Ishikawa, 2003], sans présenter d'applications pratiques. Cette méthode étant peu utilisée pour des raisons pratiques, nous avons proposé une mesure dans le cas de la méthode d'optimisation approchée α -Expansion. Nous avons proposé des algorithmes approchés qui permettent de calculer rapidement une borne inférieure de cet indice. Nous avons appliqué cet indice dans le cas de la reconstruction stéréoscopique. Qualitativement, cet indice prend des valeurs élevées là où la reconstruction s'est mal passée. Les résultats quantitatifs sont eux plus mitigés : l'indice prédit de manière imparfaite la présence d'erreurs dans la solution et peine à prédire les zones occultées dans la solution.

2 Perspectives

2.1 Formulation de la méthode SGM dans le cadre Bayésien

Dans notre analyse de la méthode SGM nous avons conclu que cette méthode est plus une nouvelle formulation de l'a priori qu'une approximation de la solution d'un champ de Markov sur toute l'image. Nous pensons qu'une interprétation plus poussée de cette méthode dans le cadre bayésien peut permettre de proposer des modèles originaux adaptés au problème de la reconstruction stéréoscopique. Ce type de réflexion pourrait permettre de faire le lien avec d'autres méthodes d'optimisation, et notamment permettre de privilégier des directions particulières dans le cas des méthodes par coupure de graphe.

2.2 Impact de la résolution de l'image sur la méthode SGM

Lors de nos expérimentations, nous avons remarqué que la méthode SGM était très sensible à la résolution de l'image. En effet, comme la méthode SGM ne traite que des alignements, et si l'objet est très résolu, alors l'information utilisée dans SGM représente une très faible partie de l'image. Ainsi, augmenter la résolution de l'image peut amener à une détérioration des résultats. Cet effet peut être compensé par un lissage local des coûts, mais ce type d'approche est assez coûteux en calculs. Certaines approches avec la méthode SGM sont multi-échelles comme dans [Hermann et Klette, 2012] et tire parti de ces effets. Cela pose la question de l'existence d'une échelle pour mettre en œuvre la méthode SGM.

2.3 Appliquer l'indice dans une chaîne de traitements

Nous avons montré que la prise en compte de l'indice en sortie d'un processus pouvait servir à améliorer le résultat de ce processus. Toutefois, la reconstruction stéréoscopique s'inscrit le plus souvent au sein d'une chaîne de traitement, de sorte que la solution de reconstruction est vue comme des données dans l'étape de traitement suivante. L'indice sur la reconstruction peut alors être utilisé comme une information d'incertitude. Ainsi, il est possible de propager les valeurs d'indice tout au long de la chaîne de traitement. Il paraît donc naturel d'utiliser l'indice sur la sortie de la reconstruction lors du recalage des reconstructions entre elles ou lors de la fusion des nuages de points.

2.4 Utilisation des indices dans un cas maîtrisé

La validation des indices sur des images naturelles est une tâche difficile. D'une part, les sources de perturbation peuvent être nombreuses et mal connues, d'autre part le contenu de ces images peut être complexe. Il paraît intéressant de se placer dans un cas où tous les éléments du problème sont connus. Nous avons explicité la valeur de l'indice de stabilité dans le cas du dé-bruitage avec un bruit gaussien. Il serait utile d'aller plus loin dans ce type d'évaluations.

De plus, nous sommes limités dans les traitements simples utilisant à l'indice. En supposant que l'indice détecte les zones où le modèle n'est pas cohérent avec les données, il faut être en mesure de proposer un modèle alternatif. Dans le cas de la reconstruction, il s'agit soit de porter les points à l'arrière plan, soit de lisser la scène. Ainsi, il peut être intéressant de traiter un cas où une partie des données suit un premier modèle et l'autre partie suit un deuxième modèle. En appliquant le deuxième modèle, là où le premier a donné de fortes valeurs d'indice, la solution doit en principe être améliorée. Un tel problème peut être formulé dans le cadre d'un problème avec des données multi-sources.

2.5 Utilisation des modèles obtenus par apprentissage

Dans ce travail, nous avons cherché à maintenir un modèle simple et lisible. Nous avons donc fait le choix d'utiliser des modèles fondés sur une interprétation probabiliste en utilisant l'approche bayésienne. Toutefois, devant la difficulté du problème, il peut être intéressant de travailler avec des modèles moins bien justifiés mais qui donnent de très bons résultats. De fait nous avons écarté les modèles par apprentissage car il est difficile d'acquérir une compréhension du phénomène modélisé avec ces méthodes. Toutefois, ce sont ces méthodes qui donnent les meilleurs modèles sur les bases d'évaluation Middlebury et KITTI. Les indices que nous proposons sont fondés sur l'énergie dérivée du modèle. Ainsi, les résultats obtenus par l'indice sont très dépendants de la validité du modèle : plus le modèle est proche du phénomène, meilleure sera l'évaluation de l'indice. Il serait donc intéressant d'évaluer les indices lorsque l'attache aux données est apprise.

Bibliographie

- ARNDT, O. J., BECKER, D., BANZ, C. et BLUME, H. (2013). Parallel implementation of real-time semi-global matching on embedded multi-core architectures. *In Embedded Computer Systems : Architectures, Modeling, and Simulation (SAMOS XIII), 2013 International Conference on*, pages 56–63. IEEE.
- BANNO, A. et IKEUCHI, K. (2011). Disparity map refinement and 3d surface smoothing via directed anisotropic diffusion. *Computer Vision and Image Understanding*, 115(5):611–619.
- BELLMAN, R. (1952). On the theory of dynamic programming. *Proceedings of the National Academy of Sciences*, 38(8):716–719.
- BELLMAN, R. et DREYFUS, S. (1958). An application of dynamic programming to the determination of optimal satellite trajectories. Rapport technique, RAND CORP SANTA MONICA CA.
- BILLIONNET, A. et JAUMARD, B. (1989). A decomposition method for minimizing quadratic pseudo-boolean functions. *Operations Research Letters*, 8(3):161–163.
- BLEYER, M., ROTHER, C., KOHLI, P., SCHARSTEIN, D. et SINHA, S. (2011). Object stereo—joint stereo matching and object segmentation. *In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3081–3088. IEEE.
- BOROS, E. et GRUBER, A. (2014). On quadratization of pseudo-boolean functions. *arXiv preprint arXiv :1404.6538*.
- BOROS, E. et HAMMER, P. L. (2002). Pseudo-boolean optimization. *Discrete applied mathematics*, 123(1):155–225.
- BOROS, E., HAMMER, P. L., SUN, R. et TAVARES, G. (2008). A max-flow approach to improved lower bounds for quadratic unconstrained binary optimization (qubo). *Discrete Optimization*, 5(2):501–529.
- BOYKOV, Y. et KOLMOGOROV, V. (2004). An experimental comparison of min-cut-max-flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137.
- BOYKOV, Y. Y. et JOLLY, M.-P. (2001). Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. *In Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 105–112. IEEE.
- BREDIES, K., KUNISCH, K. et POCK, T. (2010). Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526.
- BUADES, A., COLL, B. et MOREL, J.-M. (2005). A non-local algorithm for image denoising. *In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 60–65. IEEE.
- CANNY, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698.
- CAO, Y., WU, Z. et SHEN, C. (2017). Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*.
- CARAFFA, L., TAREL, J.-P. et CHARBONNIER, P. (2015). The guided bilateral filter : When the joint/cross bilateral filter becomes robust. *IEEE Transactions on Image Processing*, 24(4): 1199–1208.

- CHAMBON, S. et CROUZIL, A. (2011). Similarity measures for image matching despite occlusions in stereo vision. *Pattern Recognition*, 44(9):2063–2075.
- DELAGE, E., LEE, H. et NG, A. Y. (2006). A dynamic bayesian network model for autonomous 3d reconstruction from a single indoor image. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2418–2428. IEEE.
- DRORY, A., HAUBOLD, C., AVIDAN, S. et HAMPRECHT, F. A. (2014). Semi-global matching : a principled derivation in terms of message passing. In *German Conference on Pattern Recognition*, pages 43–53. Springer.
- ELIAS, P., FEINSTEIN, A. et SHANNON, C. (1956). A note on the maximum flow through a network. *IRE Transactions on Information Theory*, 2(4):117–119.
- FACCIOLO, G., FRANCHIS, C. d. et MEINHARDT, E. (2015). MGM : A Significantly More Global Matching for Stereovision. In PRESS, B., éditeur : *BMVC 2015*, Swansea, United Kingdom.
- FAUGERAS, O. D. et LUSTMAN, F. (1988). Motion and structure from motion in a piecewise planar environment. *International Journal of Pattern Recognition and Artificial Intelligence*, 2(03):485–508.
- FIX, A., GRUBER, A., BOROS, E. et ZABIH, R. (2011). A graph cut algorithm for higher-order markov random fields. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1020–1027. IEEE.
- FORD, L. R. et FULKERSON, D. (1962). *Flow in networks*. Princeton University Press.
- FREEDMAN, D. et DRINEAS, P. (2005). Energy minimization via graph cuts : Settling what is possible. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 939–946. IEEE.
- FUSIELLO, A., ROBERTO, V. et TRUCCO, E. (1997). Efficient stereo with multiple windowing. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 858–863. IEEE.
- GALLAGHER, A. C., BATRA, D. et PARIKH, D. (2011). Inference for order reduction in markov random fields. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1857–1864. IEEE.
- GEMAN, S. et GEMAN, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6): 721–741.
- GOLDBERG, A. V. et TARJAN, R. E. (1988). A new approach to the maximum-flow problem. *Journal of the ACM (JACM)*, 35(4):921–940.
- GONG, M. et YANG, Y.-H. (2007). Real-time stereo matching using orthogonal reliability-based dynamic programming. *IEEE Transactions on Image Processing*, 16(3):879–884.
- HA, J. M., JEON, B., JEON, J., JO, S. Y. et JEONG, H. (2015). Cost aggregation table : a theoretic derivation on the markov random field and its relation to message passing. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 2224–2228. IEEE.
- HAEUSLER, R., NAIR, R. et KONDERMANN, D. (2013). Ensemble learning for confidence measures in stereo vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 305–312.
- HARRIS, C. et STEPHENS, M. (1988). A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Manchester, UK.

- HARRIS, T. et ROSS, F. (1955). Fundamentals of a method for evaluating rail net capacities. Rapport technique, RAND CORP SANTA MONICA CA.
- HERMANN, S. et KLETTE, R. (2012). Iterative semi-global matching for robust driver assistance systems. *In Asian Conference on Computer Vision*, pages 465–478. Springer.
- HIRSCHMÜLLER, H. (2008). Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2):328–341.
- HIRSCHMÜLLER, H. et SCHARSTEIN, D. (2007). Evaluation of cost functions for stereo matching. *In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE.
- HIRSCHMULLER, H. et SCHARSTEIN, D. (2009). Evaluation of stereo matching costs on images with radiometric differences. *IEEE transactions on pattern analysis and machine intelligence*, 31(9):1582–1599.
- HU, X. et MORDOHAJ, P. (2012). A quantitative evaluation of confidence measures for stereo vision. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2121–2133.
- ISHIKAWA, H. (2003). Exact optimization for markov random fields with convex priors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(10):1333–1336.
- ISHIKAWA, H. (2011). Transformation of general binary mrf minimization to the first-order case. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(6):1234–1249.
- ISHIKAWA, H. (2014). Higher-order clique reduction without auxiliary variables. *In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1362–1369. IEEE.
- JEZIERSKA, A., TALBOT, H., VEKSLER, O. et WESIERSKI, D. (2011). A fast solver for truncated-convex priors : quantized-convex split moves. *In Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 45–58. Springer.
- KAHL, F. et STRANDMARK, P. (2012). Generalized roof duality. *Discrete applied mathematics*, 160(16):2419–2434.
- KENDALL, A., MARTIROSYAN, H., DASGUPTA, S., HENRY, P., KENNEDY, R., BACHRACH, A. et BRY, A. (2017). End-to-end learning of geometry and context for deep stereo regression. *arXiv preprint arXiv :1703.04309*.
- KIM, S., MIN, D., HAM, B., KIM, S. et SOHN, K. (2017). Deep stereo confidence prediction for depth estimation. *In ICIP'17, IEEE International Conference on Image Processing*, page 992p.
- KLAUS, A., SORMANN, M. et KARNER, K. (2006). Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. *In Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 3, pages 15–18. IEEE.
- KOHLI, P. et TORR, P. H. (2008). Measuring uncertainty in graph cut solutions. *Computer Vision and Image Understanding*, 112(1):30–38.
- KOLMOGOROV, V. (2006). Convergent tree-reweighted message passing for energy minimization. *IEEE transactions on pattern analysis and machine intelligence*, 28(10):1568–1583.
- KOLMOGOROV, V. (2010). Generalized roof duality and bisubmodular functions. *In Advances in neural information processing systems*, pages 1144–1152.
- KOLMOGOROV, V., MONASSE, P. et TAN, P. (2014). Kolmogorov and Zabih’s Graph Cuts Stereo Matching Algorithm. *Image Processing On Line*, 4:220–251.

- KOLMOGOROV, V. et ZABIH, R. (2004). What energy functions can be minimized via graph cuts? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2):147–159.
- LEE, J., JUNG, H., KIM, Y. et SOHN, K. (2017). Automatic 2d-to-3d conversion using multi-scale deep neural network. *In ICIP'17, IEEE International Conference on Image Processing*, page 730p.
- LEMPITSKY, V., ROTHER, C., ROTH, S. et BLAKE, A. (2010). Fusion moves for markov random field optimization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(8): 1392–1405.
- LOWE, D. G. (1999). Object recognition from local scale-invariant features. *In Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee.
- MAYER, N., ILG, E., HAUSSER, P., FISCHER, P., CREMERS, D., DOSOVITSKIY, A. et BROX, T. (2016). A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4048.
- MEI, X., SUN, X., ZHOU, M., JIAO, S., WANG, H. et ZHANG, X. (2011). On building an accurate stereo matching system on graphics hardware. *In Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 467–474. IEEE.
- MENZE, M. et GEIGER, A. (2015). Object scene flow for autonomous vehicles. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3061–3070.
- MERRELL, P., AKBARZADEH, A., WANG, L., MORDOHAI, P., FRAHM, J.-M., YANG, R., NISTÉR, D. et POLLEFEYS, M. (2007). Real-time visibility-based fusion of depth maps. *In Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE.
- PAAR, G., SIDLA, O. et PÖLZLEITNER, W. (1998). Genetic feature selection for highly-accurate stereo reconstruction of natural surfaces. *Intelligent Systems and Advanced Manufacturing : Intelligent Robots and Computer Vision XVII : Algorithms, Techniques, and Active Vision, volume SPIE-3522*, pages 455–466.
- PAGET, M., TAREL, J.-P. et CARAFFA, L. (2015). Extending α -expansion to a larger set of regularization functions. *In Proceedings of IEEE International Conference on Image Processing (ICIP'15)*, pages 1051–1055, Québec City, Canada. <http://perso.lpc.fr/tarel.jean-philippe/publis/icip15.html>.
- PAGET, M., TAREL, J.-P. et MONASSE, P. (2017). Stereo ambiguity index for semi-global matching. *In Proceedings of IEEE International Conference on Image Processing (ICIP'17)*, pages 2513–2517, Beijing, China. <http://perso.lpc.fr/tarel.jean-philippe/publis/icip17.html>.
- PAPARODITIS, N., PAPELARD, J.-P., CANNELLE, B., DEVAUX, A., SOHEILIAN, B., DAVID, N. et HOUZAY, E. (2012). Stereopolis ii : A multi-purpose and multi-sensor 3d mobile mapping system for street visualisation and 3d metrology. *Revue française de photogrammétrie et de télédétection*, 200(1):69–79.
- PARK, M.-G. et YOON, K.-J. (2015). Leveraging stereo matching with learning-based confidence measures. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 101–109.
- POGGI, M. et MATTOCCIA, S. (2017). Learning to predict stereo reliability enforcing local consistency of confidence maps. *In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2.

- RANFT, B. et STRAUSS, T. (2014). Modeling arbitrarily oriented slanted planes for efficient stereo vision based on block matching. *In Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, pages 1941–1947. IEEE.
- ROSENBERG, I. (1975). Reduction of bivalent maximization to the quadratic case. *Cahiers du Centre d'Etudes de Recherche Operationnelle*, 17:71–74.
- ROTHER, C., KOLMOGOROV, V., LEMPITSKY, V. et SZUMMER, M. (2007). Optimizing binary mrfs via extended roof duality. *In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE.
- SAXENA, A., CHUNG, S. H. et NG, A. Y. (2006). Learning depth from single monocular images. *In Advances in neural information processing systems*, pages 1161–1168.
- SCHARSTEIN, D., HIRSCHMÜLLER, H., KITAJIMA, Y., KRATHWOHL, G., NEŠIĆ, N., WANG, X. et WESTLING, P. (2014a). High-resolution stereo datasets with subpixel-accurate ground truth. *In German Conference on Pattern Recognition*, pages 31–42. Springer.
- SCHARSTEIN, D., HIRSCHMÜLLER, H., KITAJIMA, Y., KRATHWOHL, G., NEŠIĆ, N., WANG, X. et WESTLING, P. (2014b). High-resolution stereo datasets with subpixel-accurate ground truth. *In German Conference on Pattern Recognition*, pages 31–42. Springer.
- SCHARSTEIN, D. et SZELISKI, R. (2003). High-accuracy stereo depth maps using structured light. *In Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE.
- SCHRIJVER, A. (2002). On the history of the transportation and maximum flow problems. *Mathematical Programming*, 91(3):437–445.
- SEITZ, S. M., CURLESS, B., DIEBEL, J., SCHARSTEIN, D. et SZELISKI, R. (2006). A comparison and evaluation of multi-view stereo reconstruction algorithms. *In Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 519–528. IEEE.
- SEKI, A. et POLLEFEYS, M. (2016). Patch based confidence prediction for dense disparity map. *In British Machine Vision Conference (BMVC)*, volume 10.
- SPYROPOULOS, A., KOMODAKIS, N. et MORDOHAJ, P. (2014). Learning to detect ground control points for improving the accuracy of stereo matching. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1621–1628.
- STURM, P. (2011). A historical survey of geometric computer vision. *In Computer Analysis of Images and Patterns*, pages 1–8. Springer.
- SUN, J., LI, Y., KANG, S. B. et SHUM, H.-Y. (2005). Symmetric stereo matching for occlusion handling. *In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 399–406. IEEE.
- SUN, X., MEI, X., JIAO, S., ZHOU, M. et WANG, H. (2011). Stereo matching with reliable disparity propagation. *In 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2011 International Conference on*, pages 132–139. IEEE.
- TAO, H., SAWHNEY, H. S. et KUMAR, R. (2001). A global matching framework for stereo computation. *In Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 532–539. IEEE.
- TOLSTOI, A. (1930). Methods of finding the minimal total kilometrage in cargo transportation planning in space. *TransPress of the National Commissariat of Transportation*, pages 23–55.
- TOMASI, C. et MANDUCHI, R. (1998). Bilateral filtering for gray and color images. *In Computer Vision, 1998. Sixth International Conference on*, pages 839–846. IEEE.

- VEKSLER, O. (1999). *Efficient graph-based energy minimization methods in computer vision*. Thèse de doctorat, Cornell University.
- VEKSLER, O. (2003). Fast variable window for stereo correspondence using integral images. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE.
- VINCENT, P., LAROCHELLE, H., LAJOIE, I., BENGIO, Y. et MANZAGOL, P.-A. (2010). Stacked denoising autoencoders : Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(Dec):3371–3408.
- VOGEL, C., SCHINDLER, K. et ROTH, S. (2015). 3d scene flow estimation with a piecewise rigid scene model. *International Journal of Computer Vision*, 115(1):1–28.
- WEI, Y. et QUAN, L. (2005). Asymmetrical occlusion handling using graph cut for multi-view stereo. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 902–909. IEEE.
- WOODFORD, O., TORR, P., REID, I. et FITZGIBBON, A. (2009). Global stereo reconstruction under second-order smoothness priors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(12):2115–2128.
- XU, L. et JIA, J. (2008). Stereo matching : An outlier confidence approach. *Computer Vision–ECCV 2008*, pages 775–787.
- YEDIDIA, J. S., FREEMAN, W. T. et WEISS, Y. (2001). Generalized belief propagation. In *Advances in neural information processing systems*, pages 689–695.
- YOON, K.-J. et KWEON, I. S. (2006). Adaptive support-weight approach for correspondence search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):650–656.
- ZABIH, R. et WOODFILL, J. (1994). Non-parametric local transforms for computing visual correspondence. In *European conference on computer vision*, pages 151–158. Springer.
- ZHANG, K., LU, J. et LAFRUIT, G. (2009). Cross-based local stereo matching using orthogonal integral images. *IEEE transactions on circuits and systems for video technology*, 19(7):1073–1079.

Contributions

Conférences :

- Entending alpha-expansion to a larger set of regularisation functions
 - **M. Paget**, J-P. Tarel et L. Caraffa,
 - IEEE International Conference on Image Processing (ICIP'15),
 - Septembre 2015. **Poster**
- Stereo Ambiguity Index for Semi-Global Matching
 - **M. Paget**, J-P. Tarel et P. Monasse,
 - IEEE International Conference on Image Processing (ICIP'17),
 - Septembre 2017. **Poster**.

Chapitre de livre :

- Optimisation de fonctions pseudo-bouliennes quadratiques.
 - L. Caraffa, J.-P. Tarel et **M. Paget**,
 - Chapitre du livre "Optimisation de formes"
 - Etudes et recherches des laboratoires des ponts et chaussées (ERLPC), série "sciences pour le génie civil".
 - Statut : sous presse.

Résumé

Les tâches réalisées en traitement d'image tendent à devenir de plus en plus complexes. Par exemple, dans le contexte routier, les systèmes d'aide à la conduite, (*Advanced driver-assistance systems*), visent à une automatisation complète de la tâche de conduite. L'évaluation de la fiabilité représente un enjeu important pour ce type d'application. Face à la difficulté des tâches à réaliser, les chaînes de traitements sont souvent divisées en de nombreuses étapes de calculs de sorte qu'il est difficile de caractériser les sorties de la chaîne en fonction des perturbations des entrées. Les étapes du traitement consistent le plus souvent en des problèmes formulés comme la minimisation d'une énergie. Cette énergie est généralement difficile à optimiser, ce qui nécessite la mise en œuvre de méthodes d'optimisation adaptées. Dans cette thèse, nous cherchons à caractériser la solution d'un traitement à partir des calculs réalisés au cours de l'étape d'optimisation. Cette approche nous a permis de proposer des indices de stabilité de la solution dans le cadre de deux méthodes d'optimisation discrètes : la coupure de graphe et la programmation dynamique. Tout d'abord, nous nous sommes intéressés au problème au dé-bruitage et au problème de la reconstruction stéréoscopique en contexte routier, dans le cadre de l'optimisation par coupure de graphe. Les modèles issus de l'interprétation bayésienne amènent à optimiser des énergies qui ne peuvent pas être traitées avec les schémas classiques d'optimisation par fusion binaire. Nous avons proposé un schéma adapté qui met en jeu des fusions binaires par expansion et par saut. L'application de ce schéma aux problèmes de la reconstruction stéréoscopique et au dé-bruitage, nous a permis d'obtenir des solutions possédant les caractéristiques que nous recherchions : des contours d'objets nets et des dégradés progressifs dans les zones homogènes. Ensuite, dans le contexte de la programmation dynamique, nous avons réinterprété l'a priori mis en jeu dans la méthode de reconstruction *Semi-Global Matching* ainsi que certaines de ses variantes. Nous avons proposé d'ajouter un paramètre à ces méthodes afin de modifier les directions privilégiées dans l'a priori. Enfin, nous avons proposé des indices de stabilité de la solution dans le cadre de la coupure de graphe et de la programmation dynamique. La prise en compte de ces indices, dans une étape de raffinement des solutions, permet une amélioration des résultats.

Abstract

Problems solved with image processing tend to be more and more complex. For instance, in the road traffic context, ADAS (Advanced driver-assistance systems) aim a completely automatic driving task. Evaluating system reliability is an important challenge in that case. These applications being hard to perform, processing chains are often divided into numerous computing steps. As a consequence, characterizing the output of a chain using input uncertainty is not obvious. Most of the time, each image processing step is formulated as an energy minimization. As energies are often difficult to minimize, suitable optimization methods are needed. In this thesis, our goal is to characterize the solution of a processing step using the computations performed during the optimization. With this approach, we propose stability indices when two discrete optimization methods are used : graph-cut and dynamic programming. First, we focus on the denoising problem and on the stereoscopic reconstruction problem in the road context using graph-cut. Models obtained by the Bayesian interpretation lead to optimize energies which cannot be handled by classical binary fusion optimization schemes. We propose an appropriate scheme involving binary fusion by expansion and by jump. When applied to stereoscopic reconstruction and denoising, such schemes produce solutions which have the wanted characteristics : sharp edges at contours and smooth shades in homogeneous areas. Then, in dynamic programming context, we reinterpret the prior used in the Semi-Global Matching (SGM) stereoscopic reconstruction method and in some of its variants. We propose an additional parameter in order to alter the favored direction in the prior. At last, we propose stability index of the solution in graph-cut and dynamic programming context. We show improvements in the results by using these indices in a refinement step.