

# ANR Report - Registration error estimation framework

Guillaume POTIER

09 octobre 2019

## Introduction

Le problème de recalage entre deux images consiste à trouver une application permettant de déformer une image source pour la superposer au mieux sur une image cible. La transformation ainsi trouvée n'est qu'une approximation de la déformation réelle. Par conséquent il existe une différence de position entre l'image source transformée et l'image cible. Cette différence de position est appelée erreur de recalage. Nous nous intéressons particulièrement à cette erreur de recalage et nous souhaitons pouvoir l'estimer dans le cas où les transformations appliquées suivent différents modèles d'hypothèses. Nous allons décrire dans une première partie comment estimer l'erreur de recalage dans le cas où le modèle de transformation est supposé rigide. Dans une seconde partie nous allons montrer comment le problème de recalage peut être assimilé à un problème de régression linéaire, puis comment utiliser les résultats connus sur la régression linéaire pour estimer l'erreur de recalage dans le cas où le modèle de transformation est étendu au modèle affine. Enfin nous allons étudier grâce à plusieurs simulations la différence d'erreur induite par l'utilisation de modèle rigide ou affine.

## Modèle affine

### Le modèle de régression linéaire multivariée

Selon l'ouvrage *Applied multivariate statistical analysis sixth edition* nous formulons le modèle de regression linéaire multivarié suivant :

Soit  $m$  réponses  $y_1, y_2, \dots, y_m$  et un set de  $r$  variables prédictrices  $z_1, z_2, \dots, z_r$ . Nous supposons que chaque réponse suit un modèle de régression linéaire de sorte que :

$$\left\{ \begin{array}{l} y_1 = \beta_{01} + \beta_{11}z_1 + \dots + \beta_{r1}z_r + \epsilon_1 \\ y_2 = \beta_{02} + \beta_{12}z_1 + \dots + \beta_{r2}z_r + \epsilon_2 \\ \vdots \\ y_m = \beta_{0m} + \beta_{1m}z_1 + \dots + \beta_{rm}z_r + \epsilon_m \end{array} \right.$$

Le vecteur réponse pour la  $i^{\text{ème}}$  observation peut être écrit sous la forme suivante :

$$\begin{bmatrix} y_{i1} \\ y_{i2} \\ \vdots \\ y_{im} \end{bmatrix} = \begin{bmatrix} \beta_{01} \\ \beta_{02} \\ \vdots \\ \beta_{0m} \end{bmatrix} + \begin{bmatrix} \beta_{11} & \beta_{21} & \dots & \beta_{r1} \\ \beta_{12} & \beta_{22} & \dots & \beta_{r2} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{1m} & \beta_{2m} & \dots & \beta_{rm} \end{bmatrix} \times \begin{bmatrix} z_{i1} \\ z_{i2} \\ \vdots \\ z_{ir} \end{bmatrix} + \begin{bmatrix} \epsilon_{i1} \\ \epsilon_{i2} \\ \vdots \\ \epsilon_{im} \end{bmatrix}$$

En notation matricielle pour  $n$  observations composées de  $m$  variables nous avons :

$$\underset{(n \times m)}{Y} = \underset{(n \times (r+1))}{Z} \times \underset{((r+1) \times m)}{\beta} + \underset{(n \times m)}{\epsilon}$$

$$\underset{(n \times m)}{Y} = \begin{bmatrix} y_{11} & y_{12} & \dots & y_{1m} \\ y_{21} & y_{22} & \dots & y_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ y_{n1} & y_{n2} & \dots & y_{nm} \end{bmatrix}$$

$$\underset{(n \times (r+1))}{Z} = \begin{bmatrix} 1 & z_{11} & \dots & z_{1r} \\ 1 & z_{21} & \dots & z_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & z_{n1} & \dots & z_{nr} \end{bmatrix}$$

$$\underset{((r+1) \times m)}{\beta} = \begin{bmatrix} \beta_{01} & \beta_{02} & \dots & \beta_{0m} \\ \beta_{11} & \beta_{12} & \dots & \beta_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{r1} & \beta_{r2} & \dots & \beta_{rm} \end{bmatrix}$$

$$\underset{(n \times m)}{\epsilon} = \begin{bmatrix} \epsilon_{11} & \epsilon_{12} & \dots & \epsilon_{1m} \\ \epsilon_{21} & \epsilon_{22} & \dots & \epsilon_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \epsilon_{n1} & \epsilon_{n2} & \dots & \epsilon_{nm} \end{bmatrix}$$

### Application au problème de recalage

Afin de pouvoir trouver la transformation liant nos deux images, il faut placer un nuage de points fiduciaux. Chaque point sur une image possède un homologue sur l'autre image. Ainsi nous pouvons construire deux tableaux contenant les coordonnées des points fiduciaux de chaque image, dans le même ordre. Soit  $Z$  le tableau des points fiduciaux de l'image source et  $Y$  celui de l'image cible. Nous ne considérons que le cas des transformations affines. Cela signifie que les transformations possibles liant les points fiduciaux de l'image source et de l'image cible sont composées de translation, rotation, mise à l'échelle et cisaillement. Ainsi nous pouvons écrire en coordonnées homogènes :

$$\underset{(n \times m)}{Y} = \underset{(n \times (r+1))}{Z} \times \underset{((r+1) \times m)}{\beta} + \underset{(n \times m)}{\epsilon}$$

$\beta$  est la matrice de la transformation à estimer et correspond à la matrice des coefficients de régression. Il s'agit d'une matrice de taille  $(r+1) \times m$  où  $m$  est la dimension de l'image

cible et  $r$  la dimension de l'image source. La matrice  $\epsilon$  correspond au déplacement après application de la transformation aussi appelé résidu. Nous considérons que les résidus sont indépendants et identiquement distribués suivant  $\mathcal{N}_m(0, \Sigma)$ . Il n'y a pas de contraintes particulières sur  $\Sigma$ . Cela signifie que nous nous plaçons dans le cas de bruit anisotropique et que la covariance du bruit suivant les différentes dimensions n'est pas forcément nulle. Trouver la meilleure solution de ce problème de recalage dans le cadre des transformations affines revient à résoudre un problème de régression linéaire multivariée.

## Estimation de l'erreur de recalage

Soit  $\hat{\beta} = (Z'Z)^{-1}Z'Y$  l'estimateur de  $\beta$  par la méthode des moindres carrés.  $\hat{\beta}$  est le meilleur estimateur linéaire sans biais de  $\beta$ . Soit  $\hat{\epsilon} = Y - Z\hat{\beta}$  l'estimation de l'erreur résiduelle  $\epsilon$  après application de la transformation approchée  $\hat{\beta}$ . Soit  $\hat{\Sigma} = \frac{\hat{\epsilon}'\hat{\epsilon}}{n}$  une estimation de  $\Sigma$ .  $\hat{\beta}$  est l'estimation de  $\beta$  qui minimise la trace de  $\hat{\Sigma}$ . Une fois que nous avons ajusté le modèle de régression, nous pouvons prédire les nouvelles positions pour tous les points de l'image source. Nous souhaitons également estimer l'erreur de prédiction associée à chaque nouveau point. Soit  $y_0$  l'image d'un nouveau point  $z_0$ , alors suivant le modèle de régression  $y_0 = \beta' \begin{bmatrix} 1 \\ z_0 \end{bmatrix} + \epsilon_0$ .  $\epsilon_0$  est indépendant de  $\epsilon$ . L'erreur de prédiction associée est  $y_0 - \hat{\beta}' \begin{bmatrix} 1 \\ z_0 \end{bmatrix} = (\beta - \hat{\beta})' \begin{bmatrix} 1 \\ z_0 \end{bmatrix} + \epsilon_0$ .  $y_0 - \hat{\beta}' \begin{bmatrix} 1 \\ z_0 \end{bmatrix}$  est distribué suivant  $\mathcal{N}_m(0, (1 + \begin{bmatrix} 1 \\ z_0 \end{bmatrix}' (Z'Z)^{-1} \begin{bmatrix} 1 \\ z_0 \end{bmatrix}) \Sigma)$ . L'ellipse de prédiction au seuil  $(1 - \alpha)$  pour  $y_0$  est donnée par :

$$\begin{aligned} & (Y_0 - \hat{\beta}' \begin{bmatrix} 1 \\ z_0 \end{bmatrix})' \left( \frac{n}{n-r-1} \hat{\Sigma} \right)^{-1} (Y_0 - \hat{\beta}' \begin{bmatrix} 1 \\ z_0 \end{bmatrix}) \\ & \leq \\ & (1 + \begin{bmatrix} 1 \\ z_0 \end{bmatrix}' (Z'Z)^{-1} \begin{bmatrix} 1 \\ z_0 \end{bmatrix}) \left( \frac{m(n-r-1)}{n-r-m} \right) F_{m, n-r-m}(1-\alpha) \end{aligned}$$

$F_{m, n-r-m}(1-\alpha)$  correspond au  $(1-\alpha)^{ième}$  percentile d'une loi de Fisher de paramètres  $m$  et  $n-r-m$ .

## Modèle rigide

Sous l'hypothèse du modèle rigide nous supposons que la transformation recherchée est composée d'une rotation  $R_\theta$  et

d'une translation  $t$ . Les conditions sur  $\epsilon$  restent inchangées.

$$\begin{aligned} Y &= Z \times \beta + \epsilon \\ \beta &= \begin{bmatrix} t \\ R_\theta \end{bmatrix} \\ \hat{\beta} &= \begin{bmatrix} \hat{t} \\ \hat{R}_\theta \end{bmatrix} \end{aligned}$$

Soient  $y_i$ ,  $z_i$  et  $\epsilon_i$  les  $i^{ième}$  vecteurs ligne de  $Y$ ,  $Z$  et  $\epsilon$  respectivement. Soit la fonction de vraisemblance :

$$\begin{aligned} \mathcal{L}(Y, Z, \theta, t) &= P(Y|Z, \theta, t) \\ &= \prod_{i=1}^n P(y_i|z_i, \theta, t) \end{aligned}$$

Puisque les  $\epsilon_i$  sont identiquement distribués suivant  $\mathcal{N}_m(0, \Sigma)$  :

$$\begin{aligned} & P(y_i|z_i, \theta, t) \\ &= \\ & \frac{1}{\sqrt{2\pi|\Sigma|}} \exp \left( - \frac{[y_i - (R_\theta z_i + t)]' \Sigma^{-1} [y_i - (R_\theta z_i + t)]}{2} \right) \end{aligned}$$

Puisque le log est strictement croissant, maximiser la vraisemblance est équivalent à maximiser la log-vraisemblance :

$$\begin{aligned} & \log(\mathcal{L}(Y, Z, \theta, t)) \\ &= \\ & \sum_{i=1}^n \frac{1}{\sqrt{2\pi|\Sigma|}} - \frac{1}{2} \sum_{i=1}^n [y_i - (R_\theta z_i + t)]' \Sigma^{-1} [y_i - (R_\theta z_i + t)] \end{aligned}$$

Trouver  $\hat{R}_\theta$  et  $\hat{t}$  pour maximiser la log-vraisemblance est équivalent à minimiser  $\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' \Sigma^{-1} [y_i - (\hat{R}_\theta z_i + \hat{t})]$  qui équivaut à minimiser  $\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' [y_i - (\hat{R}_\theta z_i + \hat{t})]$  qui est équivalent à minimiser la trace de  $\hat{\Sigma}$ . Le problème de recalage dans ce cas est aussi connu comme le problème de Procrustes orthogonal. (*Schönemann, 1966*) propose une solution à ce problème dans le cas où la matrice  $\hat{R}_\theta$  est orthogonale. Il peut s'agir d'une rotation ou d'une réflexion. (*Kabsch, 1978*) donne une solution similaire sous contrainte que  $\hat{R}_\theta$  soit effectivement une rotation.

Soit un point  $z_0$  appartenant à l'image source. Soit  $\Delta_{z_0}(\hat{\theta}, \hat{t})$  l'erreur de recalage au point  $z_0$ .

$$\begin{aligned} \Delta_{z_0}(\hat{\theta}, \hat{t}) &= R_\theta z_0 + t + \epsilon_0 - (\hat{R}_\theta z_0 + \hat{t}) \\ &= e_{z_0}(\hat{\theta}, \hat{t}) + \epsilon_0 \end{aligned}$$

$\Delta_{z_0}(\hat{\theta}, \hat{t})$  admet une matrice de covariance  $\Sigma_{\Delta_{z_0}}$ . Puisque  $\epsilon_0$  est indépendant de  $e_{z_0}$ , nous avons  $\Sigma_{\Delta_{z_0}} = \Sigma_{e_{z_0}} + \Sigma$ .

En dimension 2 :

$$\begin{aligned} y_i &= \begin{bmatrix} y_{ix} \\ y_{iy} \end{bmatrix} & z_i &= \begin{bmatrix} z_{ix} \\ z_{iy} \end{bmatrix} \\ z_0 &= \begin{bmatrix} z_{0x} \\ z_{0y} \end{bmatrix} & t &= \begin{bmatrix} t_x \\ t_y \end{bmatrix} \\ R_\theta &= \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} & e_{z_0}(\hat{\theta}, \hat{t}_x, \hat{t}_y) &= \begin{bmatrix} f_1(\hat{\theta}, \hat{t}_x, \hat{t}_y) \\ f_2(\hat{\theta}, \hat{t}_x, \hat{t}_y) \end{bmatrix} \end{aligned}$$

Avec :

$$\begin{aligned} f_1(\hat{\theta}, \hat{t}_x, \hat{t}_y) &= z_{0x} \cos \theta - z_{0y} \sin \theta + t_x \\ &\quad - z_{0x} \cos \hat{\theta} + z_{0y} \sin \hat{\theta} - \hat{t}_x \end{aligned}$$

$$\begin{aligned} f_2(\hat{\theta}, \hat{t}_x, \hat{t}_y) &= z_{0x} \sin \theta + z_{0y} \cos \theta + t_y \\ &\quad - z_{0x} \sin \hat{\theta} - z_{0y} \cos \hat{\theta} - \hat{t}_y \end{aligned}$$

Soit  $J_{e_{z_0}}(\hat{\theta}, \hat{t}_x, \hat{t}_y)$  la matrice Jacobienne de  $e_{z_0}(\hat{\theta}, \hat{t})$  :

$$\begin{aligned} J_{e_{z_0}}(\hat{\theta}, \hat{t}_x, \hat{t}_y) &= \begin{bmatrix} \frac{\partial f_1(\hat{\theta}, \hat{t}_x, \hat{t}_y)}{\partial \hat{\theta}} & \frac{\partial f_1(\hat{\theta}, \hat{t}_x, \hat{t}_y)}{\partial \hat{t}_x} & \frac{\partial f_1(\hat{\theta}, \hat{t}_x, \hat{t}_y)}{\partial \hat{t}_y} \\ \frac{\partial f_2(\hat{\theta}, \hat{t}_x, \hat{t}_y)}{\partial \hat{\theta}} & \frac{\partial f_2(\hat{\theta}, \hat{t}_x, \hat{t}_y)}{\partial \hat{t}_x} & \frac{\partial f_2(\hat{\theta}, \hat{t}_x, \hat{t}_y)}{\partial \hat{t}_y} \end{bmatrix} \\ &= \begin{bmatrix} z_{0x} \sin \hat{\theta} + z_{0y} \cos \hat{\theta} & -1 & 0 \\ -z_{0x} \cos \hat{\theta} + z_{0y} \sin \hat{\theta} & 0 & -1 \end{bmatrix} \end{aligned}$$

Selon la loi de propagation des incertitudes,  $\Sigma_{e_{z_0}}$  est asymptotiquement égal à :

$$\Sigma_{e_{z_0}} = J_{e_{z_0}} \Sigma_{\hat{\theta}, \hat{t}_x, \hat{t}_y} J_{e_{z_0}}'$$

Où  $\Sigma_{\hat{\theta}, \hat{t}_x, \hat{t}_y}$  correspond à la matrice de covariance des paramètres  $\hat{\theta}$ ,  $\hat{t}_x$  et  $\hat{t}_y$ . Finalement nous avons :

$$\Sigma_{e_{z_0}} = \begin{bmatrix} \Sigma_{e_{z_0}}^{11} & \Sigma_{e_{z_0}}^{12} \\ \Sigma_{e_{z_0}}^{21} & \Sigma_{e_{z_0}}^{22} \end{bmatrix}$$

$$\begin{aligned} \Sigma_{e_{z_0}}^{11} &= \text{cov}(\hat{\theta}, \hat{\theta})(z_{0x} \sin \hat{\theta} + z_{0y} \cos \hat{\theta})^2 \\ &\quad - 2\text{cov}(\hat{\theta}, \hat{t}_x)(z_{0x} \sin \hat{\theta} + z_{0y} \cos \hat{\theta}) \\ &\quad + \text{cov}(\hat{t}_x, \hat{t}_x) \end{aligned}$$

$$\begin{aligned} \Sigma_{e_{z_0}}^{12} &= \text{cov}(\hat{\theta}, \hat{\theta})(z_{0x} \sin \hat{\theta} + z_{0y} \cos \hat{\theta})(-z_{0x} \cos \hat{\theta} + z_{0y} \sin \hat{\theta}) \\ &\quad - \text{cov}(\hat{\theta}, \hat{t}_x)(-z_{0x} \cos \hat{\theta} + z_{0y} \sin \hat{\theta}) \\ &\quad - \text{cov}(\hat{\theta}, \hat{t}_y)(z_{0x} \sin \hat{\theta} + z_{0y} \cos \hat{\theta}) \\ &\quad + \text{cov}(\hat{t}_x, \hat{t}_y) \end{aligned}$$

$$\Sigma_{e_{z_0}}^{12} = \Sigma_{e_{z_0}}^{21}$$

$$\begin{aligned} \Sigma_{e_{z_0}}^{22} &= \text{cov}(\hat{\theta}, \hat{\theta})(-z_{0x} \cos \hat{\theta} + z_{0y} \sin \hat{\theta})^2 \\ &\quad - 2\text{cov}(\hat{\theta}, \hat{t}_y)(-z_{0x} \cos \hat{\theta} + z_{0y} \sin \hat{\theta}) \\ &\quad + \text{cov}(\hat{t}_y, \hat{t}_y) \end{aligned}$$

Il reste à déterminer  $\Sigma_{\hat{\theta}, \hat{t}_x, \hat{t}_y}$ . Selon l'inégalité de Cramer-Rao, nous avons asymptotiquement :

$$\Sigma_{\hat{\theta}, \hat{t}_x, \hat{t}_y} = \begin{bmatrix} \Sigma_{\hat{t}, \hat{t}} & \Sigma_{\hat{t}, \hat{\theta}} \\ \Sigma_{\hat{\theta}, \hat{t}} & \Sigma_{\hat{\theta}, \hat{\theta}} \end{bmatrix} \geq \begin{bmatrix} -\mathcal{H}_{t,t} & -\mathcal{H}_{t,\theta} \\ -\mathcal{H}_{\theta,t} & -\mathcal{H}_{\theta,\theta} \end{bmatrix}^{-1} = \mathcal{I}^{-1}$$

Où  $\mathcal{I}$  représente la matrice d'information de Fisher et  $\mathcal{H}$  représente la matrice Hessienne de la log-vraisemblance.

$$\mathcal{H}_{t,t} = \frac{\partial^2 \log(P(Y|Z, t, \theta))}{\partial t^2}(\hat{\theta}, \hat{t}_x, \hat{t}_y)$$

$$\mathcal{H}_{t,\theta} = \frac{\partial^2 \log(P(Y|Z, t, \theta))}{\partial t \partial \theta}(\hat{\theta}, \hat{t}_x, \hat{t}_y)$$

$$\mathcal{H}_{\theta,\theta} = \frac{\partial^2 \log(P(Y|Z, t, \theta))}{\partial \theta^2}(\hat{\theta}, \hat{t}_x, \hat{t}_y)$$

$$\mathcal{H}_{t,t} = \sum_{i=1}^N -\Sigma^{-1}$$

$$\mathcal{H}_{t,\theta} = \sum_{i=1}^N \Sigma^{-1} \begin{bmatrix} z_{ix} \sin \hat{\theta} + z_{iy} \cos \hat{\theta} \\ -z_{ix} \cos \hat{\theta} + z_{iy} \sin \hat{\theta} \end{bmatrix}$$

$$\begin{aligned} \mathcal{H}_{\theta,\theta} &= \sum_{i=1}^N \begin{bmatrix} z_{ix} \sin \hat{\theta} + z_{iy} \cos \hat{\theta} \\ -z_{ix} \cos \hat{\theta} + z_{iy} \sin \hat{\theta} \end{bmatrix}' \Sigma^{-1} \begin{bmatrix} z_{ix} \sin \hat{\theta} + z_{iy} \cos \hat{\theta} \\ -z_{ix} \cos \hat{\theta} + z_{iy} \sin \hat{\theta} \end{bmatrix} \\ &\quad + \begin{bmatrix} z_{ix} \cos \hat{\theta} - z_{iy} \sin \hat{\theta} \\ -z_{ix} \sin \hat{\theta} + z_{iy} \cos \hat{\theta} \end{bmatrix}' \Sigma^{-1} \begin{bmatrix} y_{ix} - z_{ix} \cos \hat{\theta} + z_{iy} \sin \hat{\theta} - \hat{t}_x \\ y_{iy} - z_{ix} \sin \hat{\theta} - z_{iy} \cos \hat{\theta} - \hat{t}_y \end{bmatrix} \end{aligned}$$

$y_i$  et  $z_i$  correspondent au  $i^{\text{ième}}$  point fiduciaire de l'image cible et de l'image source respectivement.

## Annexe 1

Preuve que  $\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' \Sigma^{-1} [y_i - (\hat{R}_\theta z_i + \hat{t})]$  est équivalent à minimiser

$$\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' [y_i - (\hat{R}_\theta z_i + \hat{t})].$$

$\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' [y_i - (\hat{R}_\theta z_i + \hat{t})]$  peut-être écrit comme  $\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' A [y_i - (\hat{R}_\theta z_i + \hat{t})]$  avec  $A = I$  une matrice symétrique. De même  $\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' \Sigma^{-1} [y_i - (\hat{R}_\theta z_i + \hat{t})]$  peut-être écrit comme  $\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' A [y_i - (\hat{R}_\theta z_i + \hat{t})]$  avec  $A = \Sigma^{-1}$ .  $A$  est symétrique car  $\Sigma$  est une matrice symétrique.

Soit

$$g(\hat{R}_\theta, \hat{t}) = \sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' A [y_i - (\hat{R}_\theta z_i + \hat{t})]$$

$$= \sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' \times nA \times \sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]$$

Soit  $f(\hat{R}_\theta, \hat{t}) = \sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]$ . Nous avons donc :

$$g(\hat{R}_\theta, \hat{t}) = f(\hat{R}_\theta, \hat{t})' \times nA \times f(\hat{R}_\theta, \hat{t})$$

Trouver un minimum ou maximum de  $g(\hat{R}_\theta, \hat{t})$  revient à trouver  $\hat{R}_\theta$  et  $\hat{t}$  tels que  $\frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial \hat{t}} = 0$  et  $\frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial \hat{R}_\theta} = 0$ . En utilisant la composition des dérivée nous avons :

$$\frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial \hat{t}} = \frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial f(\hat{R}_\theta, \hat{t})} \times \frac{\partial f(\hat{R}_\theta, \hat{t})}{\partial \hat{t}}$$

$$\frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial \hat{R}_\theta} = \frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial f(\hat{R}_\theta, \hat{t})} \times \frac{\partial f(\hat{R}_\theta, \hat{t})}{\partial \hat{R}_\theta}$$

Sachant que  $\frac{\partial x'Ax}{\partial x} = 2x'A$  quand  $A$  est une matrice symétrique nous avons :

$$\frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial \hat{t}} = 2nf(\hat{R}_\theta, \hat{t})' A \times \frac{\partial f(\hat{R}_\theta, \hat{t})}{\partial \hat{t}}$$

$$\frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial \hat{R}_\theta} = 2nf(\hat{R}_\theta, \hat{t})' A \times \frac{\partial f(\hat{R}_\theta, \hat{t})}{\partial \hat{R}_\theta}$$

Supposant que  $A \neq 0$ , alors  $\frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial \hat{t}} = 0 \Rightarrow f(\hat{R}_\theta, \hat{t})' = 0$  ou  $\frac{\partial f(\hat{R}_\theta, \hat{t})}{\partial \hat{t}} = 0$  indépendamment de  $A$ . De même,  $\frac{\partial g(\hat{R}_\theta, \hat{t})}{\partial \hat{R}_\theta} = 0 \Rightarrow f(\hat{R}_\theta, \hat{t})' = 0$  ou  $\frac{\partial f(\hat{R}_\theta, \hat{t})}{\partial \hat{R}_\theta} = 0$  indépendamment de  $A$ . Ainsi  $\hat{R}_\theta$  et  $\hat{t}$  qui minimisent  $\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' A [y_i - (\hat{R}_\theta z_i + \hat{t})]$  sont indépendants de  $A$ . Donc minimiser  $\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' \Sigma^{-1} [y_i - (\hat{R}_\theta z_i + \hat{t})]$  est équivalent à minimiser  $\sum_{i=1}^n [y_i - (\hat{R}_\theta z_i + \hat{t})]' [y_i - (\hat{R}_\theta z_i + \hat{t})]$ .

## Simulations

Nous avons effectué plusieurs simulations pour vérifier la validité et la précision de nos développements. Le protocole de simulation est le suivant :

- Générer un set de points fiduciaires associés via une transformation connue  $T$
- Ajouter un bruit gaussien au set de fiduciaires cible
- Sélectionner un point  $x_0$  aléatoirement dans l'espace source
- Sélectionner un point  $y_0$  image de  $x_0$  par  $T$  avec un bruit gaussien
- Dessiner l'ellipse de prédiction de  $y_0$  à 95%
- Exécuter le processus de recalage pour les modèles rigide et affine
- Dessiner l'ellipse de prédiction en  $x_0$  à 95% pour les modèles rigide et affine

La simulation est répétée 1 000 000 de fois. Afin de vérifier la taille des ellipses, nous calculons le nombre de fois où le point  $y_0$  est compris dans l'ellipse de prédiction ainsi que l'aire des ellipses. La matrice de covariance du bruit est :

$$\Sigma = \begin{bmatrix} 100 & 50 \\ 50 & 200 \end{bmatrix}$$

## Transformation rigide

Dans cette simulation la transformation connue est une transformation rigide choisie aléatoirement.

Modèle	#n	% $y_0 \in ellipse_{95\%}$	Aire $ellipse_{95\%}$
Vrai	10	95.004	3 339.197
Rigide	10	95.053	5 696.962
Affine	10	95.058	8 918.060
Vrai	25	94.982	3 339.197
Rigide	25	94.183	3 779.010
Affine	25	94.994	4 415.793
Vrai	100	94.982	3 339.197
Rigide	100	94.717	3 413.270
Affine	100	94.995	3 539.127

Dans le cas où la vraie transformation est rigide, les ellipses trouvées contiennent le point  $y_0$  environ 95% du temps. Cela semble cohérent puisque elles doivent correspondre à des ellipses de prédiction à 95%. Quand le nombre de points fiduciaires est petit le modèle rigide est plus précis que le modèle affine car l'aire de l'ellipse de prédiction correspondante est plus petite. Quand  $n$  devient grand les modèles rigide et affine tendent à se confondre et convergent vers le vrai modèle.

## Transformation affine

Dans cette simulation la transformation connue est une transformation affine choisie aléatoirement.

Modèle	#n	% $y_0 \in ellipse_{95\%}$	Aire $ellipse_{95\%}$
Vrai	10	95.004	3 339.197
Rigide	10	97.824	55 941.551
Affine	10	95.152	8 915.317
Vrai	25	95.005	3 339.197
Rigide	25	97.792	35 719.012
Affine	25	95.033	4 415.452
Vrai	100	95.014	3 339.197
Rigide	100	98.435	31 365.125
Affine	100	95.004	3 539.246

Dans le cas où la vraie transformation est affine mais non-rigide, le modèle affine donne de bons résultats mais le modèle rigide n'est plus valide. Pour le modèle rigide  $y_0$  est inclu dans l'ellipse plus de 97% du temps et l'aire de l'ellipse est bien plus grande. Cela indique que la méthode utilisée pour estimer l'ellipse de prédiction n'est plus valide et que l'erreur de recalage associée est bien plus importante.

## Conclusion

Dans une première partie nous avons montré comment le problème de recalage peut être vu comme un problème de régression linéaire. Cela permet notamment de recalibrer des images dans le cas où la transformation recherchée est affine. Nous avons montré que les outils de la régression linéaire permettent d'estimer l'erreur de recalage. Par ailleurs nous avons montré une méthode décrite dans la littérature permettant d'estimer l'erreur de recalage dans le cadre spécifique de transformation rigide. Enfin nous montrons grâce à des simulations que le recalage par la régression linéaire est plus robuste que la méthode rigide dans le cas où la vraie transformation n'est effectivement pas rigide. Cela est important d'un point de vue pratique où le vrai modèle de transformation est incertain.