

Recalage direct d'images sans région d'intérêt

21 mai 2010

F. Brunet^{1,2} A. Bartoli¹ R. Malgouyres¹ N. Navab²
¹Université d'Auvergne ²CAMPAR
Clermont-Ferrand Technische Universität München
France 85748 Garching bei München
Allemagne
florent.brunet@u-clermont1.fr

ABSTRACT

Le recalage d'images consiste à déterminer une transformation géométrique (ou déformation) qui aligne une image source et une image cible. Le recalage par *approche directe* estime la transformation géométrique en maximisant la ressemblance photométrique entre l'image source et l'image cible déformée. Cette approche est habituellement exprimée sous forme d'un problème d'optimisation dont la fonction de coût n'est définie que pour un sous-ensemble des pixels de l'image source : la *région d'intérêt* (RDI). Cette RDI doit impérativement être contenue dans le chevauchement des deux images à recalcer c'est-à-dire la partie de la scène visible dans les deux images. Si cette condition n'est pas vérifiée, la fonction de coût n'est pas définie et l'algorithme d'optimisation est mis en échec. Or, le chevauchement réel des deux images est inconnu avant recalage, ce qui rend très difficile la définition d'une RDI correcte. Nous proposons donc une nouvelle approche du recalage direct d'images qui ne nécessite pas la donnée d'une RDI, réglant ainsi les problèmes qui lui sont attachés. Les fondements théoriques de notre approche reposent sur une unification du traitement des pixels de l'image source invisibles dans l'image cible (*pixels hors-cible*), soit parce qu'ils sont occultés, soit parce qu'ils sont hors du champ de vue de la caméra cible.

KEYWORDS

Recalage direct, région d'intérêt, alignement, déformation, M-estimateur.

1 Introduction

Le problème du recalage d'images consiste à déterminer une transformation géométrique alignant deux images¹ : une image source S et une image cible T . Un exemple de recalage illustre ce problème à la figure 1². Il s'agit d'un problème fondamental en vision par ordinateur dont les applications sont nombreuses et variées : mosaïques d'images (images aériennes [5], par exemple), suivi d'objets [10], super-résolution [8], imagerie médicale [13], *etc.*

Nous considérons ici le recalage d'images par approche paramétrique, c'est-à-dire que nous fixons *a priori* un modèle de déformation puis nous recherchons les paramètres de ce modèle à partir de l'information contenue dans les images. De nombreux modèles de déformation peuvent être utilisés. Par exemple, sous certaines conditions, l'homographie peut être utilisée en environnement rigide (paramétrée par les 8 coefficients libres de sa matrice). En environnement déformable, les B-splines constituent un modèle générique mais efficace.

Il existe deux approches principales pour recalcer des images [16] : l'*approche géométrique* et l'*approche directe*. Dans l'approche géométrique [17], un ensemble d'éléments caractéristiques (typiquement des points) sont extraits des images et mis en correspondance. La déformation est ensuite estimée à partir de ces correspondances. Dans l'approche directe [2, 3, 11], approche considérée dans cet article, la déformation est estimée directement à partir de la couleur des pixels. Plus précisément, la déformation recherchée est celle qui minimise la différence de couleur entre les pixels de l'image source et les pixels qui leur

1. Une image I est considérée comme une fonction continue de $\Omega_I \subset \mathbb{R}^2$ dans \mathbb{R}^h où h est le nombre de canaux et Ω_I est le domaine de l'image. L'interpolation bilinéaire est utilisée pour évaluer l'image pour des coordonnées non-entières.

2. Il est préférable de regarder les illustrations de cet article en couleur.

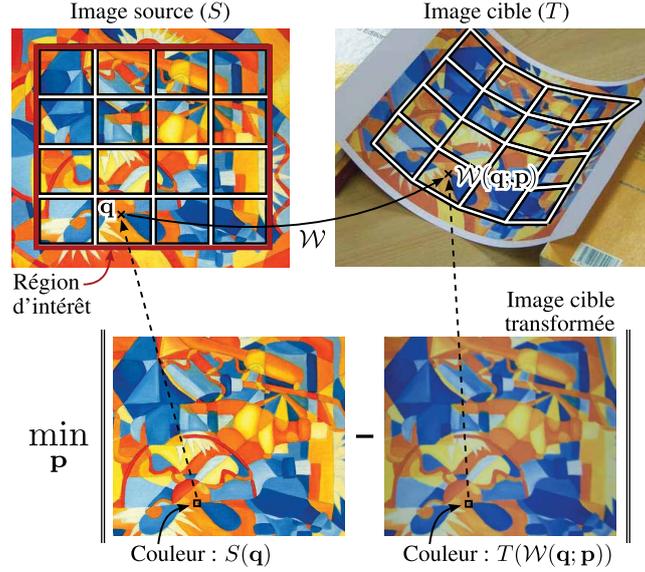


FIGURE 1 – Le principe général du recalage direct d’images consiste à utiliser la couleur des pixels pour déterminer les paramètres \mathbf{p} d’un modèle de déformation \mathcal{W} de manière à aligner une image source S et une image cible T . Cela s’écrit généralement sous la forme d’un problème de minimisation dont la fonction de coût mesure la différence de couleur entre une partie des pixels de l’image source (la région d’intérêt, RDI) et les pixels leur correspondant dans l’image cible. Ce problème est résolu en utilisant un algorithme d’optimisation itératif. La déformation représentée sur cette figure (grille de visualisation noire et blanche) correspond à la solution courante de l’algorithme d’optimisation lors d’une de ses dernières itérations (le recalage est donc pour ainsi dire parfait). Notons que même si la déformation \mathcal{W} est une fonction de Ω_S dans Ω_T , il est plus facile de déformer l’image cible plutôt que l’image source (et cela sans avoir besoin de la déformation inverse \mathcal{W}^{-1} : l’image cible déformée est l’image dont la couleur du pixel \mathbf{q} est la couleur du pixel $\mathcal{W}(\mathbf{q}; \mathbf{p})$ dans l’image cible originale). Notre contribution permet de résoudre l’épineux problème de la définition d’une RDI correcte.

correspondent dans l’image cible déformée (voir la figure 1). Il n’est pas toujours possible de calculer cette différence de couleur. C’est le cas, notamment, pour les pixels de l’image source qui, une fois transformés, n’appartiennent pas au domaine de l’image cible (*pixels hors-cible*). C’est pourquoi la différence de couleur n’est habituellement calculée que pour un sous-ensemble des pixels de l’image source : la Région d’Intérêt (RDI).

Soit $\mathcal{W} : \mathbb{R}^2 \times \mathbb{R}^p \rightarrow \mathbb{R}^2$ une déformation paramétrée par un vecteur $\mathbf{p} \in \mathbb{R}^p$. Si l’on considère qu’il n’existe pas de changement d’illumination significatif entre les images source et cible, alors le recalage d’images par approche directe peut s’exprimer ainsi :

$$\min_{\mathbf{p}} \sum_{\mathbf{q} \in \mathfrak{R}} d(\mathbf{q}; \mathbf{p})^2, \quad (1)$$

où $d(\mathbf{q}; \mathbf{p}) = \|S(\mathbf{q}) - T(\mathcal{W}(\mathbf{q}; \mathbf{p}))\|$ et où \mathfrak{R} désigne la RDI. Cette formulation fait l’hypothèse qu’à chaque pixel de la RDI correspond un pixel dans l’image cible et que ces pixels sont sensiblement de la même couleur. Cette hypothèse n’est plus vérifiée en présence, par exemple, de spéularités ou d’occultations. Ces pixels « cachés » peuvent être considérés comme des données erronées (*outliers* en anglais). Pour prendre en compte de telles données, la fonction de coût du problème (1) doit être rendue robuste en utilisant, par exemple, un M-estimateur [1] :

$$\min_{\mathbf{p}} \sum_{\mathbf{q} \in \mathfrak{R}} \rho(d(\mathbf{q}; \mathbf{p})). \quad (2)$$

Comme cela sera montré ultérieurement, notre contribution exploite les propriétés des M-estimateurs saturés³. Dans cet article et sans perte de généralité, nous utilisons le M-estimateur reposant sur la fonction bicarrée de Tukey :

$$\rho(x) = \begin{cases} \frac{c^2}{6} \left(1 - \left(1 - \frac{x^2}{c^2}\right)^3\right) & \text{si } |x| \leq c \\ \frac{c^2}{6} & \text{sinon,} \end{cases} \quad (3)$$

3. Un M-estimateur est dit saturé lorsqu’il admet une asymptote horizontale en $\pm\infty$.

où c est une constante fixant la sensibilité du M-estimateur aux données erronées. Cette valeur devrait idéalement être égale à 4.685 fois l'écart-type du bruit contenu dans les images [1]. En pratique, l'intensité du bruit n'étant pas précisément connue, nous considérons qu'il est égal à 20% de la valeur maximale d'un pixel. Notons qu'un pixel \mathbf{q} de la RDI produisant une différence de couleur $x = d(\mathbf{q}; \mathbf{p})$ telle que $\rho(x) = \frac{c^2}{6}$ peut être considéré comme une donnée erronée.

Le problème (1) est un problème de moindres carrés non linéaires pouvant être résolu itérativement en utilisant un algorithme de type Gauß-Newton [4]. Le problème (2), quant à lui, peut être résolu avec un algorithme de moindres carrés itérativement re-pondérés [7] (IRLS, Iteratively Reweighted Least-Squares).

Chevauchement et RDI. Le recalage d'images par approche directe a l'avantage de ne pas nécessiter de phase préalable de détection et d'appariement d'éléments caractéristiques. En revanche, cette approche nécessite qu'une RDI correcte soit donnée. Pour qu'une RDI soit correcte, elle doit être entièrement contenue dans le chevauchement réel des deux images. En effet, si un pixel de la RDI est hors du chevauchement, la fonction de coût du problème (1) ne peut être évaluée (puisque la transformation géométrique de certains pixels de l'image source tomberait en dehors du domaine de l'image cible). De plus, il est préférable d'utiliser une RDI aussi grande que possible afin de disposer d'un maximum d'informations pour effectuer le recalage. Déterminer une telle RDI est un problème délicat puisque le chevauchement des deux images n'est connu qu'après un recalage réussi des images (réussite dont la condition *sine qua non* est une RDI correctement déterminée !).

Travaux précédents. Nous verrons en §2.1 que la RDI est souvent définie comme un polygone dans le domaine de l'image source déterminé soit « à la main » par l'utilisateur, soit par des méthodes *ad hoc* [3]. Ces types d'approches manquent d'automatisme et ne sont pas forcément fiables. La *région d'intérêt adaptative* est une autre approche proposée par [14]. Avec cette méthode, la RDI est d'abord définie comme l'intégralité du domaine de l'image source. Cette RDI est ensuite ajustée dynamiquement en fonction de la déformation courante lors de chacune des itérations de l'algorithme d'optimisation. Comme nous le verrons plus en détail en §2.2, la fonction de coût engendrée par cette approche est mal posée et, de plus, très difficile à minimiser rigoureusement.

Notre contribution. Dans cet article, nous proposons une nouvelle solution au problème posé par la RDI en recalage direct d'images. Notre approche se distingue des approches précédentes puisqu'elle fonctionne sans RDI. Cela est rendu possible en considérant les pixels hors-cible comme des données erronées ; les fondements théoriques de ce principe sont expliqués en §3. De manière synthétique, la fonction de coût que nous proposons prend en compte *tous* les pixels de l'image source. Une pénalité fixée (et qui correspond à la pénalité habituellement donnée aux pixels erronés pour cause d'occultation ou de spéularité) est donnée aux pixels hors-cible. Nous unifions donc dans notre approche le traitement des pixels erronés et celui des pixels hors-cible. Le chevauchement réel entre les deux images recalées est naturellement obtenu comme l'ensemble des pixels qui ne sont pas erronés (*inliers*).

2 Région d'intérêt : état de l'art

2.1 Région d'intérêt polygonale

L'approche classique pour déterminer une RDI consiste à deviner l'amplitude maximale de la déformation. La RDI est ensuite définie comme un sous-domaine rectangulaire obtenu en supprimant une marge au domaine de l'image source dont la largeur correspond à l'amplitude maximale. Idéalement, cette largeur devrait être la plus proche possible de l'amplitude maximale réelle. Comme celle-ci est inconnue avant recalage, il est commun de surestimer la marge afin d'éviter les erreurs fatales dues à la présence de pixels hors-cible dans le RDI. Utiliser une grande marge (c'est-à-dire une petite RDI) réduit la quantité d'information disponible pour estimer la déformation. De plus, l'allure de la fonction de coût de (1) est sensible à la taille de la RDI. Ce phénomène est illustré par une expérience simple, inspirée de [14]. La figure 2 montre l'évolution du coût en fonction d'un unique paramètre de déformation (l'amplitude d'une translation) pour différentes tailles de marge. Les images source et cible sont identiques (image utilisée dans la figure 5) à l'exception d'un bruit blanc gaussien additif d'écart-type égal à 5% de la valeur maximale d'un pixel. La figure 2 montre qu'une petite marge (c'est-à-dire une grande RDI) conduit à une fonction de coût relativement lisse mais réduit l'espace des transformations admissibles. Au contraire, une marge plus grande (c'est-à-dire une RDI plus petite) augmente l'amplitude maximale des transformations admissibles mais produit des fonctions de coût présentant beaucoup plus de minima locaux. Les avantages et les inconvénients respectifs d'une petite et d'une grande marge sont résumés à la figure 3.

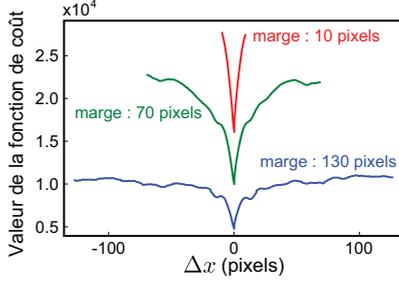


FIGURE 2 – Profil de la fonction de coût du problème (2) pour des RDI rectangulaires avec des marges mesurant de 10 à 130 pixels (pour des images de taille 640×480).

	Grande marge (petite RDI)	Petite margin (grande RDI)
Amplitude maximale de la transformation	+	-
Quantité d'information disponible pour estimer la transformation	-	+

FIGURE 3 – Avantages et inconvénients respectifs de l'utilisation d'une grande ou d'une petite marge pour définir une RDI rectangulaire. Remarquons qu'aucune d'entre elles ne cumule tous les avantages.

2.2 Région d'intérêt adaptative

Une alternative aux RDI rectangulaires a été proposée par [14]. Dans cette approche, la RDI constante \mathfrak{R} est remplacée par une RDI adaptative $\mathfrak{R}_A(\mathbf{p})$:

$$\min_{\mathbf{p}} \sum_{\mathbf{q} \in \mathfrak{R}_A(\mathbf{p})} d(\mathbf{q}; \mathbf{p})^2. \quad (4)$$

Pour un jeu donné de paramètres \mathbf{p} , $\mathfrak{R}_A(\mathbf{p})$ est l'ensemble des pixels de l'image source qui, une fois transformés, appartiennent au domaine de l'image cible, c'est-à-dire $\mathfrak{R}_A(\mathbf{p}) = \{\mathbf{q} \in \Omega_S \mid \mathbf{q}' \in \Omega_T\}$ avec $\mathbf{q}' = \mathcal{W}(\mathbf{q}; \mathbf{p})$. Puisque la RDI adaptative \mathfrak{R}_A dépend des paramètres \mathbf{p} du modèle de déformation, elle évolue au cours des itérations de l'algorithme qui résout le problème (4). Bien que cette méthode ne requiert pas la définition *a priori* d'une RDI, elle n'est cependant pas satisfaisante. Premièrement, le problème (4) est mal posé dans la mesure où sa fonction de coût possède une infinité de minima qui ne correspondent pas à la solution recherchée. Ces minima apparaissent dès lors que les deux images ne se chevauchent plus. Cela est illustré à la figure 4 avec une expérience similaire à celle utilisée en §2.1. La figure 4 (schéma de gauche) montre que la fonction de coût utilisée dans l'approche adaptative est nulle, et donc minimale, dès lors qu'il n'existe plus aucun chevauchement entre les domaines de l'image source et de l'image cible transformée ($|\Delta x| > 640$). Deuxièmement, la dépendance en \mathbf{p} de \mathfrak{R}_A rend le problème (4) très difficile à résoudre rigoureusement (fonction de coût non-continue dont le nombre de termes varie en fonction des itérations). Les auteurs de [14] proposent d'utiliser l'algorithme de Gauß-Newton en alternant l'estimation de \mathbf{p} et de \mathfrak{R}_A . Troisièmement, la fonction de coût de (4) n'est pas robuste et, en tant que telle, ne permet pas de gérer les données erronées provoquées par les occultations ou les spécularités.

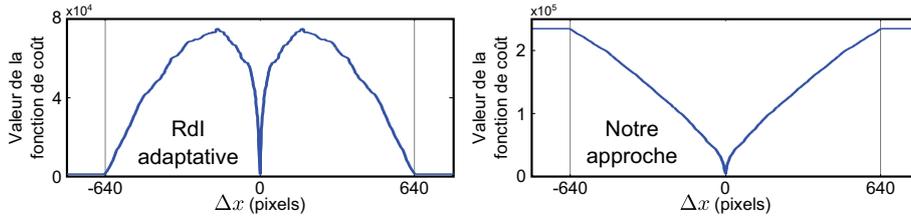


FIGURE 4 – Allures des fonctions de coût de (à gauche) l'approche à RDI adaptative de [14] et (à droite) notre approche. Les images source et cible sont identiques et mesurent 640 pixels de large. Le modèle de déformation est une translation d'amplitude Δx le long de l'axe des abscisses. La fonction de coût proposée dans [14] s'annule lorsque la transformation est telle qu'il n'existe plus de chevauchement entre l'image source et l'image cible transformée. Un des avantages de notre fonction de coût est de résoudre ce genre de problèmes.

3 Recalage direct d'images sans région d'intérêt

Nous proposons une nouvelle approche pour le recalage direct d'images qui ne nécessite pas de RDI. Ainsi, notre méthode résout tous les problèmes précédemment mentionnés. De la même manière que l'approche adaptative de [14], notre fonction de coût utilise l'intégralité des pixels de l'image source. Cependant, comme le montre la figure 4, notre coût n'a pas de faux

minima. L'idée sous-jacente de notre approche est de pénaliser les pixels hors-cible en leur attribuant un coût fixe. Le coût associé aux autres pixels reste la classique différence robuste de couleur, comme dans le problème (2). Pénaliser trop fortement les pixels hors-cible tend à maximiser la taille du chevauchement entre l'image source et l'image cible transformée. Afin de ne pas introduire de biais dans cette direction, la pénalité donnée aux pixels hors-cible est la même que celle attribuée par le M-estimateur aux données erronées. Cela revient à dire que les pixels hors-cible sont des données erronées. Or, ce point de vue est parfaitement justifiable.

Justification. Imaginons une caméra cible avec un champ de vue illimité. Une telle caméra produirait des images de taille infinie. Imaginons maintenant qu'un plan disposant d'une petite ouverture rectangulaire soit placé entre la caméra et la scène, comme cela est illustré à la figure 5b. La partie de la scène visible à travers l'ouverture correspond à l'image cible. Le reste de la scène est invisible puisque *occulté* par le plan, exactement comme pour un pixel erroné pour cause d'occultation (figure 5a). Avec ce raisonnement, il devient tout naturel de considérer les pixels hors-cible comme de simples pixels erronés.

Mise en équation. Une traduction directe mais incomplète de notre idée est :

$$\min_{\mathbf{p}} \sum_{\substack{\mathbf{q} \in \Omega_S \\ \mathbf{q}' \in \Omega_T}} \rho(S(\mathbf{q}) - T(\mathbf{q}')) + \sum_{\substack{\mathbf{q} \in \Omega_S \\ \mathbf{q}' \notin \Omega_T}} \frac{c^2}{6}. \quad (5)$$

Résoudre (5) n'est pas aisé dans la mesure où sa fonction de coût est constituée de deux sommes dont le nombre de termes varie en fonction de \mathbf{p} (puisque $\mathbf{q}' = \mathcal{W}(\mathbf{q}; \mathbf{p})$). Remarquons que (5) peut être réécrit de manière équivalente en :

$$\min_{\mathbf{p}} \sum_{\substack{\mathbf{q} \in \Omega_S \\ \mathbf{q}' \in \Omega_T}} \rho(S(\mathbf{q}) - T(\mathbf{q}')) + \sum_{\substack{\mathbf{q} \in \Omega_S \\ \mathbf{q}' \notin \Omega_T}} \rho(x_0), \quad (6)$$

où x_0 est une valeur saturant le M-estimateur. Avec la fonction bicarrée de Tukey, n'importe quelle valeur x_0 telle que $|x_0| \geq c$ convient (car ainsi $\rho(x_0) = \frac{c^2}{6}$). Les deux sommes de (6) peuvent être regroupées de la manière suivante :

$$\min_{\mathbf{p}} \sum_{\mathbf{q} \in \Omega_S} \rho\left([\mathbf{q}' \in \Omega_T](S(\mathbf{q}) - T(\mathbf{q}')) + [\mathbf{q}' \notin \Omega_T]x_0\right), \quad (7)$$

où $[\]$ est l'opérateur telle que $[a] = 1$ si a est vrai et $[a] = 0$ sinon. Nous réécrivons (7) en introduisant l'image cible étendue T_∞ :

$$T_\infty(\mathbf{q}) = \begin{cases} T(\mathbf{q}) & \text{si } \mathbf{q} \in \Omega_T \\ \alpha & \text{sinon} \end{cases} \quad \text{et} \quad \Omega_{T_\infty} = \mathbb{R}^2, \quad (8)$$

où α est une valeur telle que $\|S(\mathbf{q}) - \alpha\| > x_0$. Finalement, notre méthode se résume au problème de minimisation suivant :

$$\min_{\mathbf{p}} \sum_{\mathbf{q} \in \Omega_S} \rho(\|S(\mathbf{q}) - T_\infty(\mathbf{q}')\|). \quad (9)$$

Dans la mesure où l'évaluation des images est effectuée par interpolation bilinéaire, la fonction de coût de (9) est dérivable et peut être minimisée en utilisant un algorithme de moindres carrés itérativement re-pondérés.

M-estimateur et chevauchement. Une propriété intéressante de notre approche est qu'elle découvre de manière automatique le chevauchement entre les images. En effet, avec notre approche, le recouvrement exact est constitué des pixels non-erronés. Avec la fonction bicarrée de Tukey, le chevauchement dans l'image source est l'ensemble des pixels \mathbf{q} tels que $|\rho(S(\mathbf{q}) - T_\infty(\mathbf{q}')) - \frac{c^2}{6}| \leq \varepsilon$ avec ε une petite constante (par exemple, $\varepsilon = 10^{-6}$). Le chevauchement dans l'image cible est la transformée du chevauchement dans l'image source. Des exemples de chevauchements déterminés avec notre approche sont montrés en §4.3.

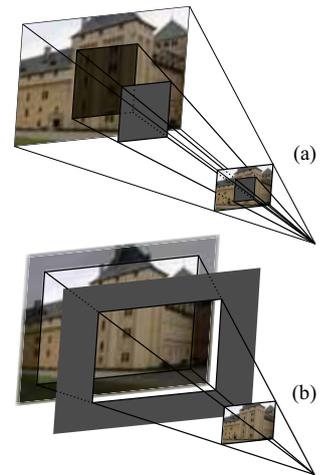


FIGURE 5 – Les pixels hors du champ de vue (a) de la caméra cible peuvent être considérés comme des données erronées classiques (b).

4 Résultats expérimentaux

4.1 Transformations géométriques

Homographie. L'homographie \mathcal{W}_H est un modèle de transformation géométrique globale qui explique la déformation entre deux images prises par une caméra qui tourne autour de son centre optique ou qui observe une scène plane [9]. Cette déformation est définie de la manière suivante (avec $\mathbf{p} \in \mathbb{R}^8$ et $\mathbf{q} = (x, y)$) :

$$\mathcal{W}_H(\mathbf{q}; \mathbf{p}) = \frac{1}{p_7x + p_8y + 1} \begin{pmatrix} p_1x + p_2y + p_3 \\ p_4x + p_5y + p_6 \end{pmatrix}. \quad (10)$$

Déformations libres (Free-Form Deformations). Les B-splines cubiques sont un cas particulier de déformations libres [15]. Elles sont paramétrées par un ensemble de points de contrôle 2D qui agissent comme des « attracteurs » sur la déformation. Ils sont situés sur une grille de taille $n_x \times n_y$ et groupés dans un vecteur $\mathbf{p} \in \mathbb{R}^{2n_x n_y}$. La forme analytique de cette déformation est exprimée comme le produit tensoriel de deux B-splines cubiques mono-dimensionnelles :

$$\mathcal{W}_B(\mathbf{q}; \mathbf{p}) = \sum_{j=1}^{n_y} \sum_{i=1}^{n_x} \begin{pmatrix} p_{2(j(n_x-1)+i)-1} \\ p_{2(j(n_x-1)+i)} \end{pmatrix} N_i(x)N_j(y), \quad (11)$$

où N_i est la i -ème fonction de base de l'espace vectorielle des B-splines [6, 12].

4.2 Données synthétiques

Génération des données. Une déformation (homographie ou B-spline) est déterminée en interpolant des correspondances de points générés aléatoirement (en perturbant avec un bruit uniforme des points disposés sur une grille régulière). Cette déformation est utilisée pour générer l'image source à partir d'une image texturée choisie parmi un stock de 15 images. L'image texturée est utilisée comme image cible. L'amplitude de la déformation γ (en pixels) est définie comme la moyenne des perturbations appliquées à la grille régulière. Des occultations sont simulées en remplaçant une proportion α (en pourcentage) de pixels des images source et cible avec des pixels provenant d'une image différente. Finalement, un bruit gaussien centré d'écart-type σ est ajouté aux images. Nous utilisons des images couleurs dont les intensités sont codées en valeurs flottantes comprises entre 0 et 1. La taille des images est de 320×240 pixels.

Processus expérimental. L'amplitude de la déformation γ , le niveau de bruit σ et la proportion de données erronées α sont étudiés de manière indépendante. Les valeurs par défaut sont : $\gamma = 8$ pixels, $\alpha = 10\%$ et $\sigma = 0.1$. Plusieurs approches sont comparées : les RDI rectangulaires (RECT), la RDI adaptative de [14] (ADAP) et notre approche (SANSRDI). Plusieurs variantes de l'approche à RDI rectangulaires sont considérées : petite (10%) et grande (25%) marge sans M-estimateur (RECTP, RECTG) et avec M-estimateur (RECTPM, RECTGM). Les valeurs reportées sont les moyennes sur 100 essais différents.

Échec du processus d'optimisation. Comme expliqué en §2.1, une RDI de taille constante peut entraîner un échec de l'algorithme d'optimisation. La figure 6 montre la proportion de tels échecs sur les expériences des trois prochains paragraphes. Remarquons qu'une convergence vers une fausse solution (minimum local) n'est pas comptée comme un échec. Nous observons que le taux d'échecs avec de petites marges (RECTP) est supérieur à celui obtenu avec de grandes marges (RECTG). Le taux d'échec est aussi diminué lorsqu'un M-estimateur est utilisé (RECTPM et RECTGM). Cela provient du fait qu'avec un M-estimateur, les pas de l'algorithme tendent à être plus petits que sans M-estimateur. L'apparition de pixels hors-cible au cours des itérations est alors moins probable. Dans la suite des expériences, lorsqu'un algorithme faillit, les résultats de la dernière itération valide sont pris en compte.



FIGURE 6 – Taux d'échec de l'algorithme d'optimisation. Les algorithmes ADAP et SANSRDI ne faillissent jamais puisqu'ils ne reposent pas sur une RDI fixée.

Nombre d'itérations. La figure 7 reporte le nombre d'itérations effectuées par les algorithmes d'optimisation. Globalement, la convergence est plus rapide avec les homographies qu'avec les B-Splines ; la nature « globale » des homographies en est la raison. L'apparente rapidité des méthodes à base de RDI rectangulaire est expliquée par le fait que ces approches peuvent s'arrêter prématurément lorsque la RDI n'est plus valide. De manière générale, les résultats obtenus avec notre approche sont meilleurs que ceux obtenus avec l'algorithme ADAP, seule autre méthode ne nécessitant pas de RDI. La vitesse de convergence de notre approche est amortie lorsque l'amplitude de la transformation recherchée est importante. Cela vient de la pénalisation des pixels hors-cible, ceux-ci étant nombreux pour les transformations de grande amplitude.

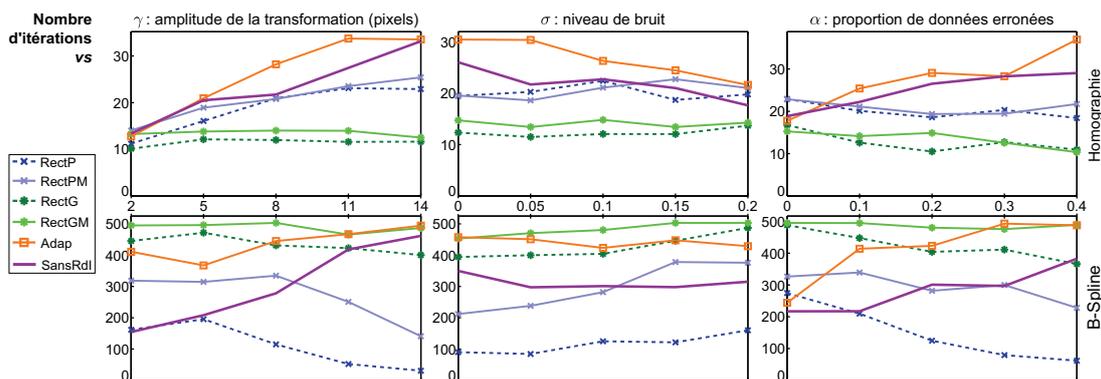


FIGURE 7 – Nombre d’itérations effectuées par les algorithmes d’optimisation. Le nombre d’itérations des algorithmes à base de RDI rectangulaires est faible car ces méthodes s’arrêtent prématurément lorsque la RDI n’est plus valide.

Erreur géométrique. La figure 8 reporte l’erreur géométrique, c’est-à-dire l’écart en pixels entre la transformation estimée et celle utilisée pour générer les données. Nous observons que, pour tous les algorithmes et toutes les transformations, l’erreur géométrique est peu influencée par le niveau de bruit mais dépend de l’amplitude des transformations et de la proportion de données erronées. Cela est particulièrement vrai pour les approches n’utilisant pas de M-estimateur. Comparée aux autres méthodes, notre approche est celle donnant les meilleurs résultats, atteignant des erreurs de l’ordre d’un pixel (et souvent moins). Cette expérience montre que *la pénalisation des pixels hors-cible ne biaise pas la précision du recalage*.

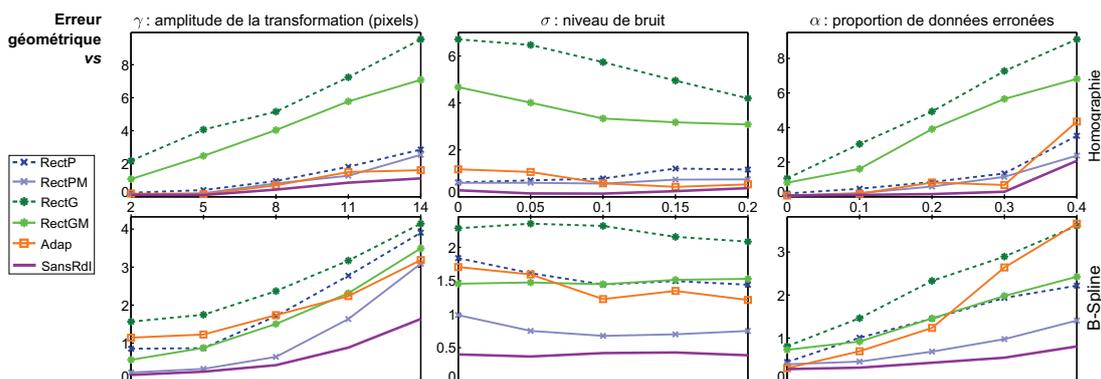


FIGURE 8 – Impact de différents facteurs sur l’erreur géométrique. Notre méthode (SANSRDI) est celle donnant les meilleurs résultats. Globalement, les approches utilisant un M-estimateur sont les meilleures.

Erreur photométrique. L’erreur photométrique moyenne pour la dernière itération des algorithmes est reportée à la figure 9. Notre approche aboutit aux erreurs les plus faibles quel que soit le facteur étudié et quel que soit la transformation géométrique considérée.

4.3 Données réelles

Zones de recouvrement. La figure 10 montre les zones de recouvrement déterminées lors de la dernière itération des algorithmes RECTPM, RECTGM, ADAP et SANSRDI. SANSRDI est la seule méthode estimant correctement l’homographie recherchée. Nous observons aussi que notre méthode est la seule à déterminer précisément la zone de recouvrement, y compris les pixels erronés. RECTPM ne détermine pas le bon résultat car la taille de la marge est inférieure au déplacement. Le faible nombre de pixels utilisés par RECTGM conduit à une estimation imprécise. Le résultat de l’algorithme ADAP est biaisé par les données erronées.

Panorama maximal. Une scène est filmée par une caméra en rotation autour de son centre optique et dont le mouvement de gauche à droite est uniforme. La figure 11 montre la mosaïque constituée de la première image de la vidéo et de l’image la plus

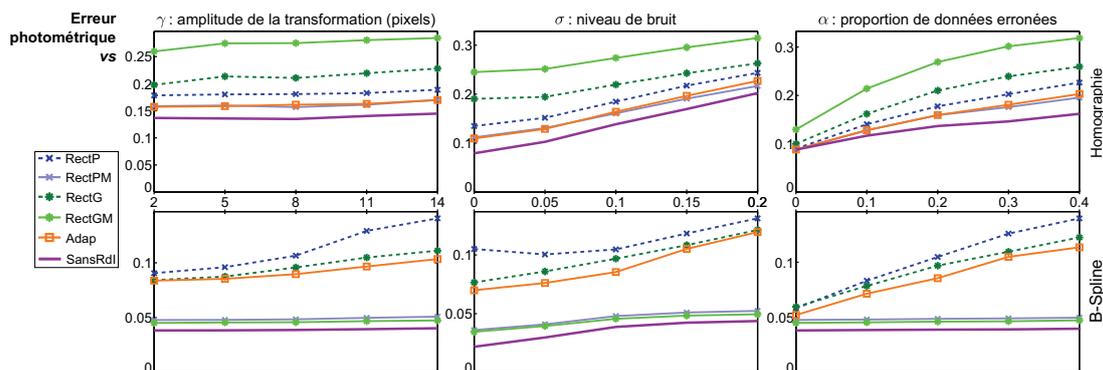


FIGURE 9 – Évolution de l’erreur photométrique moyenne en fonction de différents facteurs.



FIGURE 10 – Exemples de recalages pour différents algorithmes. La première et la deuxième lignes correspondent aux images source et cible respectivement. La dernière ligne représente l’image de différence entre l’image source et l’image cible transformée. Les parties rouges représentent les pixels n’appartenant pas aux chevauchements trouvés par les algorithmes.

éloignée de la première pour laquelle les algorithmes étudiés ont correctement estimé l’homographie. Le panorama le plus étendu est obtenu grâce aux algorithmes ADAP et SANSRDI. Les résultats de ces deux méthodes sont similaires car il y a très peu de données erronées. Les panoramas les moins étendus sont obtenus avec RECTP et RECTG puisque le déplacement maximal est conditionné par la taille de la marge.

Mosaïque sur scène déformable. La figure 12 est un exemple de mosaïque d’images en environnement déformable réalisé avec notre méthode.

Suivi d’un motif. La figure 13 illustre le suivi d’un motif dans une vidéo. Trois approches sont comparées : notre approche (SANSRDI), RDI rectangulaire à grande marge (RECTGM) et à petite marge (RECTPM). Le modèle de déformation est une B-spline bicubique avec 5×5 points de contrôle. Le motif à suivre (c’est-à-dire l’image source) au cours du temps est une partie sélectionnée à la main dans la première image de la vidéo. Ce motif est recalé dans chaque nouvelle image (qui tient le rôle d’image cible) en utilisant comme solution initiale la déformation déterminée à l’image précédente. La figure 13 montre que les

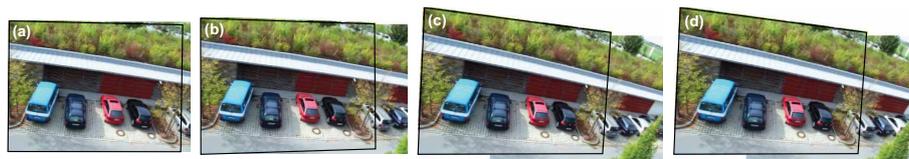


FIGURE 11 – Panorama le plus large obtenu avec (a) RECTP, (b) RECTG, (c) ADAP et (d) SANSRDI.



FIGURE 12 – Mosaïque (c) en environnement déformable. (a) : image source. (b) : image cible.

approches à base de RDI rectangulaires sont mises en échec dès qu'une partie de la RDI n'est plus visible dans l'image cible. De tels problèmes ne peuvent pas arriver avec notre approche. La figure 13 illustre aussi le chevauchement réel entre les images découvert automatiquement par notre méthode aussi bien dans l'image source que dans l'image cible. Les 4^{ème} et 5^{ème} colonnes montrent comment notre approche uniformise la gestion des pixels de l'image source invisibles dans l'image cible, qu'ils soient occultés ou hors du champ de vue.

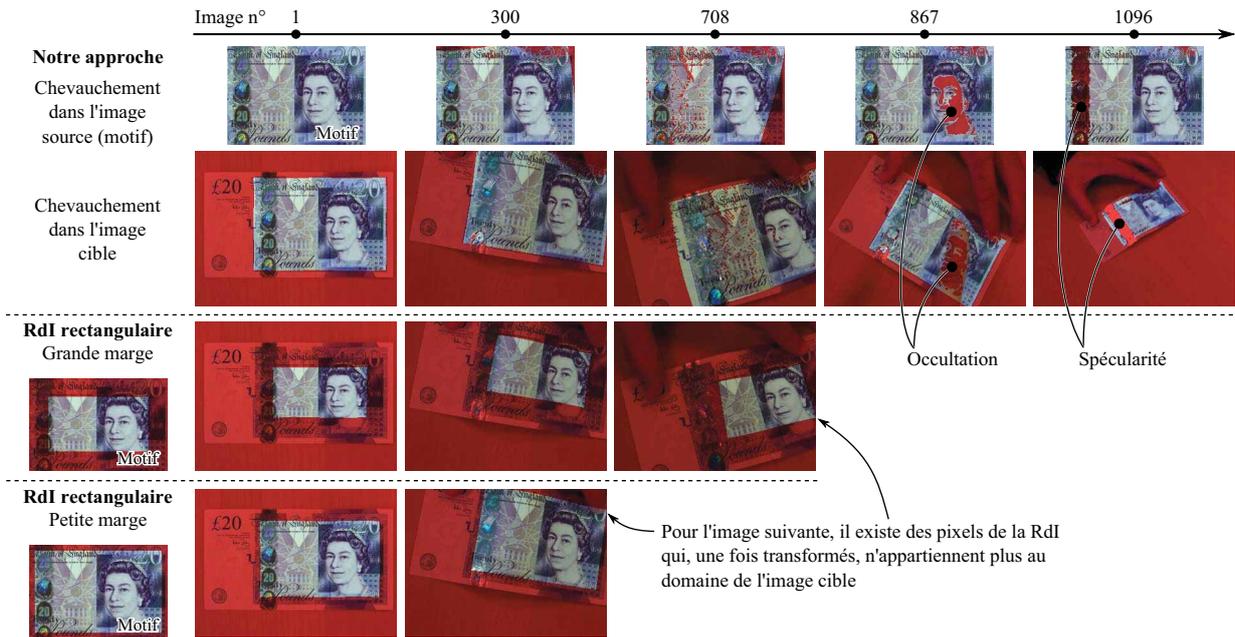


FIGURE 13 – Suivi d'un motif dans une vidéo. Pour des raisons évidentes de place, nous ne montrons ici que quelques images « intéressantes » de la vidéo. Pour notre méthode (1^{ère} et 2^{ème} lignes), nous reportons systématiquement le motif (c'est-à-dire l'image source) et l'image cible afin d'illustrer la découverte automatique du chevauchement réel dans les deux images. Pour les méthodes reposant sur une RDI rectangulaire (3^{ème} et 4^{ème} lignes), le motif n'est reporté qu'une seule fois (sous le titre de la ligne) puisque celui-ci ne varie pas au cours du temps. Pour les approches à base de RDI rectangulaires, le suivi du motif s'arrête prématurément dès lors qu'il existe des pixels de la RDI qui sortent du champ de vue de la caméra cible (image n° 300 avec une grande marge et n° 708 avec une petite marge). Les images n° 867 et n° 1096 illustrent respectivement la gestion par notre approche des occultations et des spécularités.

Suivi de nuages. La figure 14 est un exemple d'application de notre méthode pour effectuer le suivi d'un front nuageux. Nous disposons d'une séquence d'images satellites $\{I_0, \dots, I_n\}$ prises à un intervalle de 15 minutes⁴. Contrairement à l'expérience précédente, les images successives ne sont pas toutes recalées par rapport à la première image de la séquence : l'image I_t est recalée avec l'image I_{t-1} . La déformation entre l'image I_0 et I_t est donc la composition des déformations obtenues entre I_0 et I_1, I_1 et I_2, \dots, I_{t-1} et I_t . L'avantage de notre méthode pour ce genre d'images est qu'elle peut être utilisée sans se préoccuper de savoir si les nuages présents à un instant donné vont sortir du champ de vue par la suite.

4. Images du 17 mai 2010 récupérées sur <http://www.sat24.com>

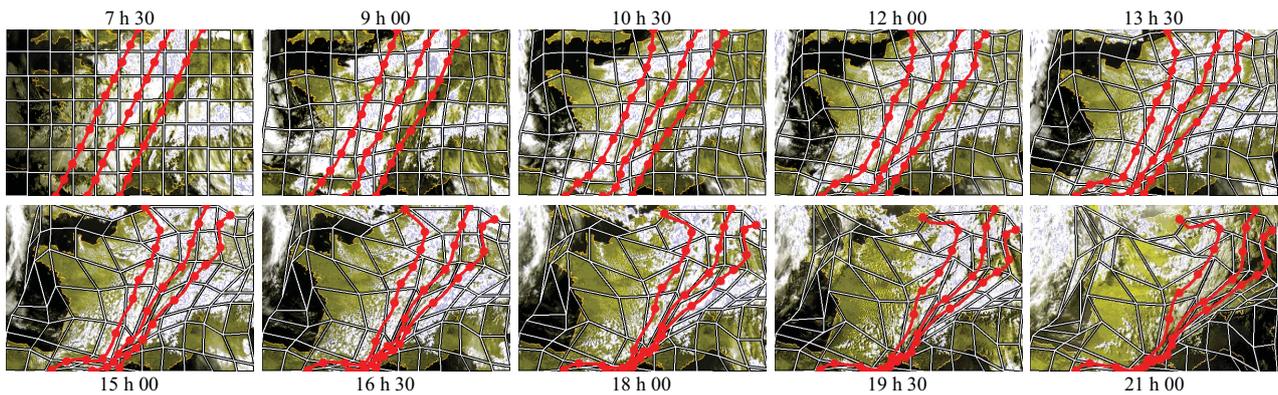


FIGURE 14 – Exemple d’utilisation de notre méthode pour effectuer le suivi d’un front nuageux.

5 Conclusion

Nous avons proposé un nouvel algorithme de recalage d’images qui, contrairement aux précédentes approches, ne nécessite pas de RDI pour fonctionner. Notre approche repose sur l’unification du traitement des pixels de l’image source invisibles dans l’image cible, qu’ils soient occultés ou qu’ils soient hors du champ de la caméra cible. Cela résulte en une élégante modification de la fonction de coût habituellement minimisée en recalage direct d’images. Les performances de notre approche en terme de précision du recalage et de robustesse ont été confirmées par un jeu complet d’expériences.

Références

- [1] K. V. Arya, P. Gupta, P. K. Kalra, and P. Mitra. Image registration using robust M-estimators. *Pattern Recognition Letters*, 28 :1957–1968, 2007.
- [2] S. Baker, R. Gross, and I. Matthews. Lucas-Kanade 20 years on : A unifying framework. *International Journal of Computer Vision*, 56 :221–255, 2004.
- [3] A. Bartoli. Groupwise geometric and photometric direct image registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30 :2098–2108, 2008.
- [4] Å. Björck. *Numerical Methods for Least Squares Problems*. SIAM, 1996.
- [5] C. Cariou and K. Chehdi. Recalage semi-rigide d’images de scanner aéroporté. In *20ème Colloque sur le traitement du signal et des images*. GRETSI, 2005.
- [6] C. de Boor. *Numerical Methods for Image Registration*. Springer, 2001.
- [7] M. Deans, C. Kunz, R. Sargent, and L. Pedersen. Terrain model registration for single cycle instrument placement. *Intelligent Robots and Systems*, 2003.
- [8] R. Hardie, K. Barnard, and E. Armstrong. Joint MAP registration and high-resolution image estimation using a sequence of undersampled images. *IEEE Transactions on Image Processing*, 12 :1621–1633, 1997.
- [9] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision – 2nd Edition*. Cambridge University Press, 2004.
- [10] B. Heigl, D. Paulus, and H. Niemann. Tracking points in sequences of color images. *Pattern Recognition and Image Understanding*, 1999.
- [11] M. Irani and P. Anandan. About direct methods. In *Workshop on Vision Algorithms*, 1999.
- [12] R. Malgouyres. *Algorithmes pour la synthèse d’images et l’animation 3D*. Dunod, 2005.
- [13] J. Modersitzki. *A Practical Guide to Splines – Revised Edition*. Oxford University Press, 2004.
- [14] B. Pires and P. Aguiar. Registration of images with small overlap. *IEEE Workshop on Multimedia Signal Processing*, 2004.
- [15] D. Rueckert, L. Sonoda, I. Hayes, D. Hill, M. Leach, and D. Hawkes. Nonrigid registration using Free-Form Deformations : Application to breast MR images. *IEEE Transactions on Medical Imaging*, 18 :712–721, 1999.
- [16] R. Szeliski. Image alignment and stitching : A tutorial. *Foundations and Trends in Computer Graphics and Vision*, 2 :1–104, 2006.
- [17] P. H. S. Torr and A. Zisserman. Feature based methods for structure and motion estimation. *Vision Algorithms*, 1999.