

Estimation robuste pour la détection et le suivi par caméra

Sio-Song Ieng

Jean-Philippe Tarel

LIVIC (INRETS-LCPC)

LCPC

13 route de la Minière

58 Bd Lefebvre

78000 Versailles-Satory

75732 Paris Cedex 15

Ieng@inrets.fr

Jean-Philippe.Tarel@lcpc.fr

Pierre Charbonnier

ERA 27 LCPC

Laboratoire des Ponts et Chaussées

11 rue Jean Mentelin, B.P. 9

67035 Strasbourg

Pierre.Charbonnier@equipement.gouv.fr

résumé : *Dans la conception d'un système d'aide à la conduite qui repose sur la détection des marquages routiers, la robustesse aux perturbations est d'une importance cruciale, et, pour être utilisable, la détection doit être accompagnée d'une mesure de confiance. Nous formulons le problème de détection des marquages dans les images comme celui de l'estimation des paramètres d'un modèle de courbe, à partir de points extraits de l'image par une procédure originale. Cette approche nous permet de bénéficier de la robustesse des M-estimateurs et d'obtenir une mesure naturelle de confiance au travers de la matrice de covariance de l'estimée.*

Après une relecture originale des M-estimateurs dans le formalisme lagrangien, nous proposons deux familles paramétriques de modèles de bruit qui permettent une transition continue entre modèles gaussiens et non gaussiens. Nous approfondissons aussi plusieurs points rarement abordés en vision par ordinateur, et pourtant incontournables en pratique, comme l'estimation des paramètres du modèle de bruit et la définition de la matrice de covariance approchée. Nous montrons expérimentalement que la précision de la matrice de covariance est beaucoup plus tributaire du choix des paramètres que celle de l'estimation. Une nouvelle forme de matrice de covariance approchée, moins sensible de ce point de vue, est proposée. Enfin, nous présentons des matrices peu coûteuses en temps de calcul et qui peuvent être utiles dans de nombreuses applications.

mots-clefs : Détection, évaluation de la détection, estimation robuste, M-estimateur, matrice de covariance, Reconnaissance, traitement d'images, système d'aide à la conduite.

title : Robust estimation for camera-based detection and tracking

abstract : *When designing a Driving Assistance System based on lane-markings detection, the robustness of the outputs is a crucial issue. Moreover, to be integrated into complex control systems, they must be accompanied with some confidence measure. Formulating road markings detection as the problem of estimating the parameters of a lane model, using features extracted from the image by an original procedure, we can benefit from the robustness of M-estimators and consider as a natural confidence measure the covariance matrix of the estimate.*

After revisiting M-estimators in an original, Lagrangian formalism, we propose two parametric families of noise models, that allow a continuous transition between Gaussian and non-Gaussian models. Then, we focus on several points of major practical importance, though seldom addressed in computer vision, such as the estimation of noise parameters and the definition of the approximate covariance matrix. We experimentally show that the accuracy of the covariance matrix depends much more on the tuning of parameters than the one of the estimation itself. A new approximation of the covariance matrix, less sensitive to the noise model, is proposed. Finally, we exhibit new matrices, faster to compute, that might be used with advantages in many other applications.

keywords : Detection, evaluation of detection, robust parameter estimation, M-estimators, covariance matrix, recognition, image processing, driving Assistance Systems.

1 introduction

Dans les applications émergentes, telles que les systèmes intelligents de transport, les caméras et leurs algorithmes d'analyse d'images sont considérés de plus en plus comme un capteur spécialisé, au même titre que d'autres. Comme tout système qui utilise un capteur, le dispositif de vision doit fournir des résultats très fiables lorsque la sécurité des usagers est impliquée. La robustesse à toute perturbation est donc nécessaire et le résultat doit être accompagné d'une *mesure de confiance* afin d'être intégré de façon sûre dans des systèmes de plus haut niveau, comme les systèmes de commande de véhicule.

Beaucoup d'algorithmes d'analyse d'images, tels que la segmentation d'images ou la détection de courbes, peuvent être formalisés, en partie ou complètement, comme un problème d'estimation. L'estimation par la méthode classique des moindres carrés est très souvent utilisée. Cette méthode fournit une matrice de covariance exacte de l'estimée, qui constitue une mesure de confiance naturelle. Cependant, en pratique, le bruit est rarement gaussien, ce qui rend la tâche d'estimation, ou de régression, plus difficile. Dans le cadre robuste, il est possible de traiter des modèles de bruit non gaussien qui conduisent généralement à des problèmes d'optimisation non-linéaires. Heureusement, des algorithmes itératifs, comme les méthodes des moindres carrés pondérés itératifs, permettent de résoudre ces problèmes d'estimation robuste, dans le contexte des M-estimateurs [18].

La classe des fonctions de bruit, pour lesquelles la convergence des M-estimateurs est prouvée, a été étendue en utilisant la théorie semi-quadratique [5, 11], dont nous proposons ici une relecture dans le formalisme lagrangien. Cependant, le choix de la fonction de potentiel est une question importante, rarement abordée en vision par ordinateur. Dans cet article, nous analy-

sons ce problème, commun à tous les algorithmes de détection ou de suivi robuste. Ceci nous conduit à introduire des modèles de bruit paramétriques qui donnent une meilleure description des perturbations observées, dans notre application. De plus, le choix du potentiel peut avoir des conséquences importantes en terme de sensibilité de l'estimation par rapport aux choix des paramètres de modèle de bruit, surtout lorsque le potentiel utilisé est non-convexe. Nous discutons également de cette difficulté et nous proposons des méthodes pour estimer ces paramètres.

Le problème de définition d'une mesure de confiance dans le cas de l'estimation robuste est également peu abordé en vision par ordinateur. Pour des raisons de non-linéarité, une définition exacte de la matrice de covariance semble difficile à obtenir et des approximations sont nécessaires. Une façon d'aborder ce problème est d'étudier le comportement asymptotique de l'estimateur robuste : Huber [18] propose plusieurs approximations de la matrice de covariance. Mais ces matrices ne sont valables que lorsque le nombre de données est très grand. Dans un contexte différent, comme le filtrage de Kalman robuste, où la question de la matrice de covariance est très importante, d'autres approximations ont été proposées dans [6] et dans [25] par exemple. Toutefois, aucune justification n'est fournie et la validation a seulement été faite sur des simulations.

Nous présentons une expérience sur des données réelles afin de comparer ces approximations de matrice de covariance. Or, d'après cette comparaison, aucune des approximations proposées ne donne des résultats suffisamment pertinents lorsque les perturbations sont fortes. Nous avons donc été amenés à en proposer une nouvelle. Contrairement à celles proposées par Huber, elle n'est pas asymptotique et ses performances expérimentales sont bien plus satisfaisantes. Notre application d'assistance au contrôle latéral du véhicule impose une contrainte de temps réel, aussi l'approximation de la matrice de covariance doit-elle non seulement être pré-

cise, mais aussi rapide à calculer. Nous prouvons finalement qu'en utilisant des équivalences entre les approximations des matrices de covariance, des expressions moins coûteuses en temps de calcul peuvent être déduites.

Dans la partie 2, nous présentons le contexte de l'application ainsi qu'un algorithme original d'extraction des marquages. Les données extraites, supposées bruitées, sont alors utilisées par un algorithme robuste d'estimation de courbes, présenté dans la partie 3. Puis, nous présentons, dans la partie 4, une étude de la robustesse des algorithmes d'estimation, en introduisant trois familles paramétriques de modèles de bruit qui permettent une transition continue entre des modèles gaussiens et non gaussiens. Ces derniers permettent de mieux modéliser les perturbations sur les données, et améliorent donc la convergence de l'algorithme. Enfin, dans la partie 5, nous proposons une méthode d'évaluation de la confiance accordée aux résultats de l'estimation en utilisant la matrice de covariance. Des nouvelles approximations de la matrice de covariance sont présentées et comparées avec les matrices existantes. Une preuve expérimentale de leur robustesse est proposée.

2 détection et suivi des marquages avec des données bruitées

Après la description de notre application, nous justifions notre approche de la détection, décrivons l'extracteur d'éléments de marquages, puis présentons le modèle analytique de la route.

2.1 systèmes d'aide à la conduite

Utiliser des caméras dans les systèmes d'aide à la conduite a pour but de fournir des informations fiables et précises, à la fois sur la route et sur l'environnement du véhicule. Un exemple important d'application est le module de détection et de suivi des marquages qui doit fournir, en temps réel, la distance latérale du véhicule par rapport à la route et, éventuellement, sa courbure. Le choix de la position de la caméra sur le véhicule et de son orientation (cf. figure 1) conditionne le type de paramètres qui peuvent être estimés ainsi que la précision de mesure. Dans la configuration où la caméra est positionnée verticalement sur le côté du véhicule, la distance aux marquages latéraux peut être estimée très précisément (avec une erreur de l'ordre de 2 cm en moyenne). Ce n'est pas le cas lorsque la caméra est dirigée vers l'avant du véhicule. En revanche, la courbure et, plus généralement, la géométrie de la route n'est mesurable que dans cette dernière configuration. C'est pourquoi la plupart des systèmes de vision pour la détection et le suivi des marquages sont conçus avec une caméra qui pointe vers l'avant du véhicule.

Dans le système de vision que nous présentons, au contraire de [9, 26, 4, 1], plusieurs caméras, montées et positionnées dans des configurations différentes, peuvent être employées en même temps. Leurs images sont alors traitées avec le même algorithme, afin de donner des estimations précises à la fois de la géométrie de la route et de la position du véhicule [19]. De surcroît, par rapport à [9, 26, 4], notre approche permet une évaluation de la qualité de la détection.

2.2 classification des perturbations

L'avantage d'avoir une image de la route est de disposer d'un nombre important d'informations géométriques et photométriques. Cependant, cette masse d'informations représente aussi des inconvénients. En effet, l'information utile peut être perdue parmi une quantité importante de perturbations induites notamment par les arbres, les panneaux de publicité ou les nuages. Notons qu'une partie de ces perturbations peut être supprimée par l'extraction de caractéristiques locales bien choisies dans les images, avant la détection proprement dite. Un autre type de perturbations susceptibles d'affecter la tâche de détection est, au contraire, le manque d'information, souvent dû, dans le cas de la détection et du suivi des marquages, à l'effacement de la peinture ou l'occultation des bandes par des obstacles. Nous pouvons donc classer les perturbations dans deux grandes classes :

1. bruits du capteur et perturbations dus à la présence de données parasites,
2. manque d'informations : peinture effacée, occultations, très mauvaise illumination.

Comment notre algorithme de détection traite-t-il ces problèmes ? Les perturbations de la première classe sont traitées en deux étapes qui consistent à utiliser un algorithme d'extraction locale des marquages puis un algorithme d'estimation robuste de courbe. Pour traiter les perturbations de la seconde classe, nous utilisons une interpolation temporelle en associant le détecteur à un filtre de Kalman. Pour simplifier la présentation, nous nous limitons ici au cas de la détection et le lecteur est invité à se référer à [25] pour une description du système de suivi temporel.

2.3 extraction du centre des marquages

Un algorithme classique de détection de contours fournit trop d'informations lorsqu'on ne s'intéresse qu'aux marquages. Il existe peu d'algorithmes de détection de bas niveau qui utilisent des critères géométriques concernant l'objet d'intérêt. Pourtant, sélectionner les données utiles avec des critères géométriques et non seulement photométriques est particulièrement efficace [14]. Dans le cas des marquages routiers, on peut utiliser la largeur de la bande, qui est approximativement constante, comme critère de sélection. Dans la première étape de l'extraction, le calcul du gradient horizontal d'intensité des pixels est effectué. Si la valeur du gradient est supérieure à S_G , alors on cherche un plateau d'intensité dont la longueur est dans un certain intervalle $[S_m, S_M]$. Enfin, le centre de ce plateau est extrait en vue de la détection.

L'objectif est d'obtenir le plus grand nombre de données qui appartiennent aux marquages de route tout en réduisant le plus possible les perturbations ; dans tous les cas le problème des perturbations résiduelles sera traité par l'algorithme d'estimation robuste. Pour ne pas manquer la détection des marquages dans certaines conditions de luminosité, nous devons fixer un seuil S_G aussi faible que possible. En outre, la détection se fait sur l'image entière. Ces faibles contraintes imposées à l'algorithme d'extraction sont nommées hypothèse de détection *conservative*. Cette hypothèse est aussi utilisée dans [14, 10], par exemple. Il serait envisageable de limiter l'extraction des données dans des zones d'intérêt grâce aux méthodes de suivi de courbe. Cependant, on risquerait des pertes de détection lorsqu'il y a disparition de l'ancien marquage et apparition d'un nouveau plus éloigné. De plus, l'algorithme d'extraction de marquage proposé est suffisamment rapide pour que chaque image soit analysée entièrement en temps réel.

Pour chaque ligne x de l'image analysée, soit y_{init} la colonne du premier pixel pour lequel le

gradient d'intensité est supérieur au seuil S_G . Soit y_{fin} la colonne du pixel qui termine le plateau d'intensité débuté en y_{init} . Alors, il y a détection si $y_{fin} - y_{init}$ est dans l'intervalle $[S_m, S_M]$ où $S_m = C_m x + D_m$ et $S_M = C_M x + D_M$. Le rôle de S_m et de S_M est prépondérant dans la détection des marquages et dans le rejet des fausses détections locales. Aussi le choix de C_m, C_M, D_m et D_M doit-il être en accord avec la configuration de la caméra. Par exemple, pour une vue plongeante de la voie comme le montre la figure 1, $C_m = C_M = 0$, D_m et D_M sont calculés en tenant compte de la calibration de la caméra et des petites variations de la largeur des marquages. Soient les paramètres suivants :

- θ est l'angle formé par l'axe de visée de la caméra avec le plan de la route,
- h est la hauteur de la caméra par rapport au sol,
- α_y et α_x sont respectivement l'inverse de la largeur d'un pixel et de sa longueur (la focale est égale à un),
- (x_0, y_0) est le centre de l'image en pixels,
- l_m et l_M sont les bornes inférieure et supérieure de la largeur des marquages sur la route.

Ainsi, compte-tenu des paramètres intrinsèques et extrinsèques du calibrage de la caméra,

C_m, C_M, D_m et D_M sont calculés à partir des formules suivantes dans le cas général :

$$C_i = \frac{l_i \alpha_y \cos(\theta)}{\alpha_x h} \quad D_i = -\frac{l_i \alpha_y \cos(\theta) x_0}{\alpha_x h} - \frac{l_i \alpha_y \sin(\theta)}{h} \quad (1)$$

L'algorithme d'extraction des marquages est alors le suivant pour chaque ligne de l'image :

- $y = 1$
- Tant que $y < \text{tailleImageLigne}$,
 1. Calcul du gradient horizontal $G(y)$
 2. Si $G(y) > S_G$ alors,

- $y_{init} = y$
 - $I = Intensité(y_{init}) + \frac{G(y_{init})}{2}$
 - $y_{fin} = y_{init} + 1$
 - Tant que $Intensité(y_{fin}) > I$ et $y_{fin} < tailleImageLigne$,
 - $y_{fin} = y_{fin} + 1$
 - Si $y_{fin} - y_{init} \in [S_m, S_M]$ alors
 - Le centre du marquage, $\frac{(y_{fin} + y_{init})}{2}$, est extrait pour la détection
 - $y = y_{fin} + 1$
 - sinon, $y = y_{init} + 1$
3. sinon, $y = y + 1$

La figure 2 montre la supériorité de l’algorithme d’extraction de marquage proposé par rapport aux détecteurs classiques de contours, dans une situation adverse du fait de la faible luminosité. Grâce à notre algorithme, on constate d’une part peu de fausses détections dans la zone fortement illuminée ; d’autre part, le centre du marquage visible est entièrement extrait. Le fait de détecter le centre du marquage au lieu des bords évite la sélection d’informations redondantes, qui seraient de toute façon inutiles pour les étapes suivantes. Dans la prochaine partie, nous décrivons le modèle de courbe que nous utilisons dans l’estimation de la forme de la route.

2.4 modèle de route

Un modèle plan de la route en néglige la courbure verticale. Pour la plupart des applications de contrôle de véhicule, la forme de la route est estimée sur une distance de 10 à 40m. Ainsi,

l'hypothèse d'une route plane demeure une approximation acceptable pour notre système. Son intérêt principal est de permettre l'estimation de la forme de la route avec une seule caméra, à partir des centres de marquages précédemment extraits. Pour ce faire, nous supposons que les données extraites sont des mesures bruitées d'une courbe décrite explicitement par le modèle paramétrique linéaire suivant :

$$\tilde{y} = \sum_{i=0}^d f_i(x) \tilde{a}_i = X(x)^t \tilde{A} \quad (2)$$

où (x, \tilde{y}) sont les coordonnées d'un point appartenant à la courbe décrivant la forme de la route dans l'image. $\tilde{A} = (\tilde{a}_i)_{0 \leq i \leq d}$ est le vecteur des paramètres de la courbe et $X(x) = (f_i(x))_{0 \leq i \leq d}$ est le vecteur de la base de fonctions $f_i(x)_{i=0, \dots, d}$.

Les familles de courbes définies par les fonctions linéaires (2) sont particulièrement utiles et pratiques dans les problèmes d'estimation de forme par régression. En effet, le faible coût de calcul qu'implique cette représentation linéaire accélère grandement la détection. En pratique, nous approchons les formes des routes par des polynômes $y = \sum_{i=0}^d a_i x^i$. Bien que les clothoïdes semblent être plus adaptées pour une modélisation précise de la forme de la route, leur complexité entraînerait des calculs plus longs. Dans la mesure où il est possible d'approcher localement les clothoïdes par des polynômes [9], le modèle polynomial convient lorsqu'une caméra latérale est utilisée. L'image d'une courbe polynomiale par la projection perspective est nommée courbe *polynôme hyperbolique* définie par $y = c_0 x + c_1 + \sum_{i=2}^d \frac{c_i}{(x-x_h)^i}$, où c_i est lié de façon linéaire à a_i comme démontré dans [24]. Le modèle de polynôme hyperbolique est donc adéquat pour les caméras frontales. Cependant, pour des raisons de stabilité, dans cette configuration, nous préférons à ce modèle celui des *polynômes fractionnaires*, définis par $y = \sum_{i=0}^d c_i x^{\frac{i}{d}}$.

Pour augmenter la stabilité numérique des calculs dans l'algorithme d'estimation, nous nor-

malisons les coordonnées de l'image, qui varieront dans $[-1, 1] \times [-1, 1]$ pour les polynômes et dans $[0, 1] \times [-1, 1]$ pour les polynômes fractionnaires.

Dans notre modèle (2), l'ordonnée x est supposée non aléatoire. Seule l'abscisse y est considérée bruitée : $y = X(x)^t \tilde{A} + b$. Dans la suite, le bruit de mesure b est supposé indépendant, identiquement distribué (iid) et centré, d'écart-type σ . Dans la partie suivante, nous expliquons l'algorithme d'estimation robuste pour la détection de marquages, fondé sur le modèle précédemment décrit.

3 estimation des paramètres

Dans cette partie, nous définirons les notations et rappellerons la méthode des moindres carrés. Ensuite, nous proposerons une relecture de l'estimation robuste dans le formalisme lagrangien.

3.1 méthode des moindres carrés

A partir de l'ensemble des n points extraits (x_i, y_i) , $i = 1, \dots, n$, il s'agit de construire un estimateur A_{LS} des paramètres de la courbe représentant la route, la valeur idéale étant notée \tilde{A} , selon le principe du maximum de vraisemblance. La probabilité conditionnelle d'un point (x_i, y_i) supposant un vecteur de paramètres quelconque A est :

$$p_i((x_i, y_i)/A) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2} \left(\frac{X_i^t A - y_i}{\sigma} \right)^2}$$

Pour simplifier les équations, nous notons $X_i = X(x_i)$. Nous pouvons alors écrire la même probabilité conditionnelle pour tous les points détectés sachant que le bruit est iid :

$$p \propto \frac{1}{\sigma^n} \prod_{i=1}^{i=n} e^{-\frac{1}{2} \left(\frac{X_i^t A - y_i}{\sigma} \right)^2} \quad (3)$$

où p est la vraisemblance de l'ensemble des points détectés, sachant le vecteur des paramètres A , et où \propto signifie l'égalité à un facteur multiplicatif près. Maximiser p par rapport à A revient à minimiser l'erreur quadratique suivante :

$$e_{LS}(A) = \frac{1}{2\sigma^2} \sum_{i=1}^{i=n} (X_i^t A - y_i)^2$$

En annulant la dérivée de e_{LS} par rapport à A , nous obtenons le système linéaire suivant :

$$XX^t A = XY \quad (4)$$

où $Y = (y_i)_{1 \leq i \leq n}$ est le vecteur des abscisses y , la matrice $X = (X_i)_{1 \leq i \leq n}$ est la *matrice des données* (design matrix). $S = XX^t$ est toujours symétrique et positive. Si S est, de plus, non singulière, alors (4) a une solution unique $A_{LS} = S^{-1}XY$. C'est la meilleure estimation au sens du maximum de vraisemblance quand le bruit est gaussien.

Puisque seul Y est aléatoire, l'espérance mathématique de A_{LS} est $E[A_{LS}] = S^{-1}XE[Y]$. De plus, les points $(X, E[Y])$ sont sur la courbe idéale. Donc, $\tilde{A} = S^{-1}XE[Y]$. Ainsi, $E[A_{LS}]$ est égal à \tilde{A} : l'estimateur A_{LS} de \tilde{A} est donc non biaisé.

La matrice de covariance C_{LS} de A_{LS} est $E[(A_{LS} - E[A_{LS}])(A_{LS} - E[A_{LS}])^t] = S^{-1}XE[(Y - E[Y])(Y - E[Y])^t]X^tS^{-t}$. Nous avons : $E[(Y - E[Y])(Y - E[Y])^t] = \sigma^2 I_{n \times n}$, puisque le bruit est iid, de variance σ^2 ($I_{n \times n}$ est la matrice identité d'ordre n). Finalement, l'inverse de la matrice de covariance de A_{LS} est :

$$C_{LS}^{-1} = \frac{1}{\sigma^2} S = Q_{LS} \quad (5)$$

Q_{LS} est aussi la matrice d'information de Fisher pour l'ensemble des n points détectés. Elle est également définie comme l'espérance mathématique de la dérivée seconde de e_{LS} suivant A .

Enfin, il est important de souligner une propriété intéressante de la méthode des moindres carrés : la solution du problème est indépendante de l'écart-type σ . En revanche, σ est indispensable au calcul de la matrice de covariance de l'estimateur. Ainsi, l'estimation de σ doit être effectuée avant le calcul de la matrice de covariance, grâce aux résidus obtenus après la résolution du problème d'estimation des paramètres. L'estimateur au sens du maximum de vraisemblance de σ est :

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^{i=n} (X_i^t A_{LS} - y_i)^2 \quad (6)$$

Il est bien connu, comme la figure 3 l'illustre, que l'estimation aux moindres carrés ne fournit pas une estimation correcte lorsque les perturbations sont trop importantes.

3.2 estimation robuste des paramètres

L'hypothèse du bruit iid et centré reste valable dans ce qui va suivre, mais nous supposons à présent que le bruit n'est plus gaussien. Le modèle de bruit peut être l'un de ceux présentés dans la partie 4.1. Le modèle de bruit est caractérisé par une fonction $\phi(t)$ telle que la probabilité du point (x_i, y_i) d'être sur une courbe donnée connaissant son vecteur de paramètres A est :

$$p_i((x_i, y_i)/A) \propto \frac{1}{s} e^{-\frac{1}{2}\phi\left(\left(\frac{X_i^t A - y_i}{s}\right)^2\right)}$$

En appliquant le principe du maximum de vraisemblance et l'approche lagrangienne, nous obtenons un algorithme d'estimation robuste identique à l'algorithme déduit par l'approche semi-quadratique ou les M-estimateurs. L'avantage de la formulation lagrangienne est de permettre de prouver facilement la convergence locale et de donner une interprétation en terme d'optimisation duale. Comme dans l'approche semi-quadratique [11, 5], $\phi(t)$ vérifie les hypothèses suivantes :

- **H0** : $\phi(t)$ est définie et continue sur $[0, +\infty[$ ainsi que ses dérivées première et seconde,
- **H1** : $\phi'(t) > 0$ (ϕ est strictement croissante),
- **H2** : $\phi''(t) < 0$ (ϕ est strictement concave).

Ces trois hypothèses sont très différentes de celles supposées dans la preuve de convergence avec l'approche des M-estimateurs. En effet, dans [18], la preuve de la convergence est fondée sur l'hypothèse de la convexité du potentiel $\rho(b) = \phi(b^2)$. Dans notre cas, la concavité et la monotonie de $\phi(t)$ impliquent que $\phi'(t)$ est bornée, mais le potentiel $\phi(b^2)$ n'est pas nécessairement convexe par rapport à b . Nous pouvons déjà signaler que la famille exponentielle lissée ($\alpha < 1$) et la famille de lois T-Student généralisées ($\beta > 0$) que nous présenterons dans la partie 4.1, vérifient ces trois hypothèses. Comme dans les M-estimateurs, le rôle de la fonction ϕ est d'affaiblir l'importance des bruits forts. Le paramètre d'échelle, s , définit la distance au-delà de laquelle une mesure bruitée peut être considérée comme une erreur.

Maximiser la vraisemblance aboutit à minimiser, par rapport à A , l'erreur :

$$e_R(A) = \frac{1}{2} \sum_{i=1}^{i=n} \phi\left(\left(\frac{X_i^t A - y_i}{s}\right)^2\right) \quad (7)$$

Notons que le cas gaussien correspond au cas particulier : $\phi(t) = t$, mais que cette dernière fonction ne vérifie pas l'hypothèse (H2). $e_{LS}(A)$ est en fait un cas limite de $e_R(A)$. Contrairement au cas gaussien, le critère robuste n'est pas quadratique. On pourrait envisager de résoudre le problème de minimisation en utilisant un algorithme de descente du gradient, mais ce type d'algorithme peut être relativement lent lorsque le gradient est proche de zéro au voisinage d'un minimum local. Les algorithmes quasi-Newton sont préférables car ils convergent de manière quadratique plutôt que de manière linéaire. Cependant, avec les algorithmes quasi-Newton, la convergence locale n'est plus forcément assurée. Ainsi, nous allons prouver dans la suite que

l'algorithme quasi-Newton utilisé converge toujours vers un minimum local. Notons que la convergence globale est possible à l'aide de techniques de recuit simulé, au prix d'un coût de calcul plus important.

Dans les deux parties suivantes, nous allons présenter un algorithme obtenu à partir de l'erreur (7) en utilisant le formalisme lagrangien. L'approche consiste à exprimer le problème comme une recherche de point selle de la fonction de Lagrange. Alors, en appliquant une minimisation alternée de la fonction duale, nous obtenons l'algorithme qui est aussi connu sous le nom des moindres carrés pondérés itératifs (MCPI) [18]. Cet algorithme s'obtient aussi par l'approche semi-quadratique [5].

3.3 moindres carrés pondérés itératifs (MCPI)

Nous réécrivons, dans un premier temps, le problème de minimisation de $e_R(A)$ comme un problème de maximisation de $-e_R$, pour introduire la fonction convexe $-\phi$ plutôt que de traiter le cas concave. Ensuite, nous introduisons les variables auxiliaires w_i qui seront égales à $(\frac{X_i^t A - y_i}{s})^2$ afin d'exprimer la valeur $-e_R(A)$ comme la solution d'un problème de minimisation avec contraintes, et ce quel que soit A . Cette apparente complication nous permet en réalité d'introduire les multiplicateurs de Lagrange, et donc de dériver la convergence locale de l'algorithme MCPI. En effet, selon (H1), pour tout A , $-e_R(A)$ est égale au minimum par rapport à $W = (w_i)_{1 \leq i \leq n}$ de :

$$E(A, W) = \frac{1}{2} \sum_{i=1}^{i=n} -\phi(w_i)$$

avec les contraintes $h_i(A, W) = w_i - (\frac{X_i^t A - y_i}{s})^2 \leq 0$.

L'algorithme est dérivé de ce problème d'optimisation sous contrainte, comme montré dans

l'annexe 8.1. L'algorithme MCPI s'écrit :

1. Initialiser A_0 , et poser $j = 1$,
2. Pour tout i ($1 \leq i \leq n$), calculer la variable auxiliaire $w_{i,j} = \left(\frac{X_i^t A_{j-1} - y_i}{s}\right)^2$ et les poids $\lambda_{i,j} = \phi'(w_{i,j})$,
3. Résoudre le système linéaire $\sum_{i=1}^{i=n} \lambda_{i,j} X_i X_i^t A_j = \sum_{i=1}^{i=n} \lambda_{i,j} X_i y_i$,
4. Si $\|A_j - A_{j-1}\| > \varepsilon$, incrémenter j de 1, et retourner en 2, sinon $A_{RLS} = A_j$.

Le test de convergence peut aussi être fait sur la variation de l'erreur e_R . Un seuil sur le nombre d'itérations peut être ajouté. Comme tout algorithme quasi-Newton, il est possible de prouver [20] que la vitesse de convergence de l'algorithme au voisinage du minimum est quadratique lorsque S est définie positive.

Finalement, la figure 4 illustre l'importance de l'estimation robuste dans le cas d'images bruitées. Même lorsque l'initialisation est proche, avec les deux premiers potentiels qui sont convexes, on n'obtient pas la solution correcte. Cela illustre l'importance du choix du modèle de bruit.

4 familles de modèles de bruit et estimation de leurs paramètres

Dans cette partie, le choix du modèle de bruit utilisé dans l'algorithme MCPI est traité. Après avoir présenté trois familles de modèles de bruit, nous nous intéresserons aux algorithmes qui permettent d'estimer leurs paramètres.

4.1 modèles de bruit

Dans une application réelle, on rencontre très rarement des perturbations qui peuvent être modélisées par un bruit gaussien. Quel modèle de bruit pourrait-il alors se substituer à la forme gaussienne, pour une description plus réaliste des bruits observés dans le contexte du traitement d'images ? Plusieurs modèles ont été proposés, entre autres, dans [17, 23]. Parfois, ce sont des combinaisons de modèles théoriques [16]. Nous allons nous intéresser ici à des distributions de probabilité (pdf) paramétriques simples qui seront pratiques à utiliser dans l'algorithme d'estimation. En estimation robuste, les algorithmes dépendent uniquement de la dérivée de la fonction ϕ , comme expliqué précédemment. D'après la partie 3.2, la fonction ϕ est reliée à $-\ln(pdf(\sqrt{t}))$ pour $t \geq 0$ ou, si nous posons $t = b^2$, à $-\ln(pdf(b))$, où $pdf(b)$ est la distribution de probabilité du bruit b . La famille exponentielle, aussi dénommée laplacienne généralisée, gaussienne généralisée, ou encore "stretched exponential" [23], est une famille de distributions de probabilité intéressante :

$$E_{\alpha,s}(b) = \frac{\alpha}{s\Gamma(\frac{1}{2\alpha})} e^{-(\frac{b^2}{s^2})^\alpha} \quad (8)$$

Les deux paramètres de cette famille sont l'échelle s et l'exposant α . La valeur de α détermine la décroissance à l'infini de la pdf du bruit. De plus, la variation de α permet une transition continue entre deux modèles classiques : le modèle gaussien ($\alpha = 1$) et le modèle laplacien ($\alpha = \frac{1}{2}$). En éliminant la constante qui ne dépend pas de b , la fonction ϕ associée est $\phi_{E_\alpha}(t) = t^\alpha$, avec $t = \frac{b^2}{s^2}$. D'après la partie 3.2, pour garantir la convergence de l'estimation robuste, la dérivée de ϕ doit être définie sur $[0, +\infty[$, ce qui n'est pas le cas lorsque $\alpha < 0$ pour la famille exponentielle. Pour cette raison, nous avons introduit dans [25], la famille exponentielle lissée

$S_{\alpha,s}$ afin de mieux décrire les distributions observées :

$$S_{\alpha,s}(b) \propto \frac{1}{s} e^{-\frac{1}{2}\phi_{S\alpha}(\frac{b^2}{s^2})} \quad (9)$$

où avec $t = \frac{b^2}{s^2}$

$$\phi_{S\alpha}(t) = \frac{1}{\alpha}((1+t)^\alpha - 1)$$

Comme dans le cas de la famille exponentielle, le paramètre α de la famille exponentielle lissée permet une transition continue entre différents modèles de bruit : Gauss ($\alpha = 1$), Laplace lissée ($\alpha = \frac{1}{2}$) et Geman & McClure [12] ($\alpha = -1$). La forme de ces lois est montrée à la figure 5.

Pour $\alpha < 0$, $S_{\alpha,s}$ peut toujours être normalisée sur un support borné, de telle sorte que l'on peut encore la considérer comme une distribution de probabilité. Lorsque α décroît, la probabilité d'avoir des erreurs très grandes (*outliers*) augmente.

TAB. 1 – La liste de quelques modèles de bruit de la famille exponentielle lissée pour l'estimation robuste.

| α | poids= $\phi'_{S\alpha}(t)$ | nom des distributions de probabilité |
|---------------|-----------------------------|--------------------------------------|
| 1 | 1 | Gauss |
| $\frac{1}{2}$ | $\frac{1}{\sqrt{1+t}}$ | Laplace lissée |
| 0 | $\frac{1}{1+t}$ | T-Student généralisée |
| -1 | $\frac{1}{(1+t)^2}$ | Geman & McClure |

Dans les algorithmes robustes d'estimation, le résidu b est pondéré par $\lambda = \phi'(\frac{b^2}{s^2})$ comme expliqué dans la partie 3.3. Les poids calculés pour différentes valeurs de α , pour la famille exponentielle lissée, sont indiqués dans le tableau 1. Lorsque $\alpha = 0$, la dérivée est de la forme $\frac{1}{1+t}$, mais, par contre, la distribution de probabilité n'est pas définie. Cependant, cette fonc-

tion de pondération est égale, à un facteur près, aux fonctions dérivées des lois d'une famille paramétrée que nous présentons maintenant, nommée famille des T-Student généralisées :

$$T_{\beta,s}(b) = \frac{\Gamma(\beta)}{\sqrt{\pi}\Gamma(\beta - \frac{1}{2})s} \frac{1}{(1 + \frac{b^2}{s^2})^\beta} \quad (10)$$

Cette famille de distributions de probabilité vérifie aussi les hypothèses pour l'algorithme d'estimation robuste ($\beta > 0$). Son appellation vient du fait qu'un paramètre d'échelle s supplémentaire est introduit par rapport à la famille classique des lois T-Student. Elle a été introduite en traitement d'images dans [17].

Nous constatons que le cas $\beta = 1$ correspond à la loi de Cauchy. Les différentes distributions sont montrées à la figure 6. La famille des lois T-Student généralisées peut s'écrire sous la forme exponentielle :

$$T_{\beta,s}(b) \propto \frac{1}{s} e^{-\frac{1}{2}\phi_{T_\beta}(\frac{b^2}{s^2})} \quad (11)$$

où $\phi_{T_\beta}(t) = 2\beta \ln(1+t)$. Notons que la dérivée de ϕ_{T_β} est égale à $\frac{1}{1+t}$, à un facteur près.

Les paramètres de cette famille de distributions sont s et β ($\beta > 0$). Ils correspondent respectivement aux paramètres s et α de la famille exponentielle lissée. Pour $0 < \beta \leq \frac{1}{2}$, la distribution de probabilité est seulement définie sur un support borné. Il est intéressant de remarquer que malgré des comportements différents à l'infini suivant la valeur de β , la pondération est identique à un facteur multiplicatif β près et par conséquent, que β ne joue aucun rôle dans l'algorithme robuste. Ainsi, le même algorithme d'estimation robuste est utilisé, quelle que soit la valeur de β . Cependant, il sera expliqué dans la partie 5 que β a une influence importante dans l'estimation de la matrice de covariance. Pour simplifier la présentation, nous utiliserons

la famille exponentielle lissée dans la suite.

4.2 estimation des paramètres des modèles de bruit

Dans la partie précédente, le modèle de bruit était supposé connu. En pratique, cependant, il est nécessaire d'estimer s ainsi que l'exposant, α ou β , de la distribution du bruit. Ceci est réalisé en étudiant la distribution des résidus sur des images de référence, pour lesquelles les paramètres A sont supposés connus ou correctement estimés.

Lorsque le support de la distribution de probabilité n'est pas borné et que α (ou β) est connu, la vraisemblance de l'échantillon s'écrit :

$$p \propto \frac{1}{s^n} \prod_{i=1}^{i=n} e^{-\frac{1}{2}\phi\left(\frac{(X_i^t A - y_i)^2}{s^2}\right)}$$

Nous dérivons l'opposé de son logarithme par rapport à s et déduisons l'équation implicite en \hat{s} suivante :

$$\hat{s}^2 = \frac{1}{n} \sum_{i=1}^{i=n} \phi'\left(\frac{(X_i^t A - y_i)^2}{\hat{s}^2}\right) (X_i^t A - y_i)^2 \quad (12)$$

Cette équation implique que l'estimation au sens du maximum de vraisemblance \hat{s} de s est un point fixe. Notons que d'autres estimations de s comme MAD [15], ont été proposées. Dans le cas d'un support borné (par exemple, lorsque $\alpha < 0$ pour la famille exponentielle lissée), une équation faisant intervenir une intégration sur le support est nécessaire. Il est intéressant de remarquer que la famille exponentielle (8) offre une expression explicite de \hat{s} :

$$\hat{s} = \left(\frac{2\alpha}{n} \sum_{i=1}^{i=n} b_i^{2\alpha}\right)^{\frac{1}{2\alpha}} \quad (13)$$

L'estimateur du maximum de vraisemblance de l'échelle doit être utilisé avec précaution. En effet, lorsque les données sont très peu bruitées, on constate que l'estimateur \hat{s} de s est clairement sous-estimé. Ce problème, très général, est dû à la discrétisation des données obtenues

dans les images numériques. Nous avons simulé cet effet en générant des données bruitées par un bruit de Cauchy ($\beta = 1$), pour des valeurs de plus en plus grandes du paramètre d'échelle s et en arrondissant les valeurs obtenues. Ensuite, \hat{s} est estimé à partir de ces données. La figure 7 montre les résultats et confirme notre observation sur les données réelles : lorsque s est inférieur à un pixel, \hat{s} est sous-estimé.

Ce résultat suggère que, d'une part lors de l'estimation du paramètre d'échelle s , nous devons empêcher l'estimateur de s de prendre des valeurs plus petites qu'un pixel. D'autre part, il est préférable de ne pas effectuer une estimation de s entre deux itérations avant la convergence de l'estimateur des paramètres de la courbe, au contraire de ce qui a été suggéré dans [18]. En effet, un facteur d'échelle sous-estimé peut figer l'estimée de la courbe à sa valeur initiale, alors qu'elle n'a pas encore convergé vers l'optimum, comme nous l'expliquons dans la partie 4.4.

On pourrait également appliquer l'approche du maximum de vraisemblance pour estimer le paramètre de l'exposant, α ou β , en supposant s connu. Cependant, en général, il n'est pas facile d'exprimer de façon analytique α ou β en fonction des autres paramètres, de telle sorte que nous devons nous cantonner à une méthode de minimisation numérique comme par exemple une méthode de descente du gradient, pour estimer ce paramètre.

Nous pouvons aussi estimer simultanément les deux paramètres, α et s , par une minimisation non linéaire à partir des résidus obtenus sur plusieurs images de référence. Ceci est illustré par la figure 8 avec 150 images. De même que pour l'estimation de s , les estimées de (α, s) doivent aussi être utilisées avec précaution. En effet, il existe des jeux de données particuliers où ces valeurs ne peuvent pas être estimées correctement.

Pour estimer \hat{s} et $\hat{\alpha}$, séparément ou simultanément, trois approches différentes peuvent être envisagées pour collecter les résidus :

1. Collecter les résidus à partir d'une série d'images de référence statistiquement représentatives de l'application. Estimer les paramètres et les utiliser comme des valeurs fixes dans l'estimation robuste.
2. Effectuer une première estimation robuste des paramètres des courbes dans l'image à traiter en utilisant des valeurs (α, s) comme en 1. Collecter les résidus sur cette image, pour une estimation des paramètres du modèle de bruit. Réitérer la procédure jusqu'à convergence.
3. Lorsqu'il s'agit de traiter une séquence d'images, on peut collecter les résidus des m images précédentes pour estimer des valeurs (α, s) qui seront utilisées directement sur l'image courante. On suppose alors une certaine cohérence temporelle des paramètres du modèle de bruit.

La première approche semble convenir aux problèmes de détection pure ou aux paramètres dont la variation est faible. Mais dans les applications de suivi, certains paramètres du modèle de bruit varient au cours du temps. La seconde approche semble alors mieux adaptée. Néanmoins, lorsque la cible à détecter disparaît, l'estimation des paramètres devient mauvaise. Expérimentalement, nous avons constaté que α varie très faiblement sur des séquences, grâce, en partie, à notre extracteur de marquages décrit dans la partie 2.3. Au contraire, s peut varier de façon significative d'une image à l'autre. Aussi avons-nous choisi la première approche pour l'estimation de α et la troisième approche pour l'estimation de s dans nos expériences.

4.3 sensibilité de l'estimation aux valeurs des paramètres

Les algorithmes d'estimation robuste sont, par construction, peu sensibles aux perturbations sur les données. Cependant, une autre question importante à étudier est leur robustesse à d'éventuelles imprécisions au niveau du modèle de bruit. Afin d'évaluer expérimentalement l'effet d'un modèle de bruit mal adapté sur l'estimation, nous avons suivi les étapes suivantes :

1. On collecte une série de 150 images représentant le même marquage dans la même position, mais avec différentes perturbations (la figure 9 montre quelques images de cette série).
2. La position de référence du marquage est mesurée précisément par un opérateur. Dans cette expérience, la courbe à estimer est une droite.
3. Pour chaque image de la série, l'équation de la droite est estimée pour différents couples de (α, s) . On suppose que le modèle de bruit est de la famille exponentielle lissée.
4. Pour chaque couple (α, s) , l'erreur entre l'estimée et la référence est mesurée. La figure 10 montre les résultats.

Sur la figure 10 (gauche) où l'initialisation est aléatoire, on remarque que la vallée selon s est très large lorsqu'on fixe une valeur de α non aberrante. La vallée selon α , pour s fixé est beaucoup moins large. De plus, lorsque l'initialisation est faite sur un voisinage proche de la référence, figure 10 (droite), la surface d'erreur de l'estimation par rapport à la référence est très plate lorsque $\alpha < 0$. Ce résultat montre que le choix de α et de s n'est pas très sensible. En fait, il n'est pas nécessaire d'estimer des valeurs précises de α et de s pour avoir des estimées correctes, mais plutôt d'être dans le bon intervalle pour α : la valeur de s peut alors varier dans un intervalle assez large sans que cela ne modifie sensiblement le résultat de l'estimation.

Ce fait peut se comprendre car la famille exponentielle lissée se comporte approximativement comme la famille exponentielle. Or, comme remarqué dans [3], le résultat de l'estimation est indépendant de s pour la famille exponentielle. En effet, la pondération calculée dans l'étape 2 de l'estimation est $\lambda_i = 2\alpha \frac{b_i^{2(\alpha-1)}}{s^{2(\alpha-1)}}$. Ainsi l'étape 3 n'est pas modifiée, quelle que soit la valeur de s , puisque nous pouvons simplifier $s^{2(\alpha-1)}$ à droite et à gauche de l'équation après substitution de l'expression des pondérations. Notons que les moindres carrés ont la même propriété d'invariance par rapport à s , comme remarqué au paragraphe 3.1. Avec un argument similaire, nous pouvons montrer que si la famille des lois T-Student généralisées était utilisée, l'algorithme serait invariant à β .

Pour résumer, dans notre contexte d'application, l'algorithme d'estimation robuste avec un modèle de bruit choisi dans la famille exponentielle lissée est relativement robuste au choix du paramètre s . Cependant le choix d'une valeur non aberrante de α demeure indispensable.

4.4 amélioration de la convergence

La relative robustesse de l'algorithme aux choix des paramètres α (ou β) et de s , discutée dans la partie précédente, peut être mise à profit pour accélérer la convergence de l'algorithme vers un minimum local proche du minimum global.

La figure 11 montre que le pic de la fonction de pondération utilisée par l'algorithme MCPI devient de plus en plus prononcé lorsque α diminue. Le même comportement est observable lorsque s décroît. Par conséquent, plus la valeur de α (ou de s) est faible, plus l'effet des fortes perturbations diminue et donc, plus robuste est l'estimation.

Lorsque α décroît, la surface d'erreur $e_R(A)$ devient de moins en moins lisse. Si $\alpha = 1$, c'est une parabolöide et il existe donc un minimum global unique. En diminuant la valeur de

α au delà de $\frac{1}{2}$, des minima locaux apparaissent. Cet aspect est illustré figure 12, où la surface d'erreur $e_R(A)$ est tracée pour trois valeurs différentes de α .

Comme expliqué dans [2], cette propriété peut être mise à profit pour faire converger l'algorithme vers un minimum local proche du minimum global. Cette approche, appelée non convexité graduelle (*Graduated Non Convexity* ou GNC), consiste, dans un premier temps à forcer la convexité locale de la fonction coût en choisissant une grande valeur de α . Ensuite, on effectue une série d'estimations *en continuation*, c'est-à-dire en utilisant le résultat de l'itération précédente pour initialiser l'estimation suivante, et en diminuant progressivement α . Cependant, la décroissance de α doit se faire avec précaution. En effet, lorsque α a une valeur trop faible, l'estimée se fige à la valeur courante durant les itérations comme remarqué dans [7]. Vu le grand nombre des minima locaux, une initialisation trop éloignée du minimum global peut entraîner une convergence vers un minimum local non pertinent. La borne inférieure de α ne doit donc pas être trop petite par rapport à l'estimation $\hat{\alpha}$ obtenue comme dans la partie 4.2.

La variation de s permet d'obtenir le même effet de lissage sur la fonction de coût. Donc, comme proposé dans [22], une grande valeur de s peut être utilisée de façon similaire à une grande valeur de α pour converger vers un minimum local proche du minimum et accélérer la convergence. Par contre, si s est trop petit, les minima locaux apparaissent et l'estimation peut être figée à sa valeur courante. Ainsi, pour accélérer la convergence tout en assurant la robustesse, les estimations $(\hat{\alpha}, \hat{s})$ obtenues par le principe du maximum de vraisemblance permettent de déterminer des bornes inférieures correctes pour la séquence décroissante de paramètres s et α .

5 évaluation des estimées

En pratique, lorsque qu'un algorithme d'estimation est utilisé, il est naturel de fournir le résultat accompagné d'une mesure de confiance afin d'aider le système utilisant cet algorithme à exploiter au mieux le résultat (comme par exemple un filtre de Kalman dans les applications de suivi [25]). Dans le cadre de l'estimation statistique, cette mesure de confiance est naturellement fournie par la matrice de covariance (ou son inverse). C'est pour cette raison que nous avons formulé le problème de détection de courbes comme un problème d'estimation statistique dans les parties précédentes. Sous l'hypothèse d'un bruit gaussien, le calcul de la matrice de covariance est simple et direct. Cependant, dans le cadre de l'estimation robuste, le calcul de la matrice de covariance est plus difficile. Il y a deux raisons pour cela. La première est que nous ne connaissons pas de formule analytique de cette matrice. Seules des approximations sont calculables. La deuxième raison est que le choix des paramètres définissant le modèle de bruit (α, s) s'avère plus critique pour le calcul de la matrice de covariance que dans le cas de l'estimation des paramètres.

5.1 matrices de confiance

Un état de l'art complet sur les différentes approches pour estimer la matrice de covariance dépasserait le cadre de cet article. Nous présentons donc une sélection de cinq approximations déjà connues, qui ont été choisies car elles sont rapides à calculer. Ensuite, nous proposons une nouvelle approximation dont les performances seront évaluées au paragraphe suivant.

L'approximation la plus simple de la matrice de covariance C_{Cipra} , appelée matrice de Cipra

dans [25], est :

$$C_{Cipra} = s^2 \left(\sum_{i=1}^{i=n} \lambda_i X_i X_i^t \right)^{-1}$$

où λ_i est la pondération à la suite de l'estimation robuste. Une autre approximation simple à calculer est aussi proposée dans [25] :

$$C_{Simple} = s^2 \left(\sum_{i=1}^{i=n} \lambda_i^2 X_i X_i^t \right)^{-1}$$

Remarquons que dans ces deux approximations, s^2 apparaît comme facteur multiplicatif. Il est donc nécessaire d'estimer correctement s pour obtenir une bonne approximation de la matrice.

Dans le chapitre 7 de [18], Huber propose trois approximations de matrice de covariance à partir d'une matrice de covariance asymptotique exacte :

$$C_{Huber1} = K^2 \frac{\frac{1}{n-d-1} \sum_{i=1}^{i=n} (\rho'(b_i))^2}{\left(\frac{1}{n} \sum_{i=1}^{i=n} \rho''(b_i) \right)^2} \left(\sum_{i=1}^{i=n} X_i X_i^t \right)^{-1},$$

$$C_{Huber2} = K \frac{\frac{1}{n-d-1} \sum_{i=1}^{i=n} (\rho'(b_i))^2}{\frac{1}{n} \sum_{i=1}^{i=n} \rho''(b_i)} \left(\sum_{i=1}^{i=n} \rho''(b_i) X_i X_i^t \right)^{-1},$$

$$C_{Huber3} = K^{-1} \frac{1}{n-d-1} \sum_{i=1}^{i=n} (\rho'(b_i))^2 W^{-1} \left(\sum_{i=1}^{i=n} X_i X_i^t \right) W^{-1},$$

avec la notation de Huber $\rho(b) = \phi(b^2)$ et $W = \sum_{i=1}^{i=n} \rho''(b_i) X_i X_i^t$. n est le nombre de points et $d+1$ est la dimension des vecteurs A et X_i . Le facteur de correction, K , est donné par :

$$K = 1 + (d+1) \frac{\sum_{i=1}^{i=n} (\rho''(b_i) - \frac{1}{n} \sum_{i=1}^{i=n} \rho''(b_i))^2}{\left(\sum_{i=1}^{i=n} \rho''(b_i) \right)^2}.$$

Pour simplifier les notations, nous posons $O_1 = \sum_{i=1}^{i=n} \lambda_i X_i X_i^t$ et $O_2 = \sum_{i=1}^{i=n} \lambda_i^2 X_i X_i^t$. Ces deux matrices sont symétriques. L'inverse de la matrice de covariance sera appelée la *matrice de confiance* dans la suite.

Dans nos expériences avec des données réelles, les approximations précédentes de la matrice de covariance ne fournissent pas des mesures pertinentes sur la qualité de détection des marquages, comme nous le montrons par des expérimentations dans la prochaine partie. Nous proposons donc une nouvelle approximation qui sera justifiée en annexe 8.2 :

$$C_{ITC} = \frac{\sum_{i=1}^{i=n} \lambda_i b_i^2}{\sum_{i=1}^{i=n} \lambda_i - \text{Trace}(O_2 O_1^{-1})} O_1^{-1} O_2 O_1^{-1}. \quad (14)$$

Cette approximation n'est pas une matrice de covariance asymptotique, contrairement à celles proposées par Huber. L'avantage de C_{ITC} est qu'elle ne nécessite pas un nombre n important de données. Pour l'obtenir, nous supposons que la pondération λ_i n'est pas aléatoire. En effet, λ_i est borné entre 0 et 1, il peut donc être considéré comme de variance beaucoup plus faible que b_i .

5.2 comparaison expérimentale des matrices de covariance

Les six matrices présentées dans la partie précédente sont évaluées à l'aide d'une série de 150 images d'un marquage réel (quelques exemples sont montrés figure 9). Deux étapes supplémentaires sont ajoutées à l'expérience de comparaison décrite dans la partie 4.3 :

5. Les six approximations de la matrice de covariance sont calculées pour chaque image et pour chaque couple de valeurs (α, s) .
6. Une matrice de référence est calculée à l'aide de l'ensemble des estimées des courbes pour chaque couple de valeurs (α, s) . Les erreurs relatives entre les approximations de la matrice de covariance et la référence sont calculées.

D'après la figure 13, les trois coefficients de la matrice symétrique varient de la même façon. Ainsi, sans perdre en généralité, l'étude peut-être limitée au premier coefficient diagonal.

La figure 14 illustre la variation des premières composantes des six matrices de confiance pour $\alpha = 0.2$ et $s = 2$. Ces variations sont relativement similaires, même si les ordres de grandeurs varient énormément d'une approximation à l'autre.

TAB. 2 – Erreurs relatives en pourcentage des coefficients C_{11} , C_{22} et C_{12} des six estimations de matrice de confiance par rapport à la matrice de référence, avec $(\alpha, s) = (0.2, 2)$.

| Type | err. rel. C_{11} | err. rel. C_{22} | err. rel. C_{12} |
|--------|--------------------|--------------------|--------------------|
| Cipra | 195% | 198% | 196% |
| Huber1 | 156% | 183% | 192% |
| Huber2 | 200% | 200% | 200% |
| Huber3 | 200% | 200% | 200% |
| Simple | 98% | 99% | 101% |
| ITC | 64% | 41% | 43% |

Sur le tableau 2, les erreurs relatives des six approximations de la matrice de confiance par rapport à la matrice de référence sont obtenues avec $(\alpha, s) = (0.2, 2)$, en moyennant sur les images. Nous obtenons des résultats similaires avec d'autres valeurs de (α, s) se trouvant dans la "vallée" de la figure 10 où l'estimation est correcte. L'erreur relative est calculée par : $\frac{2|C_{ij}-C_{ref,ij}|}{C_{ij}+C_{ref,ij}}$. Ainsi, 200% correspond aux plus mauvais cas où la valeur de C_{ij} est loin de la valeur de référence $C_{ref,ij}$. Ce tableau montre que seul C_{ITC} fournit le résultat le moins éloigné de la référence. Ce fait est aussi illustré par la figure 14 avec $\alpha = 0.2$ et $s = 2$ et où seulement C_{ITC} a un ordre de grandeur correct. Par conséquent, C_{ITC} semble plus robuste que les autres approximations.

Sur la figure 15, les erreurs relatives entre le premier coefficient de la matrice de confiance C_{ITC} et la matrice de référence sont données pour différentes valeurs de α et s . Pour les autres coefficients, nous pouvons tracer des cartes similaires. Elles montrent qu'une erreur relative de 20% peut être obtenue au voisinage de $s = 2.5$, pour $\alpha < 0$, ce qui n'est pas le cas avec les autres approximations. Cela témoigne de la robustesse de C_{ITC} au choix de α . Néanmoins, pour α donné, la vallée est assez étroite : au contraire de ce que nous avons remarqué pour l'estimation des paramètres des courbes, une bonne estimation de s demeure importante pour le calcul de C_{ITC} .

Dans la partie 4.2, le α moyen, estimé à partir des résidus collectés sur les détections effectuées avec les 150 images était $\hat{\alpha} = 0.05$. Puisque s n'est pas identique d'une image à l'autre, il est préférable d'estimer un s moyen, comme la moyenne de s sur toutes les images en fixant $\alpha = 0.05$. Cela donne $\hat{s} = 2.02$. Il est intéressant de noter que le modèle de bruit avec $(\hat{\alpha}, \hat{s}) = (0.05, 2.02)$ se trouve dans la vallée de la figure 15. Notre comparaison est donc cohérente.

Des tests similaires sur des données de synthèse ont été réalisés avec des courbes de plus haut degré en utilisant dans l'estimation les valeurs exactes des paramètres du modèle de bruit. Les courbes de plus haut degré nécessitent une régularisation dans le cas où le problème est sous-contraint [24]. Le test a consisté à générer 10000 ensembles de points perturbés par un bruit de Cauchy, puis à estimer un polynôme de degré 2. Les erreurs entre les approximations et la référence sont de -5% pour C_{ITC} , $+9\%$ pour C_{Simple} , et -16% pour C_{Cipra} . Les autres approximations de la matrice de covariance sont plus éloignées. Ces résultats concordent avec l'expérimentation sur les données réelles.

5.3 incertitudes sur la détection

Il est possible d'exploiter la matrice de confiance sur les paramètres de la courbe, pour visualiser dans l'image l'incertitude sur le positionnement local de celle-ci.

Plusieurs exemples de détection d'un marquage avec le modèle de polynôme fractionnaire de degré 3 sont montrés figure 17. Les mêmes paramètres du modèle de bruit sont utilisés pour toutes les images : $\alpha = -0.5$ et $s = 4$. Les marquages, proches de l'horizon, souvent trop fins, ne sont pas toujours détectés. Ainsi, la détection diverge parfois pour les grandes distances.

Sur cette figure, les courbes d'incertitude à plus ou moins un écart-type sur la position horizontale du marquage sont aussi affichées. La position horizontale de la courbe à la ligne x est donnée par $X(x)^t A$. Cette incertitude est donc calculée à partir de la variance $E[(X(x)^t A - E[X(x)^t A])^2] = X(x)^t E[(A - E[A])(A - E[A])^t] X(x)$ qui peut être approchée par $X(x)^t C_{ITC} X(x)$. Les courbes d'incertitudes obtenues sont plus ou moins proches, suivant que le marquage est plus ou moins dense. Sur un marquage continu, l'incertitude augmente avec la distance.

6 équivalence des matrices de covariance

6.1 équivalence expérimentale

Dans les expériences décrites dans les parties précédentes ou dans d'autres expériences similaires avec d'autres approximations de matrices de covariance, nous avons remarqué que les coefficients de ces matrices varient de la même manière sur les mêmes séquences. La différence vient principalement de leur ordre de grandeur, comme l'illustre la figure 14. Cette observation nous conduit à regrouper ces approximations en classe de matrices proportionnelles, et à étudier

les liens possibles entre ces classes. Dans cette partie, nous montrons expérimentalement que certaines classes d'approximations sont équivalentes. Ces propriétés d'équivalence peuvent être mises à profit pour obtenir des approximations de complexité algorithmique plus faible, ce qui peut être crucial dans les applications en temps réel.

Présentons d'abord trois classes de matrices M_0 , M_1 et M_2 , définies dans le tableau 3.

TAB. 3 – Les trois classes de matrices utilisées dans les approximations de la matrice de covariance. Ces classes sont représentées ici par des matrices normalisées.

| Définition | déduite de |
|---|--------------|
| $M_0 \propto \frac{1}{n} \sum_{i=1}^{i=n} X_i X_i^t$ | C_{Huber1} |
| $M_1 \propto \frac{\sum_{i=1}^{i=n} \lambda_i X_i X_i^t}{\sum_{i=1}^{i=n} \lambda_i}$ | C_{Cipra} |
| $M_2 \propto \frac{\sum_{i=1}^{i=n} \lambda_i^2 X_i X_i^t}{\sum_{i=1}^{i=n} \lambda_i^2}$ | C_{Simple} |

Nous définissons l'erreur relative entre deux matrices M et N avec la norme de Froebenius :

$$r(M, N) = 2 \frac{\sqrt{\text{Trace}((M - N)(M - N)^t)}}{\sqrt{\text{Trace}(MM^t)} + \sqrt{\text{Trace}(NN^t)}}$$

Comme on le voit dans le tableau 4, sous la condition de perturbations faibles dans les images, ces trois classes sont expérimentalement équivalentes dans un test effectué sur une série de 20 images (avec $\alpha = 0$ et $s = 2$, d'autres valeurs donnent des résultats équivalents). Lorsque nous introduisons des perturbations fortes, M_1 , M_2 restent encore expérimentalement équivalentes.

Une autre matrice normalisée peut être déduite à partir de C_{Huber2} : $M_3 = \frac{\sum_{i=1}^{i=n} \rho''(b_i) X_i X_i^t}{\sum_{i=1}^{i=n} \rho''(b_i)}$. Expérimentalement, M_3 n'est pas équivalente aux classes M_0 , M_1 et M_2 à cause du changement possible de signe de $\rho''(b_i)$. C'est aussi probablement la raison de la mauvaise performance des matrices C_{Huber2} et C_{Huber3} dans les tests comparatifs de la partie 5.2.

TAB. 4 – Erreurs relatives en pourcentage entre les trois classes M_0 , M_1 et M_2 définies dans le tableau 3. Les tests ont été effectués pour la partie gauche du tableau sur une série de 20 images considérées comme faiblement perturbées. Pour la partie droite du tableau, le résultat est obtenu à partir d’une série de 20 images plus fortement perturbées.

| $r(M, N)$ | M_0 | M_1 | M_2 | M_0 | M_1 | M_2 |
|-----------|--------|-------|-------|--------|--------|--------|
| M_0 | 0.0% | 0.67% | 1.52% | 0.0% | 34.66% | 32.23% |
| M_1 | 0.67% | 0.0% | 2.19% | 34.66% | 0.0% | 2.50% |
| M_2 | 0.152% | 2.19% | 0.0% | 32.23% | 2.50% | 0.0% |

6.2 réduction du coût de calcul

L’équivalence expérimentale entre les classes de matrices M_1 et M_2 peut être exploitée dans le contexte des applications en temps réel, afin de proposer des approximations plus rapides à calculer que la matrice C_{ITC} . En effet, le calcul de C_{ITC} (ou de son inverse) nécessite une inversion de matrice, ce qui peut être évité en utilisant les équivalences. Grâce à l’équivalence $M_1 \sim M_2$, nous obtenons :

$$O_2 \sim \frac{\sum_{i=1}^{i=n} \lambda_i^2}{\sum_{i=1}^{i=n} \lambda_i} O_1$$

Donc, $O_1^{-1} O_2 O_1^{-1} \sim \frac{\sum_{i=1}^{i=n} \lambda_i^2}{\sum_{i=1}^{i=n} \lambda_i} O_1^{-1}$ et $O_2 O_1^{-1} \sim \frac{\sum_{i=1}^{i=n} \lambda_i^2}{\sum_{i=1}^{i=n} \lambda_i} I_d$. Ces deux équations dans (14) amènent une nouvelle approximation de matrice de covariance pour l’estimation robuste :

$$C_{ITCA_{approx1}} = \frac{(\sum_{i=1}^{i=n} \lambda_i b_i^2)(\sum_{i=1}^{i=n} \lambda_i^2)}{(\sum_{i=1}^{i=n} \lambda_i)^2 - d \sum_{i=1}^{i=n} \lambda_i^2} O_1^{-1} \quad (15)$$

Nous avons fait la même expérience que celle citée dans la partie 5.2, en utilisant $C_{ITCA_{approx1}}$. La surface d’erreur relative de $C_{ITCA_{approx1}}$ en fonction de (α, s) est montrée à la figure 16. Nous pouvons la comparer à la surface d’erreur de C_{ITC} de la figure 15 dont elle est assez proche, ce

qui confirme que l'approximation n'est pas trop grossière.

Dans (15), le terme négatif au dénominateur prend généralement des valeurs faibles. Il est donc possible de le supprimer, ce qui évite le risque d'avoir un dénominateur négatif. Ainsi nous introduisons finalement une nouvelle approximation :

$$C_{ITCA_{approx2}} = \frac{\sum_{i=1}^{i=n} \lambda_i b_i^2 \sum_{i=1}^{i=n} \lambda_i^2}{(\sum_{i=1}^{i=n} \lambda_i)^2} O_1^{-1} \quad (16)$$

Expérimentalement, cette matrice présente les mêmes caractéristiques intéressantes que C_{ITC} .

7 conclusion

Nous avons présenté un système de détection et de suivi des marquages routiers par caméra. L'extraction locale des marquages tient compte de la géométrie des bandes blanches, en particulier de leur largeur. Ce critère géométrique nous permet de mieux détecter les points des marquages, tout en vérifiant l'hypothèse de détection conservative. Un détecteur est dit conservatif s'il élimine beaucoup de données parasites, tout en conservant les données informatives même si elles sont très bruitées. Les données extraites sont introduites dans un algorithme d'estimation robuste des courbes paramétriques qui caractérisent la forme de la route. Nous avons montré que les multiplicateurs de Lagrange permettent de revisiter la théorie semi-quadratique et de justifier l'utilisation de l'algorithme des moindres carrés pondérés itératifs dans la théorie des M-estimateurs, même pour des potentiels $\rho(b) = \phi(b^2)$ non convexes et donc plus robustes.

Dans ce contexte, nous avons traité la question importante qu'est le choix des paramètres du modèle de bruit. Nous avons introduit des familles de modèles de bruit à deux paramètres, qui permettent une transition continue entre le modèle gaussien et des modèles non-gaussiens. Puis, nous avons montré comment cette propriété de continuité peut être utilisée pour améliorer

la précision et la robustesse des estimées.

L'avantage de poser la détection comme un problème d'estimation est que la matrice de covariance associée fournit une évaluation naturelle des résultats de détection. A cause des non-linéarités, notamment, le calcul de la matrice de covariance est difficile dans le cas robuste et il faut recourir à des approximations. Par ailleurs, si les estimateurs donnent souvent des valeurs suffisamment pertinentes malgré le choix d'un modèle de bruit mal adapté, celui-ci a un impact nettement plus fort sur la qualité de la matrice de covariance obtenue. Comme la recherche d'un modèle de bruit précis se révèle difficile en pratique, nous avons proposé une nouvelle approximation C_{ITC} de la matrice de covariance, qui s'est avérée plus robuste, dans les tests présentés dans cet article. Elle peut être utilisée dans d'autres applications et d'autres contextes avec probablement les mêmes avantages. Finalement, certaines équivalences entre matrices de covariance approchées ont été établies expérimentalement. Elles nous ont permis de déduire de nouvelles expressions de matrices approchées, moins coûteuses à calculer.

8 annexes

8.1 optimisation sous contraintes et algorithme MCPI

Pour prouver la convergence locale de l'algorithme MCPI décrit en partie 3.3, nous allons étudier la minimisation de $E(A, W)$ par rapport à W sous les n conditions $h_i(A, W) \leq 0$, pour tout A . Ce problème est bien posé car il s'agit de la minimisation d'une fonction convexe sous des contraintes convexes. En appliquant le théorème classique de Kuhn et Tucker [13], si une solution existe, la minimisation de $E(A, W)$ par rapport à W est équivalente à la recherche de

l'unique point selle de la fonction lagrangienne du problème :

$$L_R(A, W, \lambda_i) = \frac{1}{2} \sum_{i=1}^{i=n} -\phi(w_i) + \lambda_i (w_i - (\frac{X_i^t A - y_i}{s}))^2$$

où les λ_i sont les multiplicateurs de Lagrange ($\lambda_i \geq 0$). Formellement, nous montrons que pour tout A :

$$-e_R(A) = \min_{w_i} \max_{\lambda_i} L_R(A, W, \lambda_i) \quad (17)$$

Remarquons que la fonction lagrangienne L_R est maintenant quadratique par rapport à A , au contraire de la fonction d'erreur initiale e_R . En utilisant la propriété du point selle, nous pouvons changer l'ordre de minimisation et de maximisation dans (17). $L_R(A, W, \lambda_i)$ étant convexe par rapport à W , le minimum de $L_R(A, W, \lambda_i)$ par rapport à W est le zéro de sa dérivée. Ainsi, nous déduisons :

$$\lambda_i = \phi'(w_i) \quad (18)$$

D'après cette équation et (H2), nous pouvons dire que le problème initial est équivalent à la minimisation suivante :

$$\min_A e_R(A) = \min_{A, \lambda_i} -L_R(A, \phi'^{-1}(\lambda_i), \lambda_i)$$

Soit $\mathcal{E}(A, \lambda_i) = -L_R(A, \phi'^{-1}(\lambda_i), \lambda_i)$ la fonction duale. Elle est convexe par rapport à A ainsi que par rapport à chaque λ_i (en effet, $\frac{\partial^2 \mathcal{E}}{\partial \lambda_i^2} = -\frac{1}{\phi''(\phi'^{-1}(\lambda_i))}$).

L'algorithme robuste le plus souvent utilisé dans les approches semi-quadratique et les M-estimateurs consiste à minimiser \mathcal{E} de façon alternée par rapport à A puis aux λ_i . On peut alors montrer, par exemple à partir de [20] que cet algorithme, comme tous les algorithmes de minimisation alternée, fait décroître strictement la fonction duale, si sa position courante n'est pas sur un point stationnaire de la fonction duale. Un point stationnaire correspond à des valeurs des paramètres A et λ_i pour lesquelles la dérivée de $\mathcal{E}(A, \lambda_i)$ est nulle.

Le problème des points stationnaires est facile à résoudre en pratique lorsqu'ils sont isolés. En effet, il suffit de vérifier la positivité de la matrice hessienne de $\mathcal{E}(A, \lambda_i)$, après convergence. Tant que la matrice hessienne n'est pas positive, on introduit une légère perturbation aléatoire de la position courante pour continuer à minimiser la fonction duale. Avec cette modification, la minimisation de la fonction duale est globalement convergente.

Du fait de la formulation lagrangienne, si les points stationnaires sont isolés pour la fonction duale, ceci prouve que l'algorithme est globalement convergent vers un minimum local de $e_R(A)$, quelle que soit l'initialisation A_0 . Un résultat similaire a été démontré dans [21, 8], par une approche qui porte sur les points stationnaires de $e_R(A)$ et non de la fonction duale.

Avec la formulation lagrangienne, l'unicité de la solution peut aussi être démontrée lorsque \mathcal{E} est convexe par rapport au couple (A, λ_i) .

8.2 justification de l'approximation C_{ITC}

Nous justifions ici l'approximation (14) introduite dans la partie 5.1. La matrice de covariance du vecteur des paramètres estimé A est définie comme l'espérance mathématique $E[(A - \tilde{A})(A - \tilde{A})^t]$, où \tilde{A} est la valeur vraie du vecteur de paramètres. En effet, l'estimateur est centré comme nous le montrerons ci-dessous.

L'étape 3 de l'algorithme robuste peut s'écrire sous la forme : $O_1 A = B$ avec $O_1 = \sum_{i=1}^{i=n} \lambda_i X_i X_i^t$ et $B = \sum_{i=1}^{i=n} \lambda_i y_i X_i$. Pour les vrais paramètres, on peut écrire une équation de la même forme, $\tilde{O}_1 \tilde{A} = \tilde{B}$, avec des définitions similaires pour \tilde{O}_1 et \tilde{B} , mais impliquant les observations non bruitées, \tilde{y}_i , et les valeurs vraies des pondérations. Dans la suite, nous supposons que ces dernières, les λ_i , ne sont pas aléatoires. La conséquence immédiate est que $\tilde{O}_1 = O_1$ et donc : $(A - \tilde{A}) = O_1^{-1}(B - \tilde{B})$. La matrice de covariance de l'estimée peut ainsi être approchée

par : $O_1^{-1}E[(B - \tilde{B})(B - \tilde{B})^t]O_1^{-1}$. La seconde conséquence est que \tilde{B} s'écrit : $\tilde{B} = \sum_{i=1}^{i=n} \lambda_i \tilde{y}_i X_i$. On a donc : $(B - \tilde{B}) = \sum_{i=1}^{i=n} \lambda_i (y_i - \tilde{y}_i) X_i = \sum_{i=1}^{i=n} \lambda_i b_i X_i$. Comme le bruit est à moyenne nulle, on a $E[A - \tilde{A}] = 0$: l'estimateur est donc bien centré. Comme, de plus, les b_i sont des variables aléatoires indépendantes, on a $E[(B - \tilde{B})(B - \tilde{B})^t] = \sum_{i=1}^{i=n} \lambda_i^2 E[b_i^2] X_i X_i^t = E[b^2] O_2$. Ces deux résultats une fois rassemblés, on obtient une première expression de la matrice de covariance approchée :

$$C_{ITC} = E[b^2] O_1^{-1} O_2 O_1^{-1}. \quad (19)$$

Pour pouvoir la mettre en œuvre, il est indispensable d'obtenir une approximation de la variance du bruit $E[b^2]$. L'approche la plus simple consisterait à calculer cette variance à partir de la définition du modèle de bruit. Mais l'estimation obtenue peut être mauvaise si le modèle de bruit est trop éloigné de la perturbation réelle. Il est possible d'obtenir une approximation avec un meilleur comportement lorsque le modèle de bruit n'est pas correct, en utilisant les résidus b_i et (12).

Pour cela, étudions la relation entre $E[b^2]$ et $E[\hat{s}^2]$, en gardant l'hypothèse que les λ_i ne sont pas aléatoires. L'expression (12) peut être écrite sous la forme : $\hat{s}^2 = \frac{1}{n} \sum_{i=1}^{i=n} \lambda_i (y_i - X_i^t A)^2$. En introduisant le vrai vecteur des paramètres \tilde{A} dans la somme, nous obtenons :

$$\hat{s}^2 = \frac{1}{n} \sum_{i=1}^{i=n} \lambda_i (b_i + X_i^t (\tilde{A} - A))^2 \quad (20)$$

Pour calculer $E[\hat{s}^2]$ comme une fonction de $E[b^2]$, nous développons les termes carrés de l'équation précédente. En utilisant (19), nous obtenons :

$$E[\hat{s}^2] = \frac{1}{n} \sum_{i=1}^{i=n} (E[b^2] \lambda_i + E[b^2] \lambda_i X_i^t O_1^{-1} O_2 O_1^{-1} X_i + 2 \lambda_i X_i^t E[(\tilde{A} - A) b_i]) \quad (21)$$

Puisque b_i est centré, nous avons dans (21) $E[(\tilde{A} - A) b_i] = -E[A b_i]$. Le vecteur des paramètres estimé est $A = O_1^{-1} \sum_{k=1}^{k=n} \lambda_k X_k y_k = O_1^{-1} \sum_{k=1}^{k=n} \lambda_k X_k (X_k^t \tilde{A} + b_k)$. Ainsi, $E[A b_i]$ peut s'écrire

sous la forme $E[Ab_i] = O_1^{-1} \sum_{k=1}^{k=n} \lambda_k X_k E[b_k b_i]$. L'indépendance de b_i nous permet de simplifier la dernière équation en $E[Ab_i] = O_1^{-1} \lambda_i X_i E[b^2]$. L'expression (21) s'écrit maintenant :

$$E[\hat{s}^2] = \frac{E[b^2]}{n} (\sum_{i=1}^{i=n} \lambda_i + \sum_{i=1}^{i=n} \lambda_i X_i^t O_1^{-1} O_2 O_1^{-1} X_i - 2 \sum_{i=1}^{i=n} \lambda_i^2 X_i^t O_1^{-1} X_i) \quad (22)$$

Le troisième terme de cette équation peut se réécrire de la façon suivante : $Trace(\sum_{i=1}^{i=n} \lambda_i^2 X_i^t O_1^{-1} X_i) = \sum_{i=1}^{i=n} Trace(\lambda_i^2 X_i^t O_1^{-1} X_i)$, soit $\sum_{i=1}^{i=n} Trace(\lambda_i^2 X_i X_i^t O_1^{-1}) = Trace(O_2 O_1^{-1})$ en utilisant $Trace(AB) = Trace(BA)$. Par un raisonnement similaire, on montre facilement que le second terme de (22) s'écrit également $Trace(O_2 O_1^{-1})$ et on obtient ainsi le résultat simple $E[\hat{s}^2] = \frac{E[b^2]}{n} (\sum_{i=1}^{i=n} \lambda_i - Trace(O_2 O_1^{-1}))$. En approchant $E[\hat{s}^2]$ par \hat{s}^2 , et en utilisant (20) avec $A = \tilde{A}$, nous déduisons l'estimation de $E[b^2]$ suivante :

$$E[b^2] = \frac{\sum_{i=1}^{i=n} \lambda_i b_i^2}{\sum_{i=1}^{i=n} \lambda_i - Trace(O_2 O_1^{-1})} \quad (23)$$

Enfin, en substituant (23) dans (19), nous obtenons l'approximation de la matrice de covariance proposée (14).

Remerciements

Ce travail est soutenu par une bourse de thèse co-financée par l'INRETS et la région de l'île de France. Nous souhaitons remercier l'équipe expérimentation du LIVIC et plus particulièrement Benoît Lusetti, pour son aide sur la piste d'essai.

Références

- [1] R. AUFRÈRE, F. MARMOITON, R. CHAPUIS, F. COLLANGE et J.-P. DÉRUTIN : Détection de route de suivi de véhicules par vision pour l'ACC. *Traitement du Signal*, 17(3), 2000.

- [2] A. BLAKE et A. ZISSERMAN : *Visual Reconstruction*. MIT Press, Cambridge, MA, 1987.
- [3] C. BOUMAN et K. SAUER : A generalized Gaussian image model for edge-preserving map estimation. *IEEE Trans. on Image Processing*, 2(3):296–310, 1993.
- [4] A. BROGGI : Parallel and local feature-extraction : A real-time approach to road boundary detection. *IEEE Transactions on Image Processing*, 4(2):217–223, février 1995.
- [5] P. CHARBONNIER, L. BLANC-FÉRAUD, G. AUBERT et M. BARLAUD : Deterministic edge-preserving regularization in computed imaging. *IEEE Transactions on Image Processing*, 6(2):298–311, 1997.
- [6] T. CIPRA et R. ROMERA : Robust Kalman filtering and its application in time series analysis. *Kybernetika*, 27(6):481–494, 1991.
- [7] R. DAHYOT : *Analyse d'images séquentielles de scènes routières par modèles d'apparence pour la gestion du réseau routier*. Thèse de doctorat, Université de Strasbourg I, novembre 2001.
- [8] A. H. DELANEY et Y. BRESLER : Globally convergent edge-preserving regularized reconstruction : An application to limited-angle tomography. *IEEE Transactions on Image Processing*, 7(2):204–221, février 1998.
- [9] E. D. DICKMANNNS et A. ZAPP : A curvature-based scheme for improving road vehicle guidance by computer vision. *In Proceedings of SPIE Conference on Mobile Robots*, volume 727, pages 161–168, octobre 1986.
- [10] F. FLEURET et D. GEMAN : Coarse-to-fine visual selection. *International Journal of Computer Vision*, 41(1/2):85–107, janvier 2001.

- [11] D. GEMAN et G. REYNOLDS : Constrained restoration and the recovery of discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(3):367–383, 1992.
- [12] S. GEMAN et D. MCCLURE : Statistical methods for tomographic image reconstruction. *Bull. Int. Stat. Inst.*, LII(4):5–21, 1987.
- [13] M. GONDRAN et M. MINOUX : *Graphes et algorithmes*. 3 ed. Eyrolles, Paris, 1995.
- [14] F. GUICHARD et J.-P. TAREL : Curve extraction combining perceptual grouping and a Kalman like fitting. *In IEEE International Conference on Computer Vision (ICCV'99)*, pages 1003–1008, Kerkyra, Grèce, 1999.
- [15] F.R. HAMPEL, P.J. ROUSSEEUW, E.M. RONCHETTI et W.A. STAHEL : *Robust Statistics*. John Wiley and Sons, New York, NY, USA, 1986.
- [16] D. HASLER, L. SBAIZ, S. SÜSTRUNK et M. VETTERLI : Outlier modeling in image matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(3):301–315, mars 2003.
- [17] J. HUANG et D. MUMFORD : Statistics of natural images and models. *In Proceedings of IEEE conference on Computer Vision and Pattern Recognition (CVPR'99)*, pages 1541–1547, Ft. Collins, CO, USA, 23-25 juin 1999.
- [18] P. J. HUBER : *Robust Statistics*. John Wiley and Sons, New York, NY, USA, 1981.
- [19] S.-S. IENG, J.-P. TAREL et R. LABAYRADE : On the design of a single lane-markings detector regardless the on-board camera's position. *In Proceedings of IEEE Intelligent Vehicle Symposium (IV'2003)*, pages 564–569, Columbus, OH, USA, 9-11 juin 2003.
- [20] D. G. LUENBERGER : *Introduction to linear and nonlinear programming*. Addison Wesley, 1973.

- [21] J-C. NOSMAS : Remarques sur un algorithme d'optimisation pour une classe de fonctionnelles sur R^k . *C. R. Acad. Sci. Paris*, t.328, Série I,(Analyse numérique):1237–1240, 1999.
- [22] J. M. ODOBEZ et P. BOUTHEMY : Robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation*, 6(4):348–365, 1995.
- [23] A. SRIVASTAVA, A. LEE, E. SIMONCELLI et S. ZHU : On advances in statistical modeling of natural images. *Journal of Mathematical Imaging and Vision*, 18(1):17–33, 2003.
- [24] J.-P. TAREL et F. GUICHARD : Dynamic tracking of lane markings and boundaries in curved road. *In IEEE International Conference on Image Processing (ICIP'2000)*, volume I, pages 216–219, Vancouver, Canada, 10-13 septembre 2000.
- [25] J.-P. TAREL, S.-S. IENG et P. CHARBONNIER : Using robust estimation algorithms for tracking explicit curves. *In European Conference on Computer Vision (ECCV'02)*, volume 1, pages 492–507, Copenhagen, Danemark, 28-31 mai 2002.
- [26] C. THORPE, M. HERBERT, S. SHAFER et T. KANADE : Vision and navigation for the Carnegie-Mellon navlab. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3):361–372, mai 1988.



FIG. 1 – Première ligne : la caméra est positionnée vers l'avant du véhicule prototype du LIVIC et un exemple d'image prise par la caméra dans cette configuration est présenté. Deuxième ligne : la caméra est positionnée verticalement sur le côté du véhicule prototype et un exemple d'image prise par cette caméra est montré. Les pixels sont repérés par les coordonnées (x,y) comme le montre la seconde image de la première ligne.

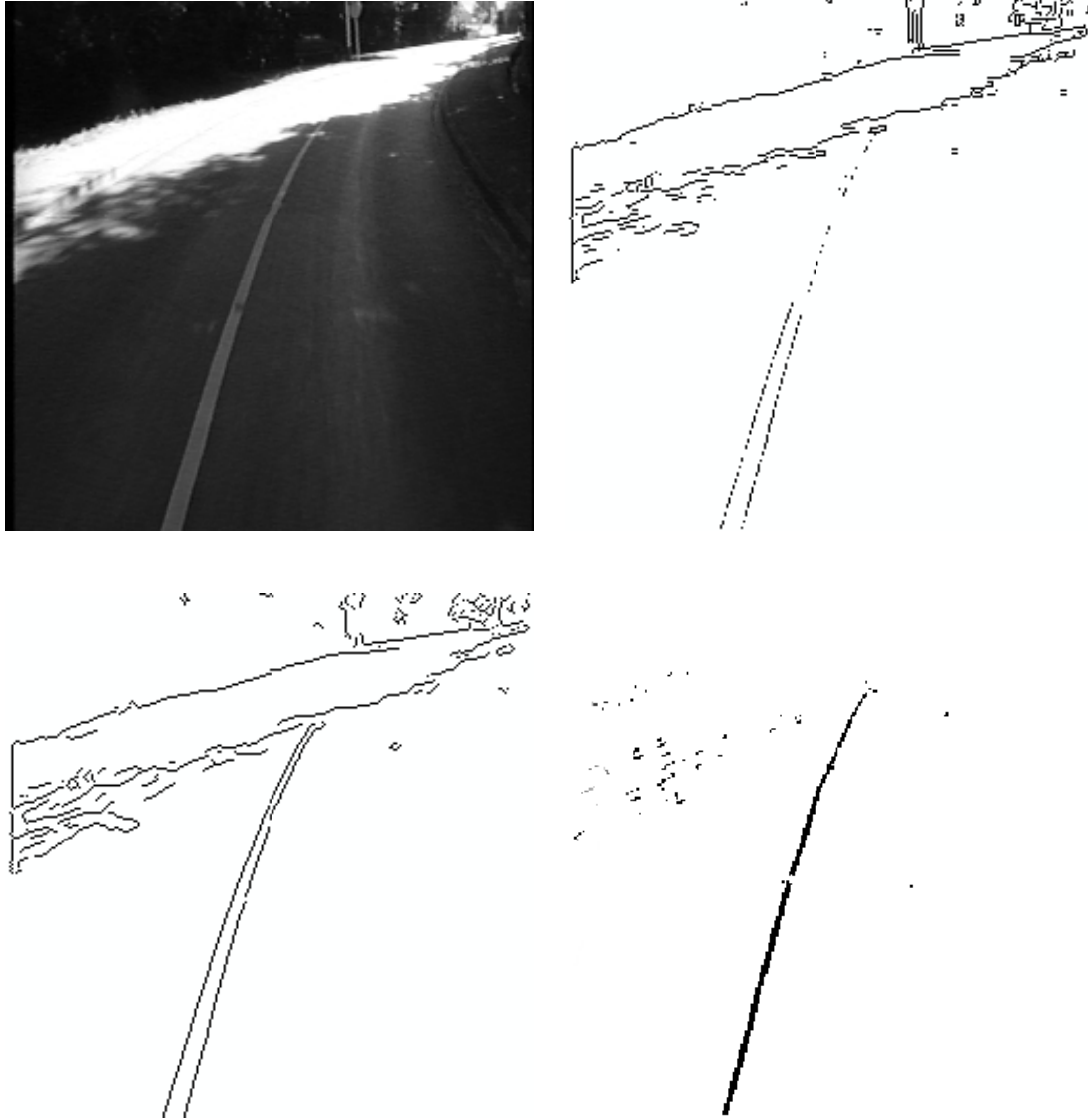


FIG. 2 – Haut-Gauche : une image de route avant traitement. Haut-Droite : résultat obtenu par le filtre de Prewitt. Bas-Gauche : résultat obtenu par le filtre de Canny. Bas-Droite : résultat obtenu avec notre algorithme d'extraction de marquage.

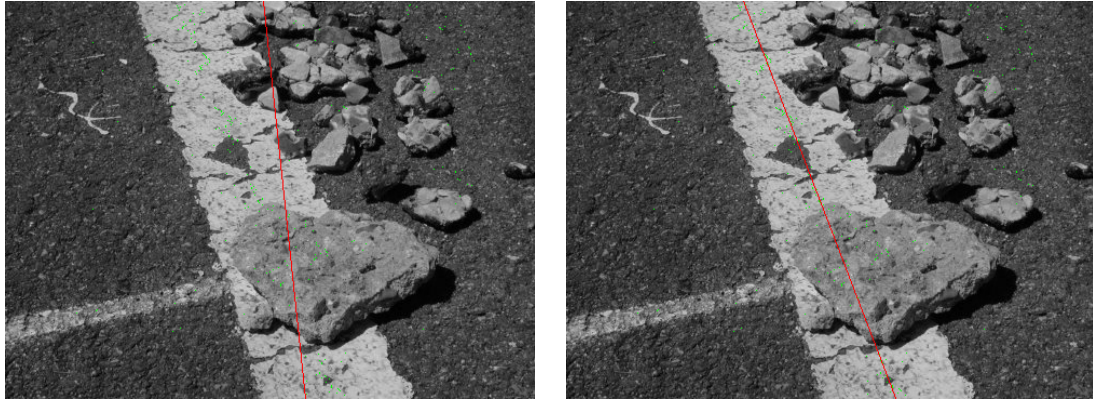


FIG. 3 – Gauche : estimation de la position du marquage en supposant que le bruit est gaussien.

Droite : estimation de la position du marquage en supposant que le bruit est T-Student.

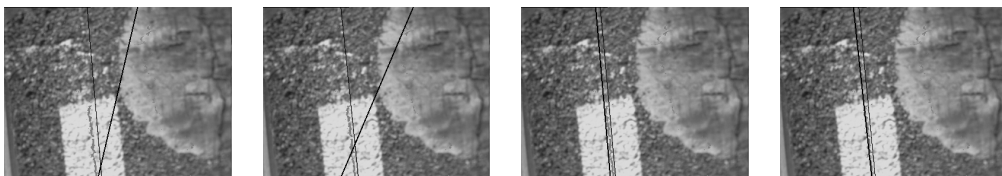


FIG. 4 – Estimation du centre des marquages à partir d'images réelles avec les hypothèses de distribution du bruit haussienne, Laplace lissée, T-Student généralisée et Geman & McClure, dans l'ordre. Les lignes fines sont les initialisations. Les lignes plus épaisses sont les estimations. Ces fonctions de densité de probabilité sont définies dans la partie 4.1.

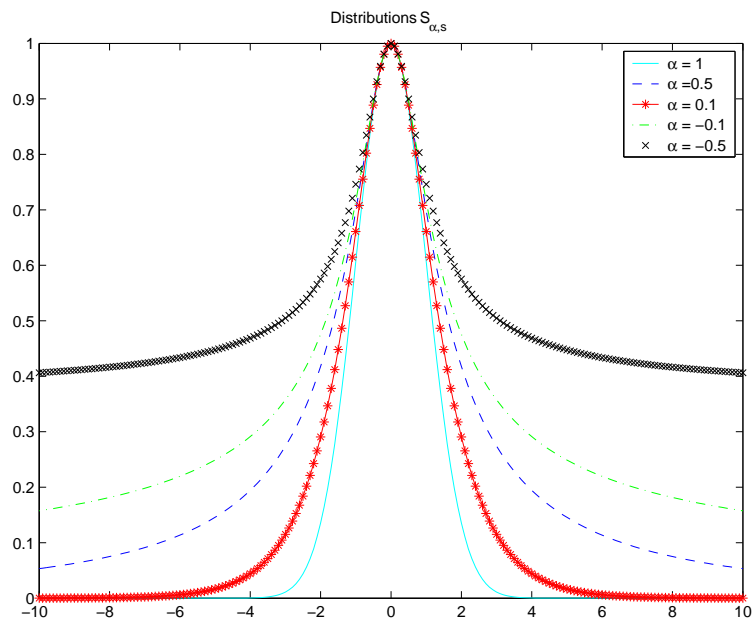


FIG. 5 – Modèles de bruit de la famille exponentielle lissée $S_{\alpha,s}$. Lorsque α décroît, les distributions décroissent de plus en plus lentement pour les grands résidus.

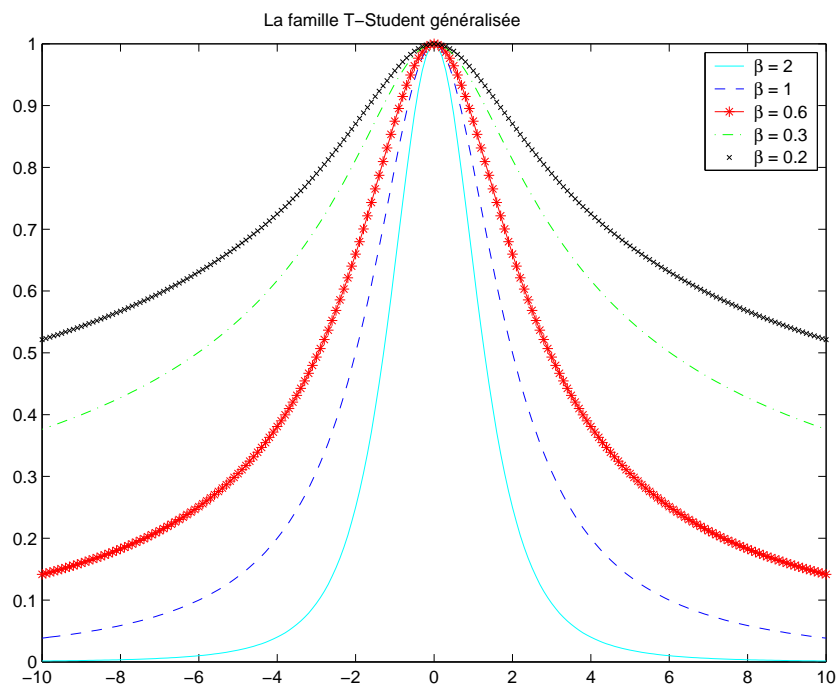


FIG. 6 – Modèles de bruit de la famille des lois T-Student généralisées $T_{\beta,s}$. Lorsque β décroît vers 0, les distributions décroissent de plus en plus lentement pour les grands résidus.

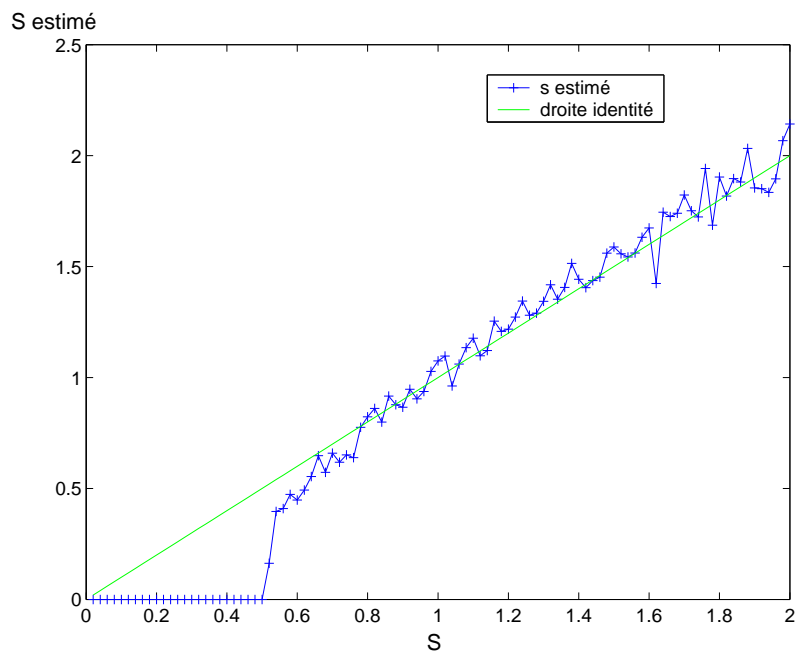


FIG. 7 – Comparaison de l’estimateur du facteur d’échelle \hat{s} et du facteur d’échelle théorique s pour un bruit de Cauchy. Nous montrons que s est sous-estimé par l’estimateur du maximum de vraisemblance lorsque $s < 1$ à cause de la discrétisation des données.

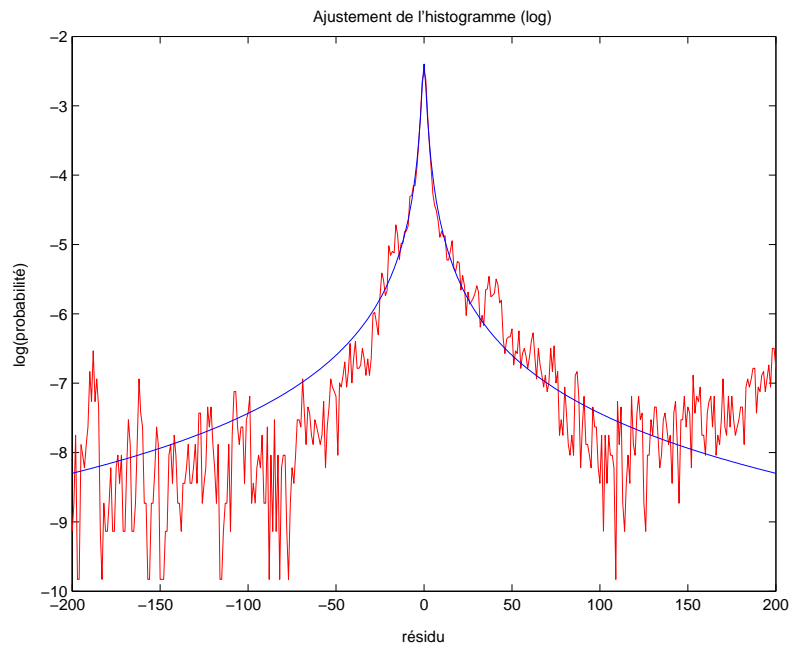


FIG. 8 – Log-histogramme de l’erreur résiduelle des détections sur 150 images après l’estimation de courbe. Avec l’approche du maximum de vraisemblance, la distribution est correctement approchée par un modèle de la famille exponentielle lissée (9) avec les paramètres $\alpha = 0.05$ et $s = 1.1$.

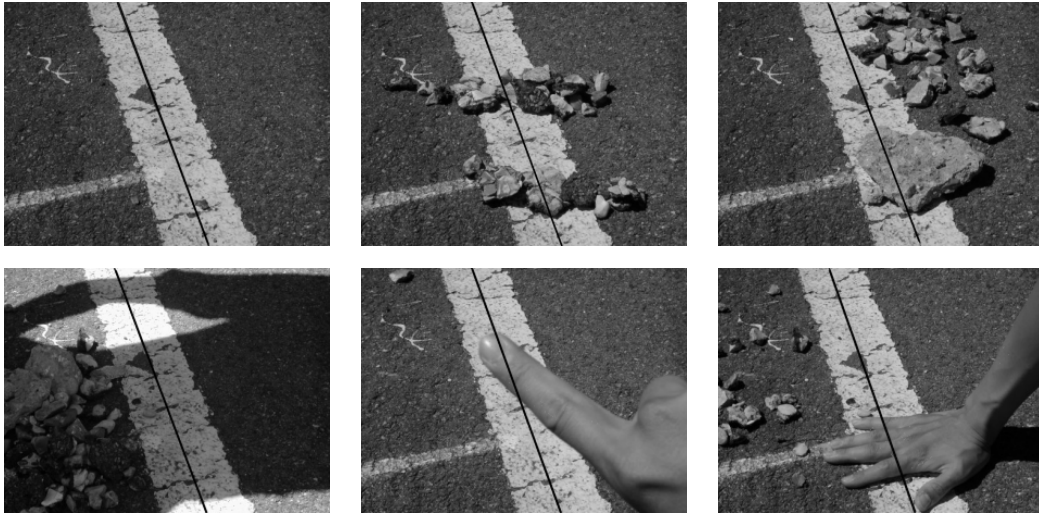


FIG. 9 – Exemples d’images extraites de la série de 150 images utilisées pour les expériences. La ligne noire correspond à la ligne de référence.

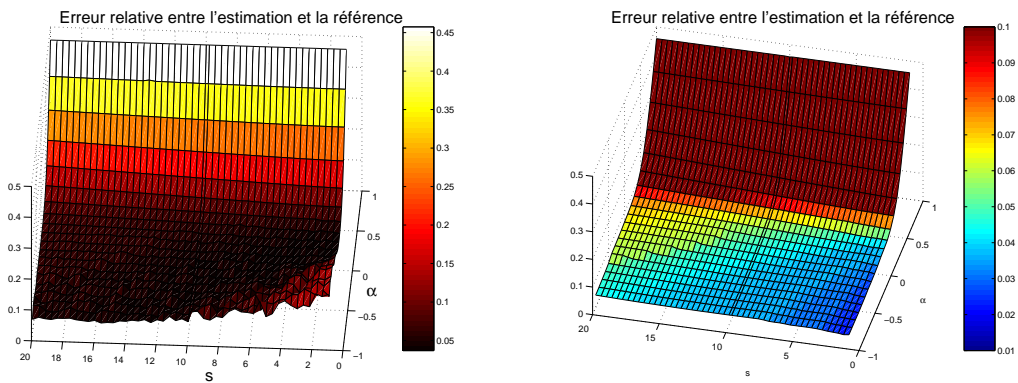


FIG. 10 – Erreur relative entre l’estimation et la courbe de référence pour différentes valeurs de α et de s . A droite, nous initialisons l’algorithme d’estimation de façon aléatoire. A gauche, A_0 est initialisé au voisinage de la référence. Lorsque α est bien choisi, la vallée suivant s est très large.

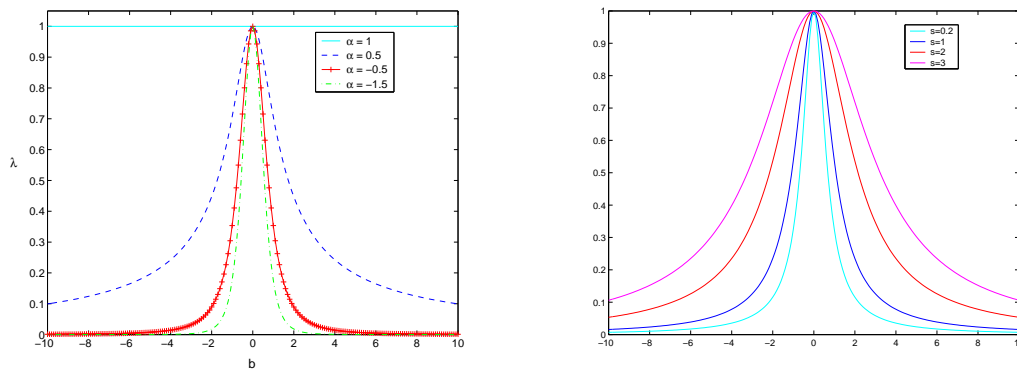


FIG. 11 – Variation de la pondération λ en fonction des résidus b , pour $s = 2.4$ et différentes valeurs de α à droite, et pour $\alpha = 0.1$ et différentes valeurs de s à gauche.

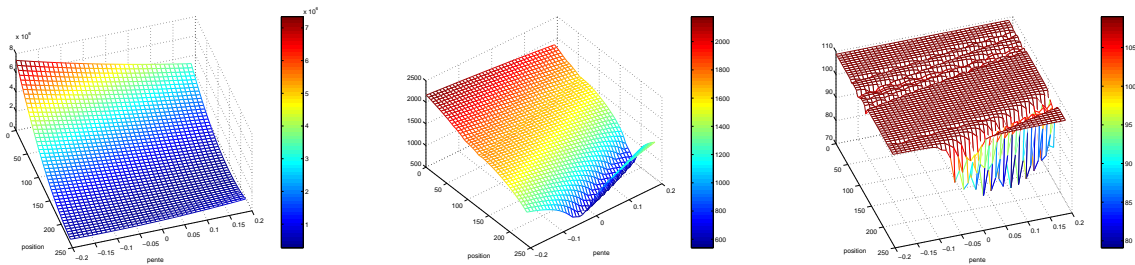


FIG. 12 – Surfaces d’erreur $e_R(A)$, en fonction des deux paramètres décrivant la droite pour différentes valeurs de α : bruit gaussien ($\alpha = 1$), modèle proche du bruit T-student ($\alpha = 0.1$), bruit Geman & McClure ($\alpha = -1$), de gauche à droite.

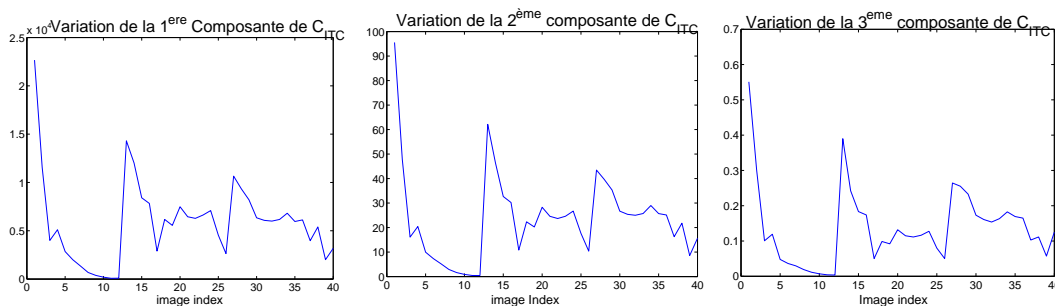


FIG. 13 – Courbes des trois coefficients de la matrice 2×2 symétrique C_{ITC} en fonction de l’indice des 40 premières images de la séquence de 150 images traitées. La première et la dernière courbe sont associées aux coefficients diagonaux de la matrice, qui est symétrique.

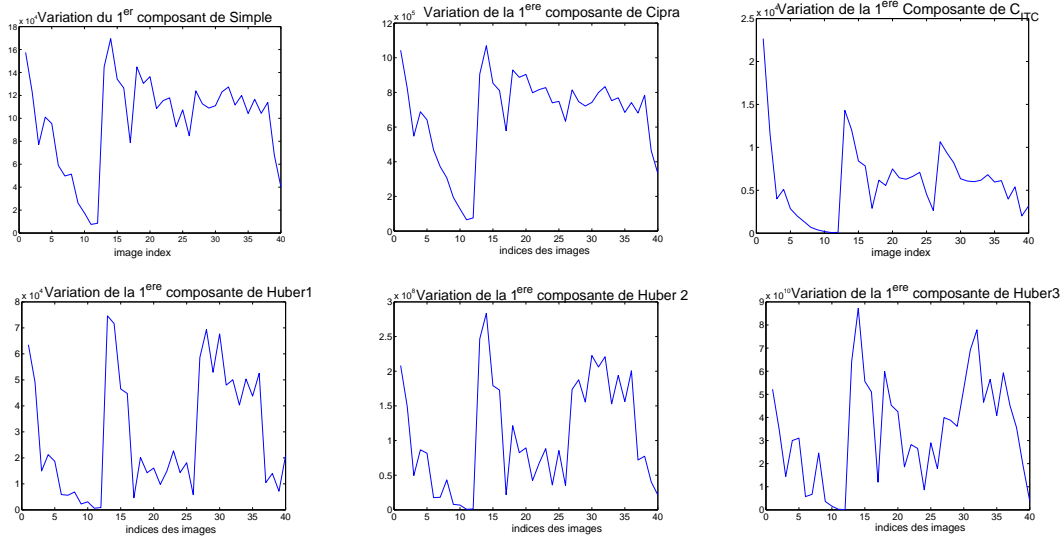


FIG. 14 – Premier coefficient des six matrices de confiance : C_{Simple} , C_{Cipra} , C_{ITC} (ligne du haut, de gauche à droite), C_{Huber1} , C_{Huber2} et C_{Huber3} (ligne du bas, de gauche à droite). Les valeurs ont des ordres de grandeur très différents, mais les courbes varient de façon analogue. Ces résultats sont obtenus avec $\alpha = 0.2$ et $s = 2$.

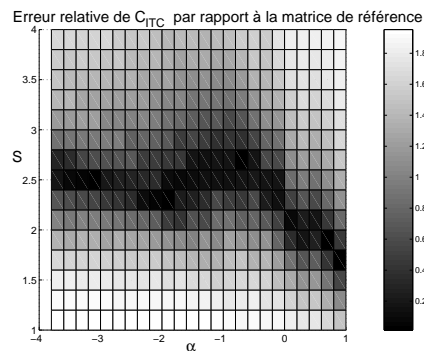


FIG. 15 – Erreur relative de C_{ITC} par rapport à la matrice de référence en fonction de (α, s) . Avec la nouvelle approximation nous obtenons une erreur relative de 20% au contraire des cinq autres qui fournissent des erreurs plus importantes, pour $\alpha < 0$.

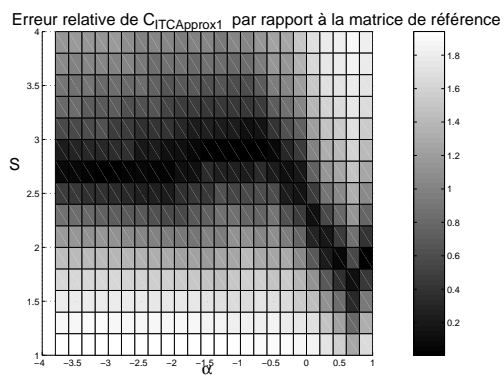


FIG. 16 – Surface d’erreur relative pour $C_{ITCAprox1}$, approximation de C_{ITC} , en fonction de (α, s) . La surface d’erreur de $C_{ITCAprox1}$ est semblable à celle de C_{ITC} , illustrée à la figure 15.



FIG. 17 – Quelques exemples de détection des marquages avec le modèle de polynôme fractionnaire de degré 3, et avec le même modèle de bruit $\alpha = -0.5$ et $s = 4$. Les courbes d'incertitude à plus ou moins un écart-type, utilisant C_{ITC} , sont aussi affichées.