Traitement de l'erreur dans les méthodes de Hachage Géométrique et d'Alignement, Une analyse théorique.

Xavier PENNEC [1]

3 novembre 1993

[1]Ecole Polytechnique, DEA IMA 1992/93, Filière Informatique Géométrique. Stage effectué à l'INRIA Sophia-Antipolis, projet EPIDAURE, sous la direction de N. AYACHE, du 1 Mars au 31 Juillet 1993.

Table des matières

Traitement de l'erreur dans les méthodes de Hachage Géométrique et d'Alignement, Une analyse théorique.

Résumé

Le sujet de ce travail s'inscrit dans le cadre de la vision artificielle, et concerne plus precisément la reconnaissance des formes géométriques.
Il existe actuellement un grand nombre de méthodes pour reconnaître des objets rigides dont on possède un modèle géométrique a priori (Grimson [6]). Parmi celles-ci, le Hachage Géométrique (Lamdan & Wolfson [1]) et l'Alignement (ou Prédiction-Vérification) (Huttenlocher & Ullman [3], Ayache [8]) permettent une réduction importante de la complexité moyenne, tout en autorisant des occultations partielles sur les modèles. La théorie ne s'appliquant qu'à des données exactes, de nombreuses heuristiques sont utilisées en pratique pour adapter les algorithmes à des données réelles où il y a une erreur de positionnement.

La présente étude fournit un cadre théorique en introduisant un modèle d'erreur uniforme et bornée. On définit ainsi la zone compatible dans l'espace image, ou zone de recherche, pour l'alignement, puis la zone de compatibilité dans l'espace des invariants pour la hachage géométrique. Le calcul de ces zones et leur influence sur les algorithmes est développé sur trois exemples de transformations couramment utilisées : les transformations affines, rigides 2D et plus particulièrement pour l'imagerie médicale, les transformations rigides 3D. La formulation générale de ces zones ouvre le problème de la discrétisation de la table de hachage pour lequel nous proposons des solutions inédites. Enfin, l'étude du mécanisme de vote inhérent aux deux algorithmes fournit la probabilité d'une "reconnaissance fantôme", ainsi que le nombre moyen de ces fausses associations. Ces nombres permettent non seulement de justifier (ou non) l'emploi de l'algorithme, mais aussi de calculer le seuil d'acceptation qui assure une efficacité fixée.

Contributions majeures

En ce qui concerne la reconnaissance pour des transformations affines, nous présentons une extension des travaux de Grimson & al. ([4, 5]) aux espaces de dimension quelconque. Une méthode originale est proposée pour traiter l'erreur lors de l'apprentissage avec le hachage géométrique.
La partie concernant les transformations rigides 3D, particulièrement interessante pour l'imagerie médicale, est entièrement nouvelle et offre de nouvelles possibilités pour le traitement de transformations quasi-rigides (recalage inter-patients).

Pour toutes ces transformations, une dicrétisation des tables de hachage adaptée au traitement de l'erreur a été développée.
Enfin, nous avons montré que les analyses du nombre de faux positifs dues à Grimson & al. et Lamdan & Wolfson étaient intrinsèquement identiques.

Remerciements

Je remercie tout d'abord Nicholas Ayache qui m'a accueilli très chaleuresement au sein du projet EPIDAURE pour ce stage et sans qui ce travail n'aurait pu avoir lieu. C'est également Nicholas qui m'avait auparavant introduit à ces nouvelles méthodes de reconnaissance lors de son cours de DEA sur la vision, et qui m'a ouvert cette voie qui s'est révélée si fertile. Je dois beaucoup à Jacques Feldmar pour m'avoir procuré avant même le début du stage les articles qui m'ont servi de base, et pour toute l'assistance qu'il m'a fournie au cours du stage.
Un grand merci aussi à Jean-Philippe Thirion qui m'a patiemment expliqué tout ce qui concerne l'imagerie médicale (et Epidaure-Lib) et m'a guidé dans de nombreux domaines. Enfin, je remercie Hervé Delingette, Serge Benayoun et Jérome Declerck, qui ont stoïquement supporté mes divagations méta-mathématiques, et bien entendu Gérard Subsol sans qui "rien ne saurait marcher".

Chapitre 1

Introduction

Dans le cadre de la vision artificielle, un problème central est la reconnaissance des formes géométriques. S'il existe actuellement un grand nombre de méthodes pour reconnaître des objets rigides dont on possède un modèle géométrique a priori (Grimson [6]), deux techniques récemment introduites, le Hachage Géométrique (Lamdan & Wolfson [1]) et l'Alignement (ou Prédiction-Vérification) (Huttenlocher & Ullman [3], Ayache [8]) permettent une réduction importante de la complexité moyenne, tout en autorisant des occultations partielles sur les modèles. La théorie ne s'appliquant qu'à des données exactes, de nombreuses heuristiques sont utilisées en pratique pour adapter les algorithmes à des données réelles où il y a une erreur de positionnement.
La recherche d'une formalisation théorique la plus générale possible pour le traitement de l'erreur dans ces algorithmes repose sur le constat suivant :

- Un cadre théorique est nécessaire pour guider, évaluer et valider les différentes heuristiques utilisées.
- Aucune des heuristiques utilisées n'assure l'exactitude de l'algorithme.
- Enfin, l'application de ce cadre général à des cas particuliers peut fournir des algorithmes tout à fait compétitifs, ou en tout cas des idées novatrices.

Nous avons choisi de développer la modélisation d'une erreur bornée proposée par Grimson & al. [4, 5], mais d'autres voies sont possibles, en particulier la modélisation gaussienne de l'erreur qui conduit à des algorithmes de reconnaissance probabilistes [15, 16, 17]. Les travaux précédents ([4, 5, 2]) se sont focalisés sur les transformations en deux dimensions, très utilisées en vision pour la robotique. Pour notre part, nous nous sommes interressé plus particulièrement (mais pas uniquement) aux transformations en 3 dimensions, puisque ce sont les objectifs de l'imagerie médicale.

L'un des buts de recherche de l'équipe INRIA Epidaure est de trouver des méthodes de traitement automatique des images 3D issues de l'instrumentation médicale, comme les images obtenues par scanner X, Résonance Magnétique (IRM), ou Tomographie par émission de positrons (TEP). Un enjeu majeur de ce domaine est le recalage de telles images entre elles de manière à pouvoir les comparer précisément. Toutefois, si l'on connaît des méthodes efficaces pour le recalage d'images de même type et du même patient [7, 18, 19, 20], le recalage inter-patient ou patient-atlas anatomique reste très difficile. Des algorithmes prenant en compte l'erreur de façon automatique pourraient trouver ici une application tout à fait adaptée.

Chapitre 2

Cadre général de l'étude

2.1 Reconnaissance d'objets basée sur des modèles

On dispose d'une bibliothèque d'objets $\mathcal{B}ib = \{\mathcal{M}^1, \ldots \mathcal{M}^M\}$, chaque objet étant défini à partir d'un ensemble $\mathcal{P}$ de primitives géométriques (points, droites, plans, courbes, surfaces ...) et d'un système de vision fournissant un scène $\mathcal{S}$ décrite dans les mêmes termes.
On veut *reconnaître* et *localiser* les objets présents dans la scène. Soit $\mathcal{T}$ l'ensemble des transformations considérées (transformations rigides, similarités, affinités ...); on considère qu'un objet de la bibliothèque est reconnu dans la scène s'il existe une transformation T de $\mathcal{T}$ qui superpose (à une certaine erreur près) un nombre suffisant de primitives de l'objet à des primitives de la scène ; on dit alors que les primitives qui se superposent sont appariées ou mises en correspondance. On cherche donc à maximiser à la fois le nombre d'appariements entre l'objet et la scène et la qualité de ces appariements, c'est à dire l'adéquation de la superposition des primitives appariées par la transformation trouvée. Les algorithmes recherchés doivent fournir l'ensemble des objets reconnus munis de la transformation qui les projette dans la scène.

2.2 Formalisation

La scène $\mathcal{S}$ est un ensemble de n primitives noté $\mathcal{S} = \{S_1, \ldots S_n\}$ avec $S_i \in \mathcal{P}$. De même, le modèle $\mathcal{M}^k$ du k^{ieme} objet de la bibliothèque est défini par un ensemble de m_k primitives: $\mathcal{M}^k = \{M_i^k, \ldots M_{m_k}^k\}$ avec $M_1^k \in \mathcal{P}$. On appelle fonction d'appariement f_k de ce modèle l'application qui fait correspondre à toutes les primitives M_i^k soit une primitive de la scène, soit la primitive nulle Γ, auquel cas la primitive n'a pas de correspondant dans la scène (elle est occultée ou l'objet est absent). Donc f_k est une application de $\mathcal{M}_k$ dans $\mathcal{S} \cup \{\Gamma\}$. On note F_k l'ensemble de ces applications[1]. Maximiser le nombre d'appariement d'un objet de

[1] F_k est isomorphe à l'ensemble des applications de $\{1 \ldots m_k\}$ dans $\{1 \ldots n+1\}$ et possède donc $(n+1)^{m_k}$ éléments.

$\mathcal{B}ib$ s'écrit donc :

$$f_k \in \arg\max_{f \in F_k} \left(m_k - Card(f^{-1}(\Gamma)) \right) \tag{2.1}$$

Parallèlement, on veut optimiser la qualité des appariements et trouver la transformation qui envoie le modèle dans la scène. Soit $\mathcal{D}ist$ une distance sur $\mathcal{P}$ (distance entre points, entre droites, ...), avec la convention que $\mathcal{D}ist(., \Gamma) = 0$; on cherche à minimiser l'erreur moyenne entre la transformée d'une primitive du modèle et son correspondant. La transformation T_k retenue pour le modèle $\mathcal{M}_k$ vérifie:

$$T_k \in \arg\min_{T \in \mathcal{T}} \left(\sum_i \mathcal{D}ist\left(T(\mathcal{M}_i) - f_k(\mathcal{M}_i)\right) \right) \tag{2.2}$$

On voit donc que ce problème couple la recherche dans l'espace des correspondances (trouver f_k dans F_k) avec la recherche dans l'espace des transformations (trouver T_k dans $\mathcal{T}$). Chacun de ces deux sous-problèmes indépendamment est simple, mais leur couplage rend la tâche beaucoup plus ardue et nécessite la définition d'un compromis.
Dans les paragraphes suivants, les méthodes développées s'intéressent à un seul objet $\mathcal{M}$ (on supprimera les indices relatifs au numéro de l'objet pour plus de simplicité). Il suffit ensuite de les appliquer à tous les objets de la bibliothèque pour obtenir l'algorithme définitif. On a donc affaire à des complexité linéaires en nombre d'objets (sauf dans le cas du hachage géométrique).

2.3 Arbre d'interprétation

Examinons les équations posées : la recherche de T dépend de la solution f trouvée. Par contre, on peut rechercher f dans l'espace F des correspondances et s'occuper à posteriori de T qui fournira le validation de l'interprétation. C'est l'idée que nous allons ici développer. L'espace F étant discret, on traite un problème classique d'optimisation combinatoire et sa résolution se fait par arbre. L'algorithme de base consiste donc à tester à chaque nœud de l'arbre l'appariement d'une primitive de $\mathcal{M}$ non encore utilisée avec chacune des primitives de $\mathcal{S} \cup \{\Gamma\}$, puis lorsqu'on arrive à une feuille, on recherche s'il existe une transformation de $\mathcal{T}$ qui convient.
Diverses méthodes classiques de l'intelligence artificielle permettent d'élaguer la recherche dans cet arbre exponentiel ($(n+1)^m$ feuilles), en particulier l'introduction de contraintes géométriques : supposons qu'on travaille avec des segments comme primitives ; la longueur du segment vu dans la scène ne peut être qu'inférieure (à l'erreur près) à celle du segment modèle, puisque les seules modifications possibles sont les erreurs de mesure et l'occultation : c'est une contrainte unaire sur l'appariement. Supposons maintenant l'appariement de deux paires de segments vérifiant les contraintes unaires ; les angles entre les deux droites supports dans la scène et dans le modèle doivent être quasiment identiques : c'est une contrainte binaire. On peut ainsi développer pour chaque type de primitives un ensemble de contraintes géométriques unaires et binaires qui permettent de conserver lors de la descente dans l'arbre une consistance locale de l'interprétation, la recherche de la transformation fournissant en

fin de compte et s'il y a lieu la preuve de la consistance globale. L'introduction de l'erreur s'effectue très simplement en propageant l'erreur de mesure sur les primitives dans le calcul des contraintes. Une étude complète de ces techniques est développée dans [6]. La complexité reste cependant $O(M(n+1)^m)$ dans le cas le pire si M est le nombre de modèles.

2.4 Transformée de Hough

La transformée de Hough opère directement dans l'espace des transformations. On note k le nombre minimum d'appariements nécessaires pour calculer une transformation entre le modèle et la scène. Pour chaque k-uplet d'appariements (M_i, S_j), on peut alors calculer les transformations possibles et on vote soit pour l'ensemble exact de ces transformations (transformée de Hough continue), soit pour les cases correspondantes dans l'espace discrétisé. Il suffit ensuite de vérifier les transformations ayant obtenues les meilleurs scores.
Cet algorithme a donné lieu à beaucoup d'études, en particulier pour traiter les erreurs de mesure. Celles-ci sont reprises dans [6]. La complexité est ici en $O(Mm^kn^k)$, mais l'algorithme nécessite le stockage en mémoire de l'espace des transformations qui peut être grand : si les transformations rigides en 2D n'occupent qu'un espace de dimension 3, on passe à une dimension 6 pour le 3D et à 9 pour les transformations affines 3D.

2.5 Alignement ou Prédiction-Vérification

La technique est un mélange de la transformée de Hough et de l'arbre d'interprétation : on commence par supposer un k-uplet d'appariements (M_i, S_j) qui permet de calculer la transformation superposant ces primitives, puis on vérifie ces hypothèses en calculant les transformées des primitives du modèle et en recherchant dans une certaine zone autour de ces primitives prévues des primitives de la scène pouvant convenir. Le score de qualité de cette hypothèse est simplement le nombre d'appariements trouvés, ou peut être un critère plus élaboré.
En théorie, on devrait réexécuter ce schéma pour tous les k-uplets d'appariements possibles et conserver les hypothèses de score maximal. En pratique, on s'arrête dès qu'on estime avoir reconnu et placé l'objet.
La complexité est en $O(Mm^kn^k)$ en prétraitant la scène grâce à une structure de 'baquets' qui permet la recherche des correspondants des primitives supposées en temps constant. L'introduction de l'erreur dans cette technique est naturelle puisqu'il suffit de propager les erreurs de mesure dans le calcul de la primitive prévue, ce qui donne une justification théorique à la dimension de la zone de recherche.

2.6 Hachage Géométrique

Le hachage géométrique est un algorithme permettant une reconnaissance sous-linéaire en nombre de modèles.

Dans le cas d'un sous-ensemble des transformations affines, on peut définir une base $\mathcal{B}$ d'un modèle en choisissant k points de celui-ci et exprimer les autres points dans cette base. Par exemple, deux points suffisent à définir une base orthonormée en dimension deux (en fait un point et une direction exactement), ou trois points non colinéaires en dimension trois, mais il faut trois points exactement en dimension deux pour une base affine et quatre en dimension trois. Les coordonnées sont alors invariantes : si $\mathcal{T}$ est une transformation de ce type et M un point, les coordonnées de $\mathcal{T}(M)$ dans $\mathcal{B}' = \mathcal{T}(\mathcal{B})$ et celles de M dans $\mathcal{B}$ sont identiques. L'idée est d'indexer chaque paire (modèle, base) par les coordonnées des autres points du modèle dans cette base grâce à une table de hachage. Il suffit alors lors de la reconnaissance de choisir une base image (constituée de k points) et d'exprimer les coordonnées des autres points image dans cette base, celles-ci servant à voter au travers de la table de hachage. Si la base fait partie d'un objet, le nombre de votes obtenus pour la paire (modèle, base) correspondante sera le nombre de points de l'objet vus (à k près). La complexité est donc en $O(n^{k+1})$ dans le pire des cas pour la reconnaissance, si le temps d'accès à un bucket est constant. Ceci est obtenu en considérant n votes pour chacune des n^k bases possibles dans la scène. La vérification n'est pas prise en compte puisque dans l'algorithme normal, les meilleurs résultats sont directement retenus.

On peut tenter de généraliser le Hachage Géométrique à des transformations dépassant le cadre linéaire, mais il faut alors beaucoup de rigueur sur les espaces considérés. On dispose déjà de deux espaces : l'espace image $\mathcal{I}$ (en général $\mathbb{R}^2$ ou $\mathbb{R}^3$) et l'espace des primitives $\mathcal{P}$, identique à l'espace image dans le cas des points, mais dont chaque point représente un ensemble dans l'espace image dans les autres cas. Par exemple, les segments orientés sont définis par deux points (les extrémités); donc $\mathcal{P}$ est homéomorphe à $\mathbb{R}^{2n}$. Dans le cas des droites, $\mathcal{P} \sim \mathcal{P}^{n-1}_{proj} \times \mathcal{P}^{n}_{proj}$ où $\mathcal{P}^{n}_{proj}$ est l'espace projectif de dimension n.
Une transformation $T \in \mathcal{T}$ opère de $\mathcal{P}$ dans $\mathcal{P}$, mais elle est définie à partir d'une transformation $T' \in \mathcal{T}'$ qui opère sur l'espace image $\mathcal{I}$. Si les applications de $\mathcal{T}'$ peuvent s'exprimer comme le plongement de $\mathcal{I}$ dans un certain espace E, puis une application linéaire de E dans E et enfin une projection de E dans $\mathcal{I}$, alors on peut définir une base de E en choisissant k points et les coordonnées des autres points sont invariantes.
Pour fixer les idées, prenons le cas de la vision classique : les perspectives de $\mathbb{R}^3$ sont un sous-ensemble des transformations projectives. On commence donc par plonger $\mathbb{R}^3$ dans $\mathcal{P}^3_{proj}$, mais $\mathcal{P}^3_{proj}$ est défini comme l'espace des directions de $\mathbb{R}^4$ et les transformations projectives sont justement l'équivalent dans $\mathcal{P}^3_{proj}$ des transformations linéaires dans $\mathbb{R}^4$ (à n'importe quelle homothétie près), ce qui explique que 5 points soient nécessaires pour définir une base. Les transformations projectives sont pour l'instant le seul exemple de cette généralisation, mais cette formalisation ouvre la voie à d'autres possibilités.

Il est à noter que le hachage géométrique peut aussi s'utiliser avec des invariants divers pour l'ensemble des transformations sans considération de base. On ne cherche alors qu'à reconnaître l'objet sans le localiser. Sous cette forme, il est souvent utilisé sans qu'on y fasse particulièrement attention. Il faut toutefois prendre garde que les mêmes considérations s'appliquent quant au traitement de l'erreur.

2.7 Introduction de l'erreur

Nous avons précédemment décrit les algorithmes théoriques qui correspondent au cas idéal. Dans la pratique, on peut mettre en évidence plusieurs facteurs se combinant lors du traitement d'image et conduisant à des erreurs de position sur les primitives de la scène :

- L'approximation de la transformation : c'est par exemple le cas en reconnaissance 2D d'objets 3D si l'on utilise une similitude pour approcher une projection orthogonale, ou lorsqu'on considère des transformations euclidiennes pour des objets très légèrement déformables (imagerie médicale).
- Les déformations introduites par le système de vision, qui ne sont en général pas linéaires.
- La discrétisation de l'image.
- Les prétraitements effectués sur l'image avant la reconnaissance : filtrages, extraction de contours ...

La présence de cette erreur, même si elle est très faible, nous oblige à reconsidérer les algorithmes. Dans le cas du geometric hashing par exemple, les erreurs de mesure dans la scène se répercutent sur le calcul des invariants et continuer à voter ponctuellement pour l'invariant trouvé conduirait à manquer un grand nombre d'appariements ; on obtiendrait alors des 'faux négatifs', c'est à dire qu'il peut ne pas y avoir reconnaissance alors qu'un objet est présent. La même chose peut se produire avec les techniques d'alignement si la zone de recherche des correspondants n'est pas adaptée à l'erreur.

Afin de pallier à ces défauts, on supposera connue une estimation de l'erreur de mesure sur les primitives, soit sous forme d'une borne sur les valeurs, soit sous forme d'une loi de probabilité. L'approche de Grimson et Huttenlocher [5, 6] ou Lamdan et Wolfson [2] consiste à propager la borne sur l'erreur de mesure dans la méthode de calcul des invariants afin d'obtenir une borne sur la zone de vote pour le hachage géométrique ou la zone de recherche pour l'alignement assurant de ne manquer aucun appariement possible. L'objet des chapitres 3 et 4 est justement de dériver ces bornes et de voir les modifications qu'elles apportent aux algorithmes.

La contrepartie de cette approche est la possibilité de reconnaissances fantômes. En effet, l'utilisation d'une zone de vote ou de recherche au lieu d'un point unique autorise l'appariement de points situés dans ces zones par hasard (conspiration). On s'attachera dans le chapitre 5 au calcul du nombre de faux positifs, ce qui nous permettra de dériver une complexité moyenne pour l'algorithme.

Une seconde approche est de considérer l'erreur sous forme probabiliste et de propager ceci dans les algorithmes, en conservant tout au long l'aspect probabiliste. On ne donnera ici faute de temps que quelques pointeurs sur ces méthodes, par ailleurs très intéressantes et prometteuses : différentes méthodes statistiques sont étudiées dans [13], mais les principaux travaux sur le hachage géométrique probabiliste sont dues à Rigoustous et Hummel [15, 16, 17], une petite extension étant présentée par Tsai dans [14].

2.8 Applications en imagerie médicale

L'imagerie médicale est un domaine particulier de la vision artificielle. Elle se propose de trouver des méthodes de traitement automatique des images 3D issues de l'instrumentation médicale, comme les images obtenues par scanner X, Résonance Magnétique (IRM) ou Tomographie par émission de positrons (TEP). Un défi majeur de ce domaine est le recalage de telles images entre elles de manière à pouvoir les comparer précisément : on peut ainsi étudier l'évolution temporelle des organes d'un patient grâce à des images prises à différentes époques, ceci principalement dans une optique de diagnostic, mais aussi de thérapie grâce à la localisation de pathologies. Les transformations que l'on recherche sont dans ce cas des transformations objectivement mesurables. Un autre centre d'intérêt est représenté par la comparaison inter-patient ou patient-atlas anatomique dans un but d'étude de la 'normalité' ou des déviations, ou de planification d'opérations chirurgicales. Un problème sur lequel travaille une partie du groupe EPIDAURE est ainsi la simulation d'opérations de chirurgie crâno-faciale. L'utilisation des algorithmes de traitement d'image est ici plutôt orientée vers la planification de la thérapie.
La différence principale par rapport aux technique classiques de reconnaissance est le traitement d'images en trois dimensions, dont les intensités en chaque pixel sont en général représentatifs d'une densité de matière. Les objets sur lesquels on travaille sont ainsi définis par des surfaces gauches non polygonales. Le problème principal pour l'application des algorithmes de reconnaissance de forme est l'extraction d''amers' ou primitives géométriques, définissant le modèle de façon fiable. Dans un premier temps, J.P. Thirion & al. ont montré la possibilité d'extraire de ces surfaces des lignes de crêtes, définies comme le lieu des maxima locaux de la courbure maximale, ce qui a permis d'effectuer des recalages précis entre images d'un même type et d'un même patient de façon fiable (voir [7, 19, 20, 21]). Récemment, J.P. Thirion a introduit une nouvelle méthode permettant d'extraire à partir de ces courbes des points extrémaux stables [18]. Les primitives ainsi extraites ont permis le recalage précis d'images d'un même patient à différentes époques. Le problème posé par le recalage inter-patients ou patient-atlas aux algorithmes que l'on utilise (alignement et hachage géométrique), est que la transformation à trouver n'est plus réelle : elle est subjective. En particulier, elle n'est plus rigide. Par contre, il est possible de se contenter de ce type de transformations si l'on considère que les points sont sujet à une erreur relativement importante : l'introduction d'algorithmes traitant l'erreur de façon intrinsèque offrira peut être une voie pour la résolution de ce problème. Enfin, il conviendrait de comprendre pourquoi les méthodes basées sur la minimisation du potentiel d'attraction entre deux objets, telles que les méthodes proposées par G. Malandain dans [22, 23] et basées sur l'utilisation de la mécanique donnent également de bons résultats pour le recalage mono-patient, dans le cas d'images de types différents.

Chapitre 3

Alignement : zone de recherche compatible

Dans ce chapitre et le suivant, on ne considérera qu'un sous-ensemble des transformations affines appliquées sur des primitives simples : des points.

3.1 Généralités

Connaissant ou supposant un appariement d'une bases du modèle $\mathcal{B}_{mod}$ et d'une base de l'image $\mathcal{B}_{im}$, le problème pour les techniques d'alignement est de savoir où rechercher les autres points du modèle dans l'image.
On possède un modèle parfait constitué de points dont les coordonnées sont exprimées dans la base modèle canonique $\mathcal{BM}$. De même, la scène est un ensemble de points exprimés dans la base image canonique $\mathcal{BI}$. On suppose l'appariement de k points de la scène et du modèle, k étant le nombre de points juste suffisant pour définir une base ou pour calculer la transformation. On peut alors utiliser deux types de méthodes : soit on recherche directement la transformation T qui superpose les points appariés (par exemple aux moindres carrés - voir 3.3 et 3.4), soit on définit par la même méthode une base $\mathcal{B}_{mod}$ dans le modèle et $\mathcal{B}_{im}$ dans la scène que l'on suppose se correspondre (voir 3.2).

$$\begin{tikzcd}
 & \mathcal{B}_{mod} = \mathcal{B}_{im} \arrow[dr, "T_{im}"] & \\
\mathcal{BM} \arrow[ur, "T_{mod}"] \arrow[rr, "T"] & & \mathcal{BI}
\end{tikzcd}$$

Le modèle étant parfait, le calcul de T_{mod} ne pose aucun problème. Par contre, les mesures des points de la scène (dans $\mathcal{BI}$) sont sujettes à l'erreur. Cette erreur se répercute dans le calcul de T ou T_{im} et donc sur le positionnement supposé des autres points du modèle.

Notations

Soit v une mesure sujette à l'erreur :

- v est sa valeur idéale, non corrompue par l'erreur.
- $\bar{v}$ est la valeur mesurée.
- δv est l'erreur sur la mesure : $\bar{v} = v + \delta v$

On notera m_i les points du modèle exprimés dans $\mathcal{BM}$ si on cherche T ou dans $\mathcal{B}_{mod}$ ($= \mathcal{B}_{im}$) si on cherche T_{im}, et s_i les points de la scène dans $\mathcal{BI}$. Comme on s'occupe d'un sous-ensemble des transformations affines, on peut définir la translation en choisissant une origine O_m dans le modèle et O_s dans la scène (on notera alors $x_i = m_i - O_m$ et $y_i = s_i - O_s$), ce qui permet de ne plus travailler que sur une application linéaire que l'on notera A (rotation, similarité, application linéaire générale ...).
La transformation générique s'écrit donc $y = Ax$ soit :

$$s = O_s + A(m - O_m)$$

Dans tous les cas, on réserve les k premiers indices aux k points appariés pour définir la transformation : m_i correspond à s_i (et donc x_i à y_i) pour $1 \leq i \leq k$, les autres points du modèle ou leur correspondant dans la scène étant génériquement notés sans indices.

3.1.1 Méthode générale

On se place pour la suite de cette partie dans le cadre d'une erreur bornée. Chacun des points image s_i est soumis à l'erreur, et on ne peut mesurer que $\bar{s}_i = s_i + \delta s_i$, avec $\|\delta s_i\| \leq \varepsilon_p$. A et O_s étant calculés à partir des s_i (et éventuellement des m_i mais ceux-ci sont exacts), ceci occasionne une erreur de mesure δO_s sur O_s et δA sur A. Considérons maintenant un point m du modèle (afin d'éviter des lourdeurs dans les formules, on utilisera dorénavant $x = m - O_m$). Son correspondant réel se situe en $s = O_s + A(m - O_m) = O_s + Ax$ et sera mesuré en $\bar{s} = s + \delta s$. Le correspondant prévu est lui en $s_{prev} = \bar{O}_s + \bar{A}x = s + \delta O_s + \delta A.x$. On aura donc :

$$\bar{s} = s_{prev} + \delta s - \delta O_s + \delta A.x$$

Le point $\bar{s}$ recherché peut finalement se trouver autour de l'emplacement prévu $s_{prev} = \bar{O}_s + \bar{A}x$ avec une erreur bornée en norme par :

$$\varepsilon_{al} = \max \|\delta s - \delta O_s + \delta A.x\| = \max \|\delta y - \delta Ax\| \leq \max \|\delta y\| + (\max \|\delta A\|)\|x\| \qquad (3.1)$$

Il nous reste à préciser une majoration de ces normes.

3.1.2 Approximation de δA

L'algorithme 'normal' (qui ne considère pas de borne sur l'erreur) fournit une méthode pour calculer A à partir des points image choisis pour former la base. On peut donc considérer A comme une fonction des points de base $A = \mathcal{A}(s_1, \dots s_k)$, ce qui permet d'écrire δA comme une différentielle (on note $(x)_j$ la j-ième composante de x et n la dimension de l'espace) :

$$\delta A = \bar{A} - A = \mathcal{A}(s_1 + \delta s_1, \dots s_k + \delta s_k) - \mathcal{A}(s_1, \dots s_k) \simeq \sum_{i=1}^{k} \left(\sum_{j=1}^{n} \left(\left. \frac{\partial \mathcal{A}}{\partial (s_i)_j} \right|_{(s_1, \dots s_k)} . (\delta s_i)_j \right) \right)$$

d'où la majoration :

$$\|\delta A\| \leq \varepsilon_p \, \mathcal{K}_A + O(\varepsilon_P^2) \quad \text{avec} \quad \mathcal{K}_A = \sum_{i=0}^{k} \sum_{j=1}^{n} \left\| \left. \frac{\partial \mathcal{A}}{\partial (s_i)_j} \right|_{(s_1, \dots s_k)} \right\|$$

En général, on peut assurer que $\|\delta O_s\| \leq \varepsilon_p$, et dans ce cas on obtient :

$$\varepsilon_{al} \leq \varepsilon_p \; (2 + \mathcal{K}_A \|x\|) + O(\varepsilon_p^2) \tag{3.2}$$

Bien que cette formule ne soit pas utilisée par la suite, elle est parfaitement représentative de ce qu'on est susceptible d'obtenir. On verra en effet qu'on dérivera toujours dans notre cas une formule de ce type. Il est à noter qu'on peut séparer dans cette formule l'influence des différents termes :

- ε_p est l'erreur accidentelle particulière à chacun des points considéré
- $\varepsilon_p + \mathcal{K}_A \|x\|$ est une erreur systématique sur la transformation, engendrée par l'erreur de mesure sur les k points constituant la base image : ε_p est l'erreur sur l'origine, donc la translation, et $\mathcal{K}_A \|x\|$ est produit par la partie linéaire de la transformation.

Ceci souligne bien ce qu'on a à gagner en précision en affinant au maximum la transformation.

3.2 Transformations affines

Comme précédemment, on note n la dimension de l'espace considéré (usuellement 2 ou 3). $n{+}1$ points en situation générale sont alors nécessaires pour définir une base affine : le premier point constitue l'origine, et les n autres points sont les extrémités des vecteurs de base. On définit donc une telle base $\mathcal{B}_{mod}$ pour le modèle dans laquelle on exprime les coordonnées des points du modèle. Celui-ci étant parfait, on peut travailler avec ces coordonnées sans problème. En ce qui concerne la scène, la base est définie de la même façon par son origine $O_s = s_1$, et par les vecteurs de base $e_i = s_{i+1} - s_1$. La matrice de passage de $\mathcal{B}_{im} (= \mathcal{B}_{mod})$ à la base image canonique $\mathcal{BI}$ s'écrit donc très simplement $A = [e_1, \dots e_n]$. Regardons maintenant l'erreur : $\delta A = [\delta s_2 - \delta s_1, \dots, \delta s_{i+1} - \delta s_1, \dots \delta s_{n+1} - \delta s_1]$. D'où l'expression :

$$\delta s - \delta O_s - \delta A x = \delta s - \delta s_1 - \sum_{i=1}^{n} x_i \delta e_i = \delta s - \sum_{i=1}^{n} x_i \delta s_{i+1} + \delta s_1 \left(-1 + \sum_{i=1}^{n} x_i \right)$$

Ce qui donne en prenant le maximum sur les δs_i :

$$\varepsilon_{al}(x) \leq \varepsilon_p \left(\left| -1 + \sum_{i=1}^{n} x_i \right| + 1 + \sum_{i=1}^{n} |x_i| \right)$$

La zone image compatible avec un point du modèle est donc caractérisée par une erreur maximum en norme de

$$\varepsilon_{al} \leq 2\varepsilon_p \left(1 + \|x\|_1\right) \tag{3.3}$$

Il est à noter qu'aucune supposition n'a été faite sur la norme de l'erreur : celle-ci peut aussi bien être la norme L_2 que L_∞. Ces formules généralisent à des dimensions supérieures les formules dérivées par Grimson dans [5].

3.3 Transformations rigides 2D

Afin de ne pas alourdir les notations, toutes les normes seront dorénavant des normes euclidiennes (norme L_2).

3.3.1 Invariant supplémentaire

Une telle transformation dépend de 3 paramètres : deux pour la translation et un pour la rotation. Deux points (4 variables) sont donc nécessaires pour définir une base. Supposons donc un appariement entre deux points du modèle (m_1, m_2) et deux points de la scène (s_1, s_2). Sachant qu'on mesure en fait $\bar{s}_1 = s_1 + \delta s_1$ et $\bar{s}_2 = s_2 + \delta s_2$ avec $\|\delta s_i\| \leq \varepsilon_p$, cet appariement n'est plausible que si :

$$\left| \|\bar{s}_1 - \bar{s}_2\| - \|m_1 - m_2\| \right| \leq 2\varepsilon_p$$

La distance entre les deux points considérés est donc un invariant qui servira à contraindre la recherche des associations. Nous préciserons ce point en fin de section.

3.3.2 Méthode de calcul de la transformation

Etant donné qu'il n'existe pas en général de transformation rigide exacte amenant les points m_0 et m_1 sur $\bar{s}_0$ et $\bar{s}_1$, la méthode consiste à choisir aux moindre carrés celle qui minimise l'erreur de superposition. On recherche donc l'origine $\bar{O}_s$ et la rotation $\bar{A}$ qui minimisent le critère

$$C = \|\bar{s}_1 - O_s - A(m_1 - O_m)\|^2 + \|\bar{s}_2 - O_s - A(m_2 - O_m)\|^2$$

On montre facilement que choisir les origines des repères aux barycentres convient pour minimiser l'erreur en translation : $O_m = (m_1 + m_2)/2$ et $\bar{O}_s = (\bar{s}_1 + \bar{s}_2)/2$, la rotation $\bar{A}$ étant alors celle qui amène les points $x_1 = m_1 - O_m$ et $x_2 = m_2 - O_m$ sur la droite $(\bar{y}_1, \bar{y}_2)$.

3.3.3 Expression de δA

On peut décomposer tout d'abord $\bar{A}$ en effectuant la rotation exacte A puis une rotation d'ajustement Γ : $\bar{A} = \Gamma A$. L'erreur sur la rotation s'écrit alors $\delta A = (\Gamma - I_d)A$ et puisqu'une rotation est de norme unité (en norme L_2), on a : $\|\delta A\| = \|\Gamma - I_d\|$. Calculer Γ équivaut simplement à se placer dans le cas particulier où $A = I_d$, c'est-à-dire calculer aux moindres carrés la rotation entre les y_i et les $\bar{y}_i$. Soit donc θ l'angle de rotation de l'ajustement :

$$\delta A = \begin{bmatrix} \cos\theta - 1 & -\sin\theta \\ \sin\theta & \cos\theta - 1 \end{bmatrix}$$

d'où l'on tire :

$$(\delta A)^t \ \delta A = \left((\cos\theta - 1)^2 + \sin^2\theta\right) \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

La norme euclidienne d'une matrice A est $\|A\| = \sqrt{\rho(A^t A)}$, si l'on note $\rho(.)$ le rayon spectral d'une matrice, i.e. sa valeur propre de norme la plus grande. On a donc dans notre cas :

$$\|\delta A\| = \sqrt{2}\sqrt{1 - \cos\theta} \tag{3.4}$$

3.3.4 Détermination de l'angle d'ajustement maximum θ_{max}

On remarquera tout d'abord que pour la rotation, on travaille en repère barycentrique, c'est-à-dire sur les y_i et $\bar{y}_i$. On a en particulier $\delta y_1 + \delta y_2 = 0$. L'erreur possible sur les y_i est $\|\delta y_1\| = \|\delta y_2\| = \frac{1}{2}\|\delta s_1 - \delta s_2\| \leq \varepsilon_p$, mais pour les autres points : $\|\delta y\| = \|\delta s - \delta O_s\| \leq 2\varepsilon_p$. L'angle θ_{max} est obtenu dans le cas de la figure 3.1 (pourvu que $\|m_1 - m_2\| > 2\varepsilon_p$).

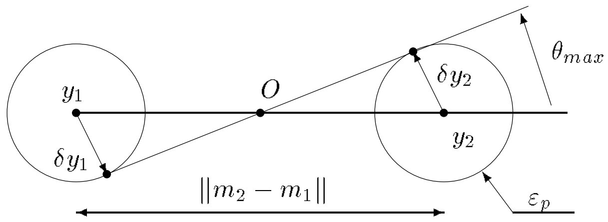


Figure 3.1 : Angle d'ajustement maximum pouvant être obtenu par la transformation aux moindres carrés.

On a alors trivialement $\sin(\theta_{max}) = \frac{2\varepsilon_p}{\|m_2 - m_1\|}$ d'où la borne exacte :

$$\|\delta A\| \leq \sqrt{2}\sqrt{1 - \sqrt{1 - \frac{4\varepsilon_p^2}{\|m_2 - m_1\|^2}}}$$

Ceci donne après un développement limité :

$$\varepsilon_{al} \leq 2\varepsilon_p \left(1 + \frac{\|x\|}{\|m_2 - m_1\|}\right) + O\left({\varepsilon_p}^3\right) \tag{3.5}$$

3.3.5 Modifications de l'algorithme et heuristique

L'existence d'un invariant supplémentaire va nous permettre de réduire considérablement la recherche des associations en utilisant tout simplement une table de hachage. En effet, si préalablement à la reconnaissance on prépare une table (unidimensionnelle) qui indexe toutes les bases modèle en fonction de leur longueur compatible (à plus ou moins $2\varepsilon_p$), il suffit lors de la reconnaissance d'examiner les bases images et en fonction de leur longueur, on récupère en temps constant les bases modèles pouvant convenir. On déroule ensuite normalement l'algorithme d'alignement, avec bien sûr la zone d'erreur prévue par la formule 3.5. Dans le pire des cas, toutes les bases modèles sont compatibles avec toutes les bases image et la complexité reste en $O(Mm^2n^2)$, mais nous verrons que l'analyse du chapitre 5 nous permettra de déterminer une complexité statistique inférieure.

La formule 3.5 fournit en même temps une heuristique pour guider la recherche des association : plus la longueur $\|m_2 - m_1\|$ de la base est grande, plus la zone d'erreur est faible. Or il est intuitif (mais cela sera démontré au chapitre 5) que plus l'erreur est faible et moins on a de chance de trouver des faux positifs : on dit que la base est plus sélective. On a donc intérêt à trier les bases possibles de l'image et à les examiner dans l'ordre décroissant de leur longueur. Cette heuristique est déjà connue et appliquée, mais elle trouve ici un fondement théorique.

3.4 Transformations rigides 3D

3.4.1 Invariants supplémentaires

Supposons l'appariement de 3 points modèles (m_1, m_2, m_3) avec 3 points de la scène $(\bar{s}_1, \bar{s}_2, \bar{s}_3)$; il ne peut être valide que si :

$$|\|\bar{s}_2 - \bar{s}_1\| - \|m_2 - m_1\|| \leq 2\varepsilon_p$$

$$|\|\bar{s}_3 - \bar{s}_2\| - \|m_3 - m_2\|| \leq 2\varepsilon_p$$

$$|\|\bar{s}_1 - \bar{s}_3\| - \|m_1 - m_3\|| \leq 2\varepsilon_p$$

On dispose donc de 3 invariants qui permettront de réduire drastiquement le nombre d'hypothèses d'appariement, par exemple grâce à une indexation : choisissant 3 points de la scène, on obtient directement les groupes de 3 points modèles qui peuvent être appariés. L'analyse de la probabilité de fausse association ou du nombre moyen d'hypothèses retenues entre dans le cadre de l'étude menée au chapitre 5.

3.4.2 Méthode de calcul de la transformation

De même que pour les transformations rigides 2D, il existe rarement une transformation exacte superposant les points (m_1, m_2, m_3) et $(\bar{s}_1, \bar{s}_2, \bar{s}_3)$. On choisit donc (aux moindres carrés) l'origine $\bar{O}_s$ et la rotation $\bar{A}$ qui minimisent le critère

$$C = \|\bar{s}_1 - O_s - A(m_1 - O_m)\|^2 + \|\bar{s}_2 - O_s - A(m_2 - O_m)\|^2 + \|\bar{s}_3 - O_s - A(m_3 - O_m)\|^2$$

Pour la translation, on montre facilement que le choix des origines aux barycentres convient, mais contrairement au cas 2D, on ne possède pas de solution sous forme analytique pour la rotation : on ne dispose que d'une méthode de résolution numérique en utilisant les quaternions (voir [9] pp 204-207). La méthode est développée dans l'appendice A.

3.4.3 Expression de δA

Comme en 2D, la décomposition $\bar{A} = \Gamma A$ permet de ramener le calcul de δA au cas 'simple' où $A = I_d$. On a alors $\delta A = \bar{A} - I_d = \sin\theta.\tilde{n} + (1 - \cos\theta)\tilde{n}^2$ en utilisant la formule de Rodrigues[1], ce qui permet d'écrire

$$\|\delta A x\| = \|(\sin\theta) n_\wedge x + (1 - \cos\theta) n_\wedge (n_\wedge x)\|$$

Or $(n_\wedge x)$ et $n_\wedge(n_\wedge x)$ sont orthogonaux d'où

$$\|\delta A x\|^2 = \sin^2\theta \|x\|^2 \sin^2(\widehat{n,x}) + (1-\cos\theta)^2 \|x\|^2 \sin^2(\widehat{n,x})$$

et par définition : $\|\delta A\| = \max_{\|x\|=1} \|\delta A x\|$ d'où

$$\|\delta A\| = \sqrt{2}\sqrt{1 - \cos\theta} \tag{3.6}$$

On remarquera que cette expression est identique au cas bidimensionnel.

3.4.4 Détermination de l'angle d'ajustement maximum θ_{max}

Puisqu'on est dans le cas particulier où $A = I_d$, on a $x_i = y_i$. On utilisera de préférence la notation y_i pour bien rappeler qu'on a éliminé la rotation A, sauf dans les cas d'invariants.

Comme dans le cas 2D, on travaille pour la rotation dans les repères barycentriques, mais l'erreur sur les y_i est un peu différente. Par exemple

$$\|\delta y_1\| = \|\frac{2}{3}\delta s_1 - \frac{1}{3}\delta s_2 - \frac{1}{3}\delta s_3\| \leq \frac{4}{3}\varepsilon_p$$

On calcule de même que

$$\|\delta y_i\| \leq \frac{4}{3}\varepsilon_p = \varepsilon_y$$

L'utilisation des repères barycentriques nous permet aussi de définir deux plans vectoriels orientés de $\mathbb{R}^3$ $\mathcal{P}$ et $\bar{\mathcal{P}}$, contenant les points y_i et $\bar{y}_i$, que l'on peut décrire par les vecteurs normaux $n = y_{1\wedge} y_2$ et $\bar{n} = \bar{y}_{1\wedge}\bar{y}_2$ (notations m et s dans l'appendice A).

Soit A_1 la rotation d'axe $n_\wedge \bar{n}$ et d'angle $\theta_1 = (\widehat{n,\bar{n}})$: elle superpose les deux plans $\mathcal{P}$ et $\bar{\mathcal{P}}$. Décomposons maintenant $\bar{A}$ en $\bar{A} = A_2 A_1$. D'après le corollaire 1 (Appendice A), A_2 est une rotation d'axe $\bar{n}$. Comme $\bar{A}$ est une rotation d'ajustement, on supposera que tous les angles sont faibles, ce qui permettra d'utiliser les développements limités à l'ordre 1 pour dériver des bornes sur les angles θ_1 et θ_2 de A_1 et A_2. On obtiendra finalement une borne sur θ.

[1] Voir appendice A

Borne sur l'angle de la rotation A_1

On note $r_1 = \frac{n_\wedge \bar{n}}{\|n_\wedge \bar{n}\|}$ l'axe de la rotation et $\theta_1 = (\widehat{n, \bar{n}})$ son angle. Remarquons d'abord que comme on est en repère barycentrique,

$$\bar{n} = \bar{y}_{1\wedge}\bar{y}_2 = \bar{y}_{2\wedge}\bar{y}_3 = \bar{y}_{3\wedge}\bar{y}_1$$

On peut donc écrire en développant $\bar{y}_i = y_i + \delta y_i$

$$\bar{n} = n + \frac{1}{3}\left((y_3 - y_2)_\wedge \delta y_1 + (y_1 - y_3)_\wedge \delta y_2 + (y_2 - y_1)_\wedge \delta y_3\right) + O({\varepsilon_y}^2)$$

$$\bar{n} = n + \delta n \quad \text{avec} \quad \|\delta n\| \leq \frac{\varepsilon_y}{3}\left(\sum_{i<j} \|x_i - x_j\|\right) + O({\varepsilon_y}^2)$$

Il est alors facile de voir que l'angle θ_1^{max} est obtenu dans la situation de la figure 3.2.
On tire facilement $\sin\theta_1^{max} = \varepsilon_y \frac{\sum_{i<j}\|x_i - x_j\|}{3\,\|x_{1\wedge} x_2\|} + O(\varepsilon_y^2)$ d'où :

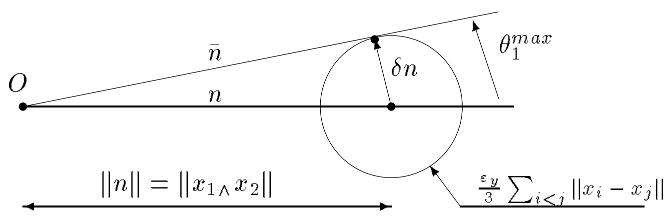


Figure 3.2 : Angle d'ajustement maximum pouvant être obtenu par la transformation aux moindres carrés.

$$\theta_1^{max} = \varepsilon_y \frac{\sum_{i<j}\|x_i - x_j\|}{3\,\|x_{1\wedge} x_2\|} + O(\varepsilon_y^2) \tag{3.7}$$

Borne sur l'angle θ_2 de la rotation A_2

L'axe de rotation est $\bar{n} = n + \delta n$ avec $\|\delta n\| \leq O(\varepsilon_y)$. On peut donc écrire le vecteur unitaire de l'axe

$$r_2 = \frac{\bar{n}}{\|\bar{n}\|} = \frac{n}{\|n\|} + O(\varepsilon_y)$$

Ceci permet de développer A_2 au premier ordre en supposant que θ_2 reste faible :

$$A_2 = I + \sin\theta_2 . \tilde{r}_2 + (1 - \cos\theta_2)\tilde{r}_2^2 = I + \theta_2 \tilde{r}_2 + O(\theta_2^2)$$

Comme on a la même chose pour A_1, on peut conclure

$$\bar{A} = A_2 A_1 = I + \theta_1 \tilde{r}_1 + \theta_2 \tilde{r}_2 + O(\theta^2) \tag{3.8}$$

en notant $O(\theta^2)$ les termes du second ordre et plus. Le critère à minimiser devient alors

$$C = \sum \|\delta y_i - \theta_1.r_{1\wedge} x_i - \theta_2.r_{2\wedge} x_i\|^2 + O(\theta^4)$$

Décomposons l'erreur δy_i en une composante $(\delta y_i)_x$ selon $y_i = x_i$, une composante $(\delta y_i)_t$ orthogonale à y_i mais qui reste dans le plan $\mathcal{P}$ et une composante $(\delta y_i)_n$ selon n, c'est-à-dire orthogonale à $\mathcal{P}$. En se rappelant que $r_1 = (n_\wedge \bar{n})/\|n_\wedge \bar{n}\|$ et $r_2 = n/\|n\| + O(\varepsilon_y)$, on scinde le critère :

$$C = \sum (\delta y_i)_x^2 + \sum \left((\delta y_i)_t - \theta_2\|x_i\|\right)^2 + \sum \left((\delta y_i)_n - \theta_1\|x_i\|\right)^2 + O({\varepsilon_y}^4)$$

Le minimum en θ_2 est obtenu en dérivant par rapport à cette variable

$$\frac{\partial C}{\partial \theta_2} = 2\sum \|x_i\| \left((\delta y_i)_t - \theta_2\|x_i\|\right) = 0 \quad \Leftrightarrow \quad \theta_2 = \frac{\sum \|x_i\|(\delta y_i)_t}{\sum \|x_i\|^2} + O({\varepsilon_y}^2)$$

On obtient donc la borne

$$\theta_2^{max} = \varepsilon_y \frac{\sum \|x_i\|}{\sum \|x_i\|^2} + O({\varepsilon_y}^2) \tag{3.9}$$

Borne sur δA

L'expression de $\bar{A}$ au premier ordre est donnée par 3.8. On a donc

$$\|\delta A\| = \|\bar{A} - I\| = \|\theta_1 \tilde{r}_1 + \theta_2 \tilde{r}_2\| + O(\theta^2)$$

Puisque r_1 et r_2 sont orthogonaux, on obtient directement

$$\|\delta A\| \leq \sqrt{(\theta_1^{max})^2 + (\theta_2^{max})^2} + O({\varepsilon_y}^2)$$

soit

$$\varepsilon_{al}(x) \leq 2\,\varepsilon_p \left(1 + \frac{2\|x\|}{3}\sqrt{\frac{\left(\sum_{i<j}\|x_i - x_j\|\right)^2}{9\,\|x_{1\wedge} x_2\|^2} + \frac{\left(\sum \|x_i\|\right)^2}{\left(\sum \|x_i\|^2\right)^2}}\right) + O\left({\varepsilon_p}^2\right) \tag{3.10}$$

3.4.5 Analyse et domaine de validité de la zone d'erreur

Par abus de langage, on appellera 'base' le triplet de points modèles (x_1, x_2, x_3) exprimé dans le repère barycentrique. Ce triplet forme un triangle dont les caractéristiques sont les suivantes :

- *Périmètre :* $p = \sum_{i<j} \|x_i - x_j\|$
- *Surface :* $\mathcal{S} = \frac{1}{2}\|(x_1 - x_2)_\wedge(x_1 - x_3)\| = \frac{3}{2}\|x_{1\wedge} x_2\|$

On a donc très simplement

$$\theta_1^{max} = \frac{p}{2\,\mathcal{S}}\varepsilon_y = \frac{\varepsilon_y}{r} = \frac{4\,\varepsilon_p}{3\,r}$$

en notant r le rayon du cercle inscrit dans le triangle [12]. La validité de cette borne est donnée par $\theta_1^{max} \ll 1$, soit

$$\varepsilon_y \ll r \quad \text{ou} \quad \varepsilon_p \ll r$$

Ceci veut donc dire que la taille du cercle inscrit dans la 'base' est un des paramètres à considérer pour effectuer notre heuristique de choix sur les bases. De plus, la formule n'est plus valable (mais on peut dire que la base non plus) lorsque les ordres de grandeur de ce rayon et de celui de l'erreur de mesure ε_p sont comparables.

En ce qui concerne θ_2^{max}, considérons $\|x_i\|$ comme la mesure d'une variable aléatoire, dont on notera $l = (\sum \|x_i\|)/3$ la moyenne et σ^2 la variance. La formule 3.9 s'écrit alors :

$$\theta_2^{max} = \varepsilon_y \frac{l}{\sigma^2 + l^2} + O(\varepsilon_y^2) \leq \frac{\varepsilon_y}{l} + O(\varepsilon_y^2)$$

La formule étant valable lorsque

$$\varepsilon_y \ll l \quad \text{ou} \quad \varepsilon_p \ll l$$

On remarquera que selon cette formule, le maximum d'erreur à moyenne l constante est obtenu pour $\sigma = 0$, c'est-à-dire pour le triangle équilatéral puisqu'on est en repère barycentrique. Ce résultat contre-intuitif doit être tempéré par le fait que d'une part la moyenne des distances au barycentre n'est pas un critère de recherche convenable car il nécessite de connaître déjà les trois points, et d'autre part les simulations numériques montrent que le terme d'erreur θ_2^{max} reste borné et a une influence très faible sur l'erreur totale.

3.4.6 Validation statistique de la borne sur l'erreur

Afin de pouvoir appliquer correctement la formule 3.10 dans les algorithmes, on a besoin de connaître la précision de celle-ci par rapport aux erreurs réelles. On se focalisera ici sur l'erreur due à la rotation : étant donné l'appariement de deux triplets de points en repère barycentrique (x_1, x_2, x_3) et $(\bar{y}_1, \bar{y}_2, \bar{y}_3)$, on recherche la rotation $\bar{A}$ qui minimise l'erreur de superposition aux moindres carrés. L'erreur réelle sur la prévision du correspondant du point x est alors $\|\delta A.x\| = \alpha\|x\|$ puisque nous avons vu en 3.4.3 que ceci est linéaire en $\|x\|$. C'est donc le facteur d'erreur $\alpha = \frac{\|\delta A.x\|}{\|x\|}$ que nous allons tenter de caractériser par rapport au facteur d'erreur prévu sur la rotation $\alpha_{prev} = \frac{4}{3}\theta_{max}$. Observons α et α_{prev} : ces deux variables dépendent du trièdre de base et de la borne sur l'erreur de mesure ε_p, mais α dépend en plus des erreurs de mesure réelles δy_i sur les points du trièdre image et de la direction de x, point dont on doit prévoir le correspondant. Le principe des statistiques présentées en figure 3.3 est donc le suivant : on choisit au hasard 3 points pour former le trièdre de base, et une borne ε_p sur l'erreur de mesure. Ceci permet déjà de calculer l'abscisse x : le facteur d'erreur prévu. On perturbe ensuite le trièdre un certain nombre de fois (1000 dans notre cas); pour chacune

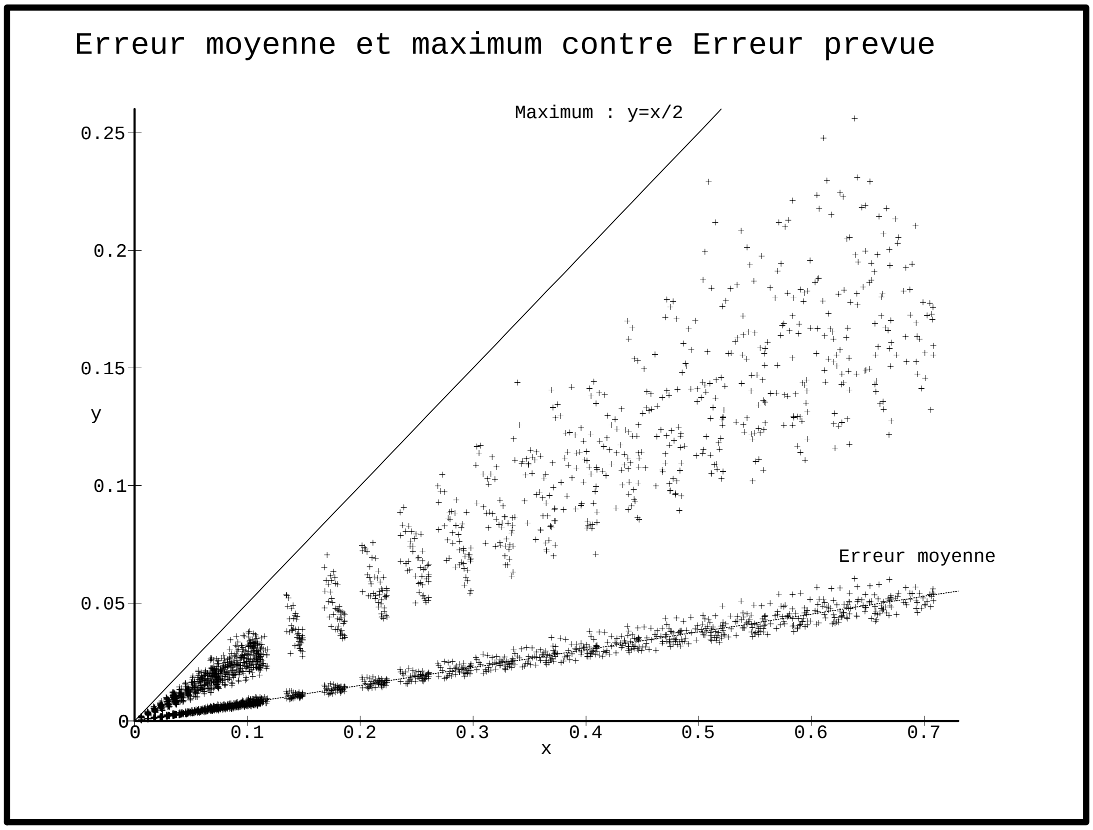


Figure 3.3 : Moyenne et maximum des facteurs d'erreur réels en fonction du facteur d'erreur prévu.

des perturbations on calcule la transformation δA entre les deux trièdres et on échantillonne le facteur d'erreur en calculant l'erreur sur un grand nombre d'x (ici 500). On obtient donc pour un couple (trièdre,ε_p) un ensemble de 500 000 mesures du facteur d'erreur : on a choisi de tracer la moyenne et le maximum de cette population. Il reste à varier la forme du trièdre et son échelle par rapport à l'erreur de mesure, ce qui est réalisé en choisissant 80 trièdres aléatoirement, et pour chacun de ces trièdres, 20 valeurs de ε_p. On obtient en fin de compte 1600 points représentatifs du maximum et de la moyenne du facteur d'erreur.

Les points représentant les maxima du facteur d'erreur sont relativement dispersés. On observera cependant l'encadrement

$$\frac{\alpha_{prev}}{5.5} \leq \alpha_{max} \leq \frac{\alpha_{prev}}{2}$$

De plus, le maximum des α_{max} est obtenu pour $\alpha_{prev}/2.2$, ce qui montre bien que la borne que nous avons trouvé est relativement serrée. En ce qui concerne la moyenne, on peut observer une répartition très groupée autour de la droite de régression $y = \frac{x}{13.227}$. Il semble donc le facteur d'erreur prévu soit, à un coefficient près, un des paramètres de la loi de α. Afin de vérifier si la forme du trièdre de base influe aussi sur cette loi, on a tracé les densités de probabilité empiriques de cette loi pour trois trièdres différents (voir fig. 3.4).

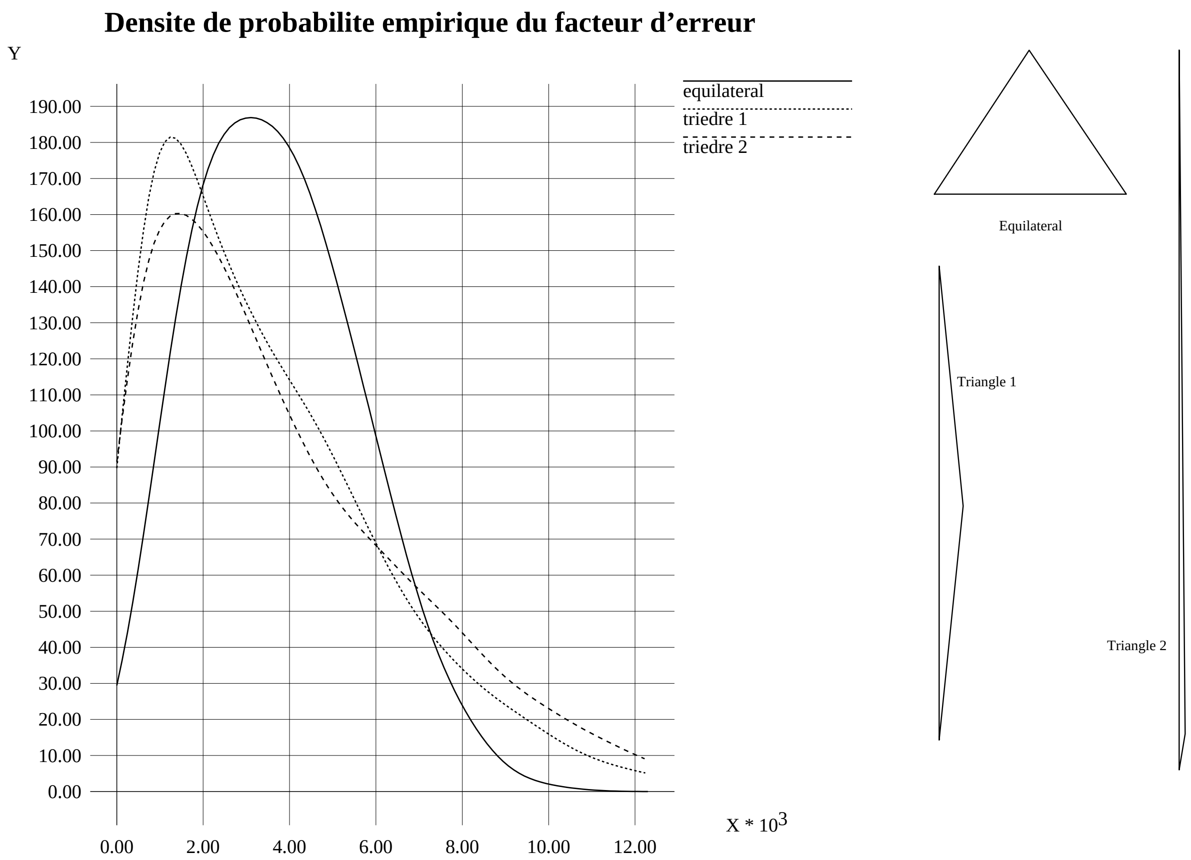


Figure 3.4 : Densité de probabilité empirique du facteur d'erreur réel pour trois triangles de forme différente. Pour chacun des triangles, on calcule l'homothétie nécessaire pour que l'erreur prévue soit de 0.05 et on applique la méthode de convolution par un noyau gaussien pour obtenir des courbes a peu près régulières.

On s'aperçoit en fait que si le mode de la distribution a une valeur plus grande pour le triangle équilatéral, la queue de la distribution est pour sa part moins allongée. La position du mode s'explique par le fait qu'après l'homothétie qui ramène l'erreur prévue à 0.05, les triangles 1 et 2 sont beaucoup plus grands que le triangle équilatéral : la plupart des erreurs ont donc une valeur relativement faible, mais de grosses erreurs peuvent toujours se produire. Au contraire, pour le triangle équilatéral, toutes les erreurs sont concentrées autour du mode, d'ailleurs peu différent de la moyenne.

3.4.7 Utilisation pratique de la borne sur l'erreur

La méthode que nous avons développée jusqu'ici vise l'élimination de tous les faux négatifs en n'oubliant aucun point pouvant être apparié. L'inconvénient majeur de la prise en compte d'une borne sur l'erreur est la croissance rapide du nombre de faux positifs avec la taille de la zone d'erreur par appariement de mauvais points situés par hasard dans une zone d'erreur. Au niveau de l'utilisation, il y a donc un équilibre à réaliser entre les deux tendances. Le fait qu'il y ait un facteur de $2,2$ entre la borne statistique trouvée sur l'erreur (voir 3.3) et la borne prévue suggère l'application d'un coefficient d'ajustement sur la taille de la zone d'erreur prévue. On a tracé à cette fin un abaque statistique de la proportion des points réels inclus et exclus de la zone d'erreur pondérée (Fig. 3.5).

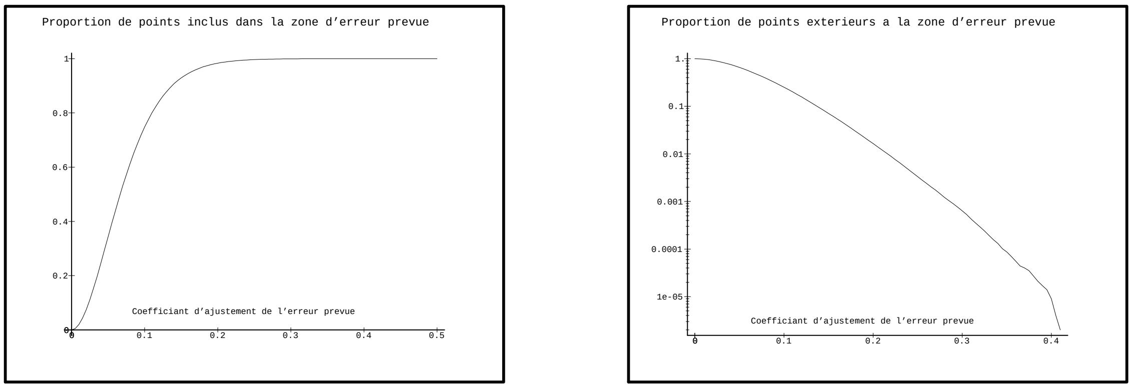


Figure 3.5 : Proportion de points réels soumis à l'erreur inclus (à gauche) et exclus (à droite) de la zone d'erreur pondérée par le coefficient en abscisse. Les statistiques sont réaliséees avec 200 trièdres aléatoires, 500 perturbations par trièdre et 200 points par perturbation.

Si l'on recherche une reconnaissance à 50% des points du modèle, on peut ainsi se permettre dans notre cas 'd'oublier' un point sur 100, ce qui équivaut à une reconnaissance à 49%, et gagner ainsi un facteur 5 environ sur la taille de l'erreur à prendre en compte, ce qui conduit à un nombre de faux positifs très nettement réduit (voir chapitre 5).

3.4.8 Dérivation d'une heuristique

Le problème que l'on se pose maintenant est de classer les $C_s^3 = O(s^3)$ trièdres possibles de la scène par ordre croissant d'erreur. En effet, nous verrons au chapitre 5 que plus l'erreur est faible, moins on a de chance de se tromper.
Puisque l'analyse théorique de la formule 3.10 n'a pas permis de résoudre ce problème, on est réduit à le faire numériquement. Supposons deux points X_1 et X_2 du triangle connus, formant le plus grand des côtés. Le problème est de savoir où choisir X_3 en priorité. Dans la figure 3.6, on a tracé la surface des erreurs θ_1^{max}, θ_2^{max} et θ^{max} en fonction de la position de X_3, l'unité de longueur étant $\frac{\|X_2-X_1\|}{\varepsilon_y} = \frac{3\|X_2-X_1\|}{4\varepsilon_y}$.

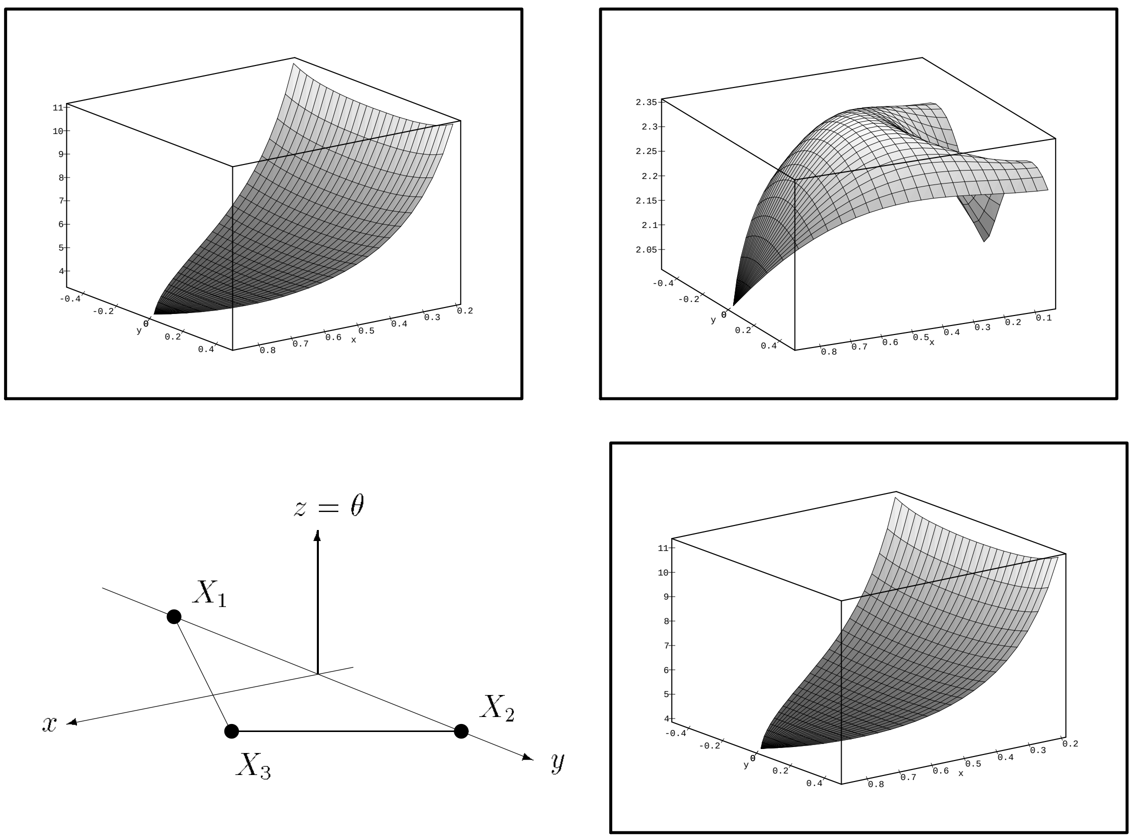


Figure 3.6 : *Erreur maximum sur les angles.* En haut de gauche à droite θ_1^{max} et θ_2^{max}, en bas la position de X_1 et X_2 par rapport aux axes, et l'erreur totale θ^{max}.

Regardons tout d'abord θ_2^{max} : on trouve en fait deux minimums de l'erreur : l'un est obtenu pour $X_3 = (X_1 + X_2)/2$, et l'autre pour le triangle équilatéral. Si le premier minimum parait curieux, le second est tout à fait satisfaisant et contredit l'interprétation 'statistique' de la formule 3.9. La seconde remarque concerne les valeurs prises par ce terme : elles varient de 2 environ aux minimums à 2.35 au maximum (l'unité étant la longueur du plus grand côté). Ce terme est en fait pratiquement constant. Ceci se retrouve bien dans l'analogie des surfaces de θ_1^{max} et θ^{max}, surfaces entre lesquelles on ne peut déceler à l'oeil qu'une translation verticale. L'analyse de θ^{max} est donc à peu de chose près celle de θ_1^{max} : on observe en particulier que la variation principale de l'erreur s'effectue selon l'axe x, c'est-à-dire selon la distance de X_3 à la droite définie par X_1 et X_2, les variations selon l'axe y, parallèlement à la droite (X_1, X_2) étant pratiquement négligeables.

On a donc mis en évidence deux paramètres : pour choisir deux points, on doit maximiser la distance entre ces deux points, et pour choisir le troisième, il faut s'éloigner au maximum de la droite définie par les deux premiers. Il reste que cela ne nous permet pas de comparer une base relativement aplatie mais un grand côté important et une base plus petite mais plus équilatérale : ces deux paramètres ne nous donnent qu'un ordre partiel sur les trièdres.

3.4.9 Modifications de l'algorithme

La méthode brutale consiste à indexer toutes les bases possibles des modèles dans une table de hachage en fonction des invariants $\|x_1 - x_2\|$, $\|x_2 - x_3\|$ et $\|x_3 - x_1\|$, puis lors de la reconnaissance, on énumère toutes les bases possibles de la scène, et on les classe par erreur θ^{max} croissante. Il suffit ensuite dans cet ordre de récupérer grâce à la table de hachage les bases modèles pouvant convenir et de dérouler l'algorithme de vérification avec la zone d'erreur prévue par la formule 3.10. Cette méthode a une complexité de $O(n^4 \log n)$ dans le pire des cas (contre $O(n^4)$ pour l'algorithme normal), mais elle devrait avoir une complexité moyenne bien meilleure. Elle nécessite cependant $O(n^3)$ en stockage pour la liste triée des bases de la scène. Il peut donc être avantageux de se contenter de l'heuristique définie précédemment: pour chaque point de la scène on recherche le second point par distance décroissante, puis le troisième le plus éloigné possible de la droite.

3.5 Résumé

Nous avons montré qu'en supposant une borne ε_p sur l'erreur de mesure des points images, le rayon de la zone de recherche autour du point x est, sous des hypothèses raisonnables, majoré par

$$\varepsilon_{al} \leq \varepsilon_p \; (2 + \mathcal{K}_A \|x\|) + O(\varepsilon_p^2)$$

où $\mathcal{K}_A$ dépend de la transformation que l'on doit trouver. Les cas simples des transformations affines et rigides 2D ayant permis de présenter la méthode de calcul de la zone de recherche, nous nous sommes ensuite concentrés sur le problème plus complexe des transformations rigides 3D : 3 points images et modèles sont nécessaires pour définir une telle transformation, et ils procurent 3 invariants supplémentaires qui restreignent le choix des associations de bases possibles. Ayant choisi 3 points images constituant une 'base', on pourra ainsi retrouver grâce à une table de hachage sur ces invariants les bases modèles compatibles. Pour chacune de celles-ci, on calcule aux moindres carrés la transformation rigide entre les deux triplets de points (en passant par les quaternions, voir Annexe A) et on recherche dans l'image les points susceptibles d'être appariés aux points du modèle. Pour cela, on projette le modèle dans l'image, et on a montré que la zone de recherche autour des points prévus doit être théoriquement de rayon ε_{al}

$$\varepsilon_{al}(x) \simeq 2\,\varepsilon_p \left(1 + \frac{2\|x\|}{3} \sqrt{\frac{\left(\sum_{i<j} \|x_i - x_j\|\right)^2}{3\,\|x_1 \wedge x_2\|^2} + \frac{\left(\sum \|x_i\|\right)^2}{\left(\sum \|x_i\|^2\right)^2}} \right) + O\left({\varepsilon_p}^2\right)$$

Les statistiques ont ensuite validé la qualité de cette borne puisque pratiquement, la borne réelle est obtenue à $\frac{2}{5}\varepsilon_{al}$ et l'erreur moyenne à $\frac{\varepsilon_{al}}{13.2}$. L'analyse de cette formule nous a alors permis de dériver une heuristique guidant l'ordre de choix des "bases" image.

L'intégration du traitement de l'erreur dans l'alignement fournit donc plus qu'un cadre théorique : un algorithme parfaitement utilisable. S'il est vrai que l'analyse de l'erreur est un peu complexe, l'utilisation en est par contre très simple et naturelle dans ce type d'algorithme.

Chapitre 4

Hachage : Zone compatible dans l'espace des invariants

On peut différencier deux techniques principales de hachage. Le hachage simple se propose de retrouver les objets présents dans la scène grâce aux invariants. Pour cela, il suffit de préparer une table de hachage discrétisant l'espace des invariants dans laquelle on indexe les objets à partir de leurs invariants. Lors de la reconnaissance, on calcule les invariants présents dans la scène, et pour chacun d'entre eux la table de hachage donne l'accès direct aux objets les possédant.

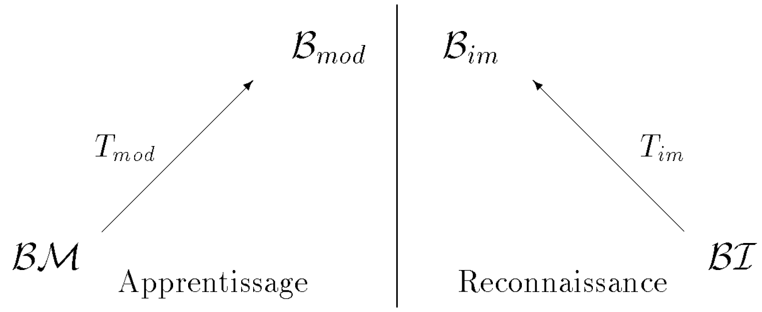


Figure 4.1 : Les coordonnées des primitives ne constituant pas la base sont identiques dans $\mathcal{B}_{mod}$ et $\mathcal{B}_{im}$ si ces deux bases sont en correspondance.

Le hachage géométrique est plus ambitieux puisqu'il permet également de retrouver le positionnement de l'objet. Il repose sur le principe suivant : supposons qu'avec k primitives du modèle on puisse définir une base $\mathcal{B}_{mod}$. On peut définir de la même façon une base image $\mathcal{B}_{im}$ avec k primitives de la scène. Si maintenant les primitives choisies pour former ces bases se correspondent 2 à 2 entre modèle et scène, les bases $\mathcal{B}_{mod}$ et $\mathcal{B}_{im}$ se correspondent aussi, ce qui détermine une transformation unique envoyant le modèle dans la scène, et les

coordonnées des autres primitives du modèle présentes dans la scène sont identiques dans $\mathcal{B}_{mod}$ et $\mathcal{B}_{im}$: ce sont des invariants (voir fig. 4.1).
Il suffit donc de préparer dans la table de hachage l'indexation des couples (modèle,base) par les coordonnées des autres primitives du modèle, et lors de la reconnaissance, pour chaque base examinée on vote pour le couple possédant une primitive à l'endroit défini par les coordonnées des primitives de la scène. Le vote comptabilise le nombre de primitives appariées, ce qui fournit directement l'estimation de la qualité de l'appariement de la base scène choisie avec les différents couples (modèle, base) possibles.
Le point important sur lequel repose le hachage est la séparation des calculs d'invariants entre le modèle et la scène. En particulier, on ne peut plus comme pour l'alignement calculer directement la transformation amenant le modèle dans la scène (aux moindres carrés par exemple) : il faut impérativement passer par la définition d'une base modèle $\mathcal{B}_{mod}$ et d'une base image $\mathcal{B}_{im}$.

Plaçons nous maintenant dans le cas d'un sous-ensemble des transformations affines concernant des points et supposons définies ces bases. La définition de $\mathcal{B}_{im}$ fournit directement la matrice de passage de $\mathcal{B}_{im}$ à $\mathcal{BI}$ sous la forme $A = [e_1, \dots e_n]$ et l'origine O_s de $\mathcal{B}_{im}$. Lors de la reconnaissance, on a les coordonnées des points images s_i dans $\mathcal{BI}$ et on veut obtenir leurs coordonnées x_i dans $\mathcal{B}_{im}$ afin de voter pour les couples (modèle,base) possédant ces points. Ceci revient à résoudre $Ax_i = s_i - O_s$. On doit bien évidemment faire la même chose lors de l'apprentissage pour préparer la table de hachage, mais les points du modèle étant supposés être exacts, la résolution est immédiate. Les points images étant eux soumis à l'erreur, le système devient

$$(A + \delta A)(x + \delta x) = s + \delta s - O_s - \delta O_s$$

Il nous faut trouver une borne sur $\|\delta x\|$ afin de n'oublier lors du vote aucun point pouvant être correct. On obtient facilement :

$$\delta x = (A + \delta A)^{-1} (\delta s - \delta O_s - \delta A.x)$$

soit :

$$\varepsilon_{gh} = \max \|\delta x\| \leq \|\bar{A}^{-1}\| \; \|\delta y - \delta A.x\| = \|\bar{A}^{-1}\| \; \varepsilon_{al}(x) \tag{4.1}$$

La zone d'erreur dans l'espace des invariants est donc simplement la transformée de la zone de recherche pour l'alignement. Il est à noter que le ε_{al} employé ici est celui obtenu en passant par la définition des bases et non pas directement par la résolution aux moindres carrés.

4.1 Traitement de l'erreur

Dans l'algorithme idéal, voter pour un point signifie voter pour les bases indexées par *une* case de la table de hachage. Avec l'introduction de l'erreur, il est plus que probable que la zone compatible avec un point dépasse les limites d'une case. Il va donc falloir voter avec toutes les cases intersectant cette zone. Il y a deux possibilités pour réaliser ceci : on peut tout d'abord conserver la table de hachage idéale et voter lors de la reconnaissance avec toutes les cases intersectées par la zone compatible ; cette approche est celle de [2, 5]. L'inconvénient majeur de cette méthode est l'accroissement de la complexité lors de la reconnaissance puisqu'il y a un vote multiple par point image et qu'il faut de plus calculer les cases de la table intersectées par la zone compatible. On conserve par contre la complexité en place de la table de hachage.

Afin de pallier à ces défauts, on peut reporter le traitement de l'erreur dans le prétraitement (phase d'apprentissage) et prendre en compte l'erreur directement dans la table de hachage en indexant pour chaque point les cases intersectées par la zone compatible [4]. Il ne reste plus lors de la reconnaissance qu'à voter ponctuellement comme pour l'algorithme idéal. On conserve ici la complexité de la phase de reconnaissance. Ce type de traitement, s'il apparaît être le meilleur, n'est toutefois pas toujours possible (voir 4.2.1).

4.2 Zone de compatibilité

Les invariants que nous aurons à traiter par le hachage simple étant déjà pourvus de leur zone d'erreur, on ne s'occupera ici que du hachage géométrique.

4.2.1 Transformations affines

Rappelons que si n est la dimension de l'espace considéré, $n+1$ points en situation générale sont nécessaires pour définir une base affine : le premier point constitue l'origine, et les n autres points sont les extrémités des vecteurs de base. On a alors d'après 3.3

$$\|\delta y - \delta A.x\| \leq \varepsilon_{al} \leq 2\varepsilon_p\,(1 + \|x\|_1)$$

d'où

$$\varepsilon_{gh} = 2\|\bar{A}^{-1}\|\varepsilon_p(1 + \|x\|_1) \tag{4.2}$$

Cette formulation de la zone d'erreur montre clairement que l'on ne peut pas ici traiter l'erreur lors de l'apprentissage puisque la zone de compatibilité dépend de la transformation image $\bar{A}$, donc de la base image choisie. Une solution à ce problème est de ne considérer que les transformations vérifiant $\|\bar{A}^{-1}\| \leq \mathcal{S}$ en arguant du fait que les bases ne vérifiant pas ce critère sont celles dont les vecteurs de base sont trop courts ou presque colinéaires. On a alors :

$$\|\delta x\| \leq \mathcal{S}\varepsilon_{al}(x)$$

Il paraît toutefois peu judicieux de préparer la table de hachage avec une borne aussi large sur l'erreur: il vaut encore mieux calculer $\|\bar{A}^{-1}\|$ lors de la reconnaissance.

La méthode que nous proposons maintenant permet une prise en compte de l'erreur lors de l'apprentissage : il s'agit de rajouter une dimension à la table de hachage (et ceci aux dépens du coût de stockage de cette table), laquelle indexe $\|\bar{A}^{-1}\|$. On peut alors indexer les baquets concernés du plan $\|\bar{A}^{-1}\| = C^{te}$ en utilisant comme valeur de l'erreur la formule 4.2. Le pas de discrétisation selon cette nouvelle dimension doit être adapté à la distribution des valeurs de $\|\bar{A}^{-1}\|$ de façon à ce que la probabilité de se trouver dans chacun des plans soit équiprobable. Il faut bien noter qu'il ne s'agit pas d'une dimension discriminante, mais simplement d'une indexation vers la table de hachage qui possède un traitement de l'erreur adaptée : chaque couple (modèle, base) indexé dans la table normale par un point est ici indexé par une droite parallèle à l'axe $\|\bar{A}^{-1}\|$ (et en fait par un cône puisque le nombre de buckets concernés augmente au long de cet axe).

4.2.2 Transformations rigides 2D

Nous savons déjà grâce à 3.3 que deux points sont nécessaires pour définir une base et que la distance entre ces deux points est un invariant supplémentaire ayant une zone d'erreur de rayon $2\varepsilon_p$. Pour l'alignement, nous avions choisi de calculer la transformation aux moindres carrés entre les 2 points modèles et les 2 points images. Ceci revenait à fixer l'origine des bases translatées aux barycentres, et choisir la rotation qui aligne (x_1, x_2) et $(\bar{y}_1, \bar{y}_2)$. La zone de recherche était alors caractérisée par un rayon

$$\varepsilon_{al} \leq 2\varepsilon_p \left(1 + \frac{\|x\|}{\|m_2 - m_1\|}\right) + O\left({\varepsilon_p}^2\right)$$

Pour le hachage géométrique, il nous faut passer par la définition d'une base modèle $\mathcal{B}_{mod}$ et d'une base image $\mathcal{B}_{im}$. On s'aperçoit facilement que si l'on choisit comme origine le barycentre et comme axe principal l'axe des deux points on obtiendra une transformation identique à la transformation aux moindres carrés. En formalisant ceci, la fonction qui définit une base à partir de deux points s'écrit :

$$\mathcal{B}(z,t) = \left(O = \frac{z+t}{2} \ ; \ e_1 = \frac{t-z}{\|t-z\|} \ ; \ e_2 = \frac{(t-z)^\perp}{\|t-z\|}\right)$$

où le signe $\perp$ signifie orthogonal dans le sens direct.
On peut donc utiliser la borne ε_{al} sur l'alignement et comme pour une transformation rigide $\|\bar{A}\| = \|\bar{A}^{-1}\| = 1$, on obtient

$$\varepsilon_{gh} \leq 2\varepsilon_p \left(1 + \frac{\|x\|}{\|m_2 - m_1\|}\right) + O\left({\varepsilon_p}^2\right) \tag{4.3}$$

Puisque toutes les variables dans la formule 4.3 ne dépendent que du modèle, il n'y a cette fois ci aucun problème pour préparer la table de hachage avec l'erreur lors de l'apprentissage. Par contre, il faut prendre en compte l'invariant supplémentaire de 'longueur de base', ce qui amène à une table de hachage de dimension 3.

4.2.3 Transformations rigides 3D

Définition de la base

Il faut trois points pour définir une base, et cela fournit 3 invariants supplémentaires, par exemple la distance entre chacun des points pris deux à deux (3.4). Il reste à trouver une

méthode pour définir une base à partir de ces trois points, et l'on voudrait si possible que, comme dans le cas 2D, la superposition des bases modèles et scène corresponde à la transformation aux moindres carrés. Le choix de l'origine au barycentre et d'un axe orthogonal au plan des trois points est évident. En effet, la translation aux moindres carrés amène le barycentre des points modèle sur celui des points image, et d'après le corollaire 1 de l'appendice A, la rotation aux moindres carrés amène les axes orthogonaux aux deux triplets de points l'un sur l'autre. Par contre, le choix des deux derniers axes dans le plan des trois points est un problème non résolu à l'heure actuelle (et sans doute non soluble).
On doit donc choisir d'autres axes de telle sorte que l'erreur finale soit la plus faible possible. Il apparaît que la direction du plus grand des côtés du trièdre formé par les points est relativement stable, le dernier axe étant pris de façon à compléter le trièdre direct formant la base. La fonction définissant une base à partir de trois points ordonnés de telle sorte que $\|b-a\| \geq \|c-b\| \geq \|c-a\|$ est la suivante

$$\mathcal{B}(a,b,c) = \left(O = \frac{a+b+c}{3} \; ; \; e_1 = \frac{(a-O)_\wedge(b-O)}{\|(a-O)_\wedge(b-O)\|} \; ; \; e_2 = \frac{b-a}{\|b-a\|} \; ; \; e_3 = e_{1\wedge}e_2 \right)$$

Lors de la préparation de la table de hachage, on fera toutefois attention que si deux voire trois côtés ont des longueurs compatibles, i.e. ont des longueurs égales à $2\varepsilon_p$ près, il y a deux ou trois ordonnancements possibles des points, c'est à dire deux ou trois bases différentes à indexer. Cette précaution permettra de ne pas se poser ces questions lors de la reconnaissance.

Dérivation de la borne sur l'erreur

Il nous faut reprendre le calcul de ε_{al} puisque la méthode utilisée n'est plus la même qu'au §3.4.4. Cependant, la majeure partie du calcul est identique, seul le calcul de l'angle maximum $\theta_{2'}^{max}$ de la rotation $A_{2'}$ est différent. En effet, l'alignement des axes orthogonaux aux plans des points modèles et images est équivalent à la rotation A_1, et les autres considérations ne dépendent pas de la méthode.

Rappelons que l'angle de la rotation $A_{2'}$ est donné directement par l'angle entre les vecteurs $e_2(y)$ et $e_2(\bar{y})$, c'est à dire selon les notations du §3.4.4 entre $y_2 - y_1 = s_2 - s_1$ et $\bar{y}_2 - \bar{y}_1 = s_2 - s_1 + \delta s_2 - \delta s_1$. L'angle maximum se calcule donc très simplement d'après la figure 4.2 :

$$\theta_{2'}^{max} = \frac{2\varepsilon_p}{\|x_2 - x_1\|} + O(\varepsilon_p^2) = \frac{2\varepsilon_p}{\max_{i,j} \|x_i - x_j\|} + O(\varepsilon_p^2)$$

La borne sur δA étant toujours la même, on tire

$$\|\delta A\| \leq \sqrt{(\theta_1^{max})^2 + (\theta_{2'}^{max})^2} + O({\varepsilon_p}^2)$$

Soit puisque $\|\bar{A}^{-1}\| = 1$

$$\varepsilon_{gh}(x) \leq 2\,\varepsilon_p \left(1 + \|x\| \sqrt{\frac{\left(4\, \sum_{i<j} \|x_i - x_j\|\right)^2}{9\,\|x_{1\wedge}x_2\|^2} + \frac{1}{\max_{i,j} \|x_i - x_j\|^2}} \right) + O\left({\varepsilon_p}^2\right) \qquad (4.4)$$

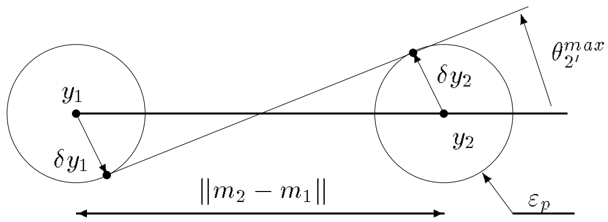


Figure 4.2 : Angle d'ajustement maximum $\theta_{2'}^{max}$ pouvant être obtenu.

Comparaison des méthodes : moindres carrés contre définition de bases

Le seul terme qui change dans la formulation de l'erreur est θ_2^{max} que l'on remplace par $\theta_{2'}^{max}$. On peut donc se contenter de comparer les valeurs de ces termes. Reprenons pour cela le principe de la figure 3.6 : on trace θ_2^{max} et $\theta_{2'}^{max}$ en fonction de la position du point X_3, le plus grand côté étant $[X_1, X_2]$, et l'unité de mesure $\frac{\|X_2 - X_1\|}{\varepsilon_p}$ (figure 4.3).

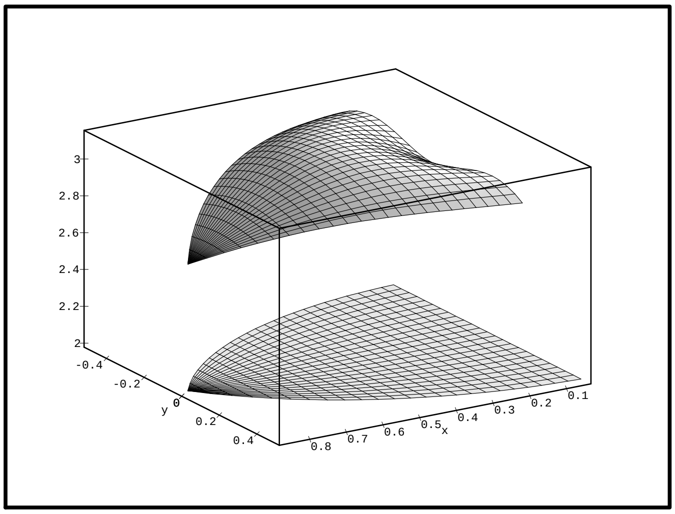


Figure 4.3 : Erreur maximum sur les angles θ_2^{max} et $\theta_{2'}^{max}$. En haut θ_2^{max}, et en bas $\theta_{2'}^{max}$.

En fait, $\theta_{2'}^{max}$ est à un facteur 2 près l'unité de mesure, et on s'aperçoit que la méthode par définition de base est meilleure. On a toutefois l'encadrement

$$\theta_{2'}^{max} \leq \theta_2^{max} \leq \frac{3}{2}\theta_{2'}^{max}$$

et quand on se rappelle l'influence minime du terme θ_2^{max} par rapport à θ_1^{max}, on se rend compte que les deux méthodes sont très proches.

Algorithme

Comme en deux dimensions, on doit inclure les invariants supplémentaires dans la table de hachage, ce qui produit ici une table de dimension 6, et il n'y a aucun problème pour traiter l'erreur lors de la préparation de cette table. On ne doit cependant pas oublier d'indexer les différentes bases possibles avec un même trièdre si plusieurs côtés sont compatibles.

4.3 Discrétisation de l'espace des invariants

4.3.1 Hachage simple

Les invariants que nous avons à traiter en hachage simple sont la 'longueur de base' dans le cas des transformations rigides 2D ou la longueur des côtés du trièdre de base dans le cas 3D. Tous ces invariants ont une borne fixe sur l'erreur, en l'occurrence $\varepsilon_l = 2\varepsilon_p$. Le problème que l'on se pose ici est de trouver une discrétisation adaptée pour la table de hachage. Les invariants étant indépendants, nous nous contenterons d'étudier le cas d'un seul invariant l, la table possédant alors une seule dimension. Il suffit pour plusieurs invariants de prendre le produit cartésien des tables de hachage.

Soit $L = l_{max} - l_{min}$ la taille de la table de hachage et supposons une discrétisation constante de pas h_l. La longueur compatible avec un invariant étant de diamètre $2\varepsilon_l$, le nombre n_l de cases intersectées par la zone d'erreur est dans notre cas

$$\left\lceil \frac{2\,\varepsilon_l}{h_l} \right\rceil \leq n_l \leq \left\lceil \frac{2\,\varepsilon_l}{h_l} \right\rceil + 1$$

En fait, le paramètre qui nous intéresse plus spécifiquement est l'étalement d'un invariant, c'est à dire la proportion de la table de hachage qui indexe cet invariant : plus l'étalement est faible, moins on a de chances de voter pour celui-ci avec un point pris au hasard. C'est en quelque sorte l'opposé de la sélectivité. On notera par la suite μ ce paramètre.
L'étalement minimal est obtenu en considérant une table de hachage continue : $\mu_l^0 = \frac{2\,\varepsilon_l}{L}$. Avec la discrétisation de la table, on obtient

$$\frac{h_l}{L} \left\lceil \frac{2\,\varepsilon_l}{h_l} \right\rceil \leq \mu_l = \frac{n_l h_l}{L} \leq \frac{h_l}{L} \left(1 + \left\lceil \frac{2\,\varepsilon_l}{h_l} \right\rceil \right)$$

ce qui s'encadre par

$$\mu_l^0 \leq \mu_l \leq \mu_l^0 + 2\frac{h_l}{L}$$

Ces formules appellent plusieurs remarques :

- Si le pas de hachage est grand devant le diamètre de la zone d'erreur $h_l \geq 2\,\varepsilon_l$, alors on a $\mu_l \leq 2\frac{h_l}{L}$. Cette majoration est irréductible car on peut toujours placer l'erreur au bord d'une case et impliquer ainsi deux cases dans l'indexation. L'avantage est une place mémoire relativement faible pour la table, et un nombre de cases à indexer par invariant plafonné à deux. La contrepartie est un étalement non minimal, donc un nombre sans doute élevé de faux positifs.

- Si le pas de hachage est très faible devant l'erreur $h_l \ll \varepsilon_l$, on a $\mu_l \simeq \mu_l^0$. L'effet de la discrétisation est amorti au niveau de l'étalement, mais le nombre d'indexations par invariant peut être élevé et surtout la complexité en place de la table est accrue.
- L'algorithme idéal utilise une case par indexation, donc $\mu_l = \frac{h_l}{L}$. Ceci montre que si l'on ne prend pas en compte l'erreur dans l'algorithme, non seulement des faux négatifs peuvent se produire, mais aussi des faux positifs !

Il y a donc un compromis à faire entre la place mémoire de la table et la perte de sélectivité. En fait, les gains sur l'étalement devenant faibles pour h_l décroissant en dessous de $\varepsilon_l/2$ ou $\varepsilon_l/4$, il semble que le choix de $h_l \approx \varepsilon_l$ soit équilibré. Toutefois, dans le cas où l'on traite l'erreur à la reconnaissance, le choix de $h_l = 2\varepsilon_l$ permet de déterminer très simplement la deuxième case à faire voter par un test de position de l'invariant image par rapport au centre de la case où il se trouve.

4.3.2 Hachage géométrique

En reprenant la formule générique 3.2 on peut écrire

$$\varepsilon_{gh}(x) = \varepsilon_p(2 + \mathcal{K}_A\|x\|)\|\bar{A}^{-1}\| + O(\varepsilon_p^2)$$

Cette formulation englobe tous les cas que nous avons vus. La table de hachage n'est plus ici unidimensionnelle : soit n sa dimension et $\mathcal{V}$ son volume. Le volume compatible avec un point de l'espace est une sphère centrée en ce point de diamètre $2\,\varepsilon_{gh}$, que l'on peut envelopper dans un pavé de côté $2\,\varepsilon_{gh}$.
On voudrait obtenir comme dans le cas du hachage simple une taille des cases de l'ordre de quelques fractions du volume de la sphère ou du pavé d'erreur. Le problème majeur est que ε_{gh} dépend de $\|x\|$ et que ceci n'est pas séparable selon chacune des coordonnées : si le pas de hachage est indépendant selon chaque axe, les cases proches d'un axe auront inévitablement une taille plus faible que les cases éloignées de tous les axes, même à norme $\|x\|$ égale. Il faudrait une discrétisation telle que la taille des cases dépende uniquement de $\|x\|$. Comme nous ne savons pas faire cela de façon simple, et le but de la table de hachage est quand même de gagner du temps, nous nous contenterons d'une discrétisation indépendante et identique selon chaque axe, mais dépendante des coordonnées : le pas selon l'axe i est $h(x_i)$. Une majoration grossière du nombre de cases intersectées par la zone compatible est en utilisant le pavé englobant

$$n_{gh}(x) \leq \prod_{i=1}^{n}\left(\left\lceil\frac{2\,\varepsilon_{gh}(x)}{h(x_i)}\right\rceil + 1\right)$$

L'étalement est donc majoré par

$$\mu_{gh}(x) \leq \frac{n_{gh}(x)\prod_{i=1}^{n} h(x_i)}{\mathcal{V}}$$

Afin de ne pas trop perdre en étalement, il serait intéressant que le plus grand côté d'une case centrée en x soit inférieur à $\varepsilon_{gh}(x)$, mais le moins possible pour ne pas perdre en place.

A norme constante, la plus grande coordonnée est obtenue si le point est sur un axe, par exemple $x = (z, 0, \ldots 0)$. On cherche donc $h(z)$ tel que :

$$h(z) \leq \varepsilon_{al}(z * [1, 0, \ldots 0]) = \varepsilon_p(2 + \mathcal{K}_A z)\|\bar{A}^{-1}\| \tag{4.5}$$

Examinons maintenant la manière de discrétiser un axe : soient $z_0 = 0, z_1, \ldots z_N$ les valeurs frontières positives des buckets selon l'axe des z. Le choix de $z_i = k\, i$ conduit à un pas de hachage $h(z) = k$, ce qui nous l'avons vu ne convient pas. Le choix $z_i = k\, i^2$ produit un pas moyen de $h_i = 2\, k\, i$, soit $h(z) \simeq 2\sqrt{k\, z}$. Il reste à régler le paramètre k : l'inéquation 4.5 se réécrit

$$h(z) = 2\sqrt{k\, z} \leq \varepsilon_p(2 + \mathcal{K}_A\, z)\|\bar{A}^{-1}\|$$

en notant $\alpha = \varepsilon_p^2\|\bar{A}^{-1}\|^2$ pour les calculs, on obtient

$$z^2\left(\frac{\mathcal{K}_A^2}{4}\right) + z\left(\mathcal{K}_A - \frac{k}{\alpha}\right) + 1 \geq 0$$

On est sûr que ceci est vérifié pour tout z si le déterminant est négatif ou nul, or

$$\Delta = \frac{k}{\alpha}\left(\frac{k}{\alpha} - 2\mathcal{K}_A\right)$$

Comme il vaut mieux choisir k maximum pour minimiser la taille de la table de hachage, il suffit de prendre $\Delta = 0$, soit :

$$k = 2\,\varepsilon_p^2\mathcal{K}_A\|\bar{A}^{-1}\|^2 \tag{4.6}$$

En ce qui concerne les transformations rigides, il faut encore rajouter les dimensions correspondant aux invariants supplémentaires traités par hachage simple. Afin de clarifier ceci, nous allons prendre un exemple pour les transformations rigides 2D.

4.3.3 Exemple : transformations rigides 2D

La zone compatible a un rayon de $\varepsilon_{gh} = 2\,\varepsilon_p\left(1 + \frac{\|x\|}{m}\right) + O(\varepsilon_p^2)$ en notant $m = \|m2 - m1\|$ la longueur de la base, dont l'erreur maximum est de $2\,\varepsilon_p$. On choisit tout d'abord un pas de discrétisation selon l'axe m de $h_m = \varepsilon_p$. Dans chaque tranche $m = C^{te}$ de la table, on doit ensuite discrétiser selon les deux axes des coordonnées des points. Pour cela, calculons k : on a $\mathcal{K}_A = 2/m$ et $\|\bar{A}^{-1}\| = 1$, ce qui donne

$$k = \frac{4\varepsilon_p^2}{m}$$

En ce qui concerne la valeur de m dans cette formule, on choisira la valeur supérieure dans la tranche, c'est à dire $\hat{m} = \left\lceil \frac{m}{h_m} \right\rceil h_m = \left\lceil \frac{m}{\varepsilon_p} \right\rceil \varepsilon_p$.

A titre d'indication, on a tracé dans la figure 4.4 un modèle d'ordinateur et la tranche de la table de hachage correspondant à la base indiquée en gras. L'erreur de mesure ε_p sur les points de l'image a été fixée à 5 pixels pour permettre la lisibilité de la figure.

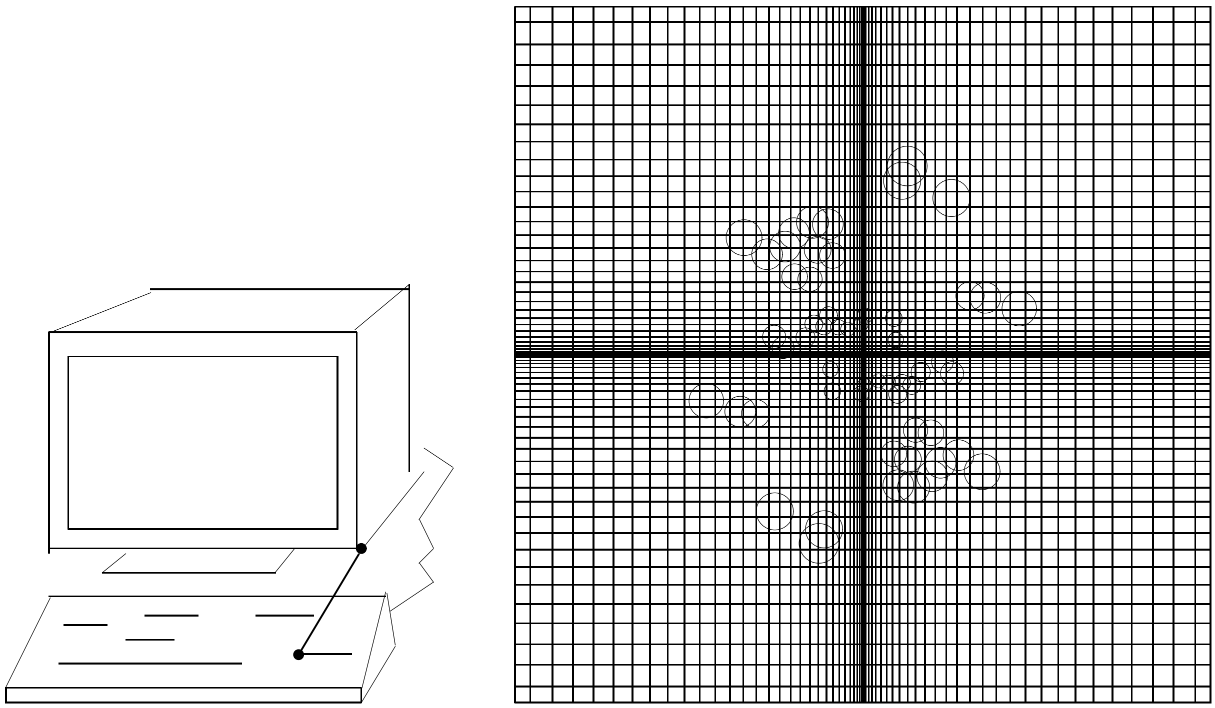


Figure 4.4 : A droite, une image 256*256. La base est indiquée en gras. A gauche, la tranche de la table de hachage correspondante ($\varepsilon_p = 5$, $m = 127$, $k = 0.77$). L'unité de mesure est le pixel.

4.4 Résumé

Nous avons tout d'abord remarqué que le hachage géométrique nécessitait la définition de bases modèle et image, alors qu'on pouvait pour l'alignement déterminer directement la transformation. On a ensuite montré que si l'on note ε_{al} le rayon de la zone de recherche pour l'alignement par la méthode de définition des bases, on pouvait définir la zone compatible dans l'espace des invariants[1] par son rayon

$$\varepsilon_{gh} = \|\bar{A}^{-1}\| \, \varepsilon_{al}$$

où $\bar{A}$ est la matrice de passage de la base image choisie à la base image canonique.

Il existe alors deux manières de traiter l'erreur : lors de l'apprentissage, en indexant la table de hachage avec ces zones, ou à la reconnaissance, en votant avec ces zones. Les transformations affines procurent un exemple où il n'est pas possible de préparer la table avec l'erreur, à cause du facteur $\|\bar{A}^{-1}\|$ qui dépend de la base image choisie, sauf si comme nous le proposons on rajoute une dimension supplémentaire à la table qui indexe justement ce

[1] On rappelle qu'il est constitué des coordonnées des points exprimées dans la base définie et des invariants relatifs à cette base.

facteur. Les transformations rigides 3D ont quant à elles posé le problème de la définition de la base. En effet, nous ne connaissons pas de méthode pour définir les bases images et modèle qui corresponde à la transformation aux moindres carrés. Il a donc fallu établir la définition d'une base, et la méthode s'est avérée être très légèrement meilleure que les moindres carrés au niveau de la borne sur l'erreur. Il faut cependant noter que cela n'implique en aucun cas la supériorité en ce qui concerne l'erreur moyenne. Nous avons alors obtenu la borne

$$\varepsilon_{gh}(x) \simeq 2\,\varepsilon_p \left(1 + \|x\| \sqrt{\frac{\left(4\ \sum_{i<j} \|x_i - x_j\|\right)^2}{9\,\|x_{1\wedge}x_2\|^2} + \frac{1}{\max_{i,j} \|x_i - x_j\|^2}} \right) + O\left({\varepsilon_p}^2\right)$$

Nous avons ensuite adressé le problème de la discrétisation de l'espace des invariants dans le cas du hachage simple, où la borne constante sur l'erreur est parfaitement en accord avec un pas de discrétisation constant, puis pour le hachage géométrique où la borne dépendant de la position x nous a amené à développer une discrétisation de pas non constant par une méthode relativement simple et entièrement automatisable. L'exemple présenté concernant les transformations rigides 2D montre d'ailleurs une adéquation très correcte des zones d'erreur avec la taille des cases de la table.

Encore une fois, le cadre théorique développé est tout à fait applicable et fournit par ailleurs des réponses intéressantes et inédites à des problèmes comme la discrétisation de la table de hachage qui sont somme toute assez peu étudiés.

Chapitre 5

Probabilité de fausse alarme et complexité moyenne

Nous avons vu jusqu'à présent le calcul des zones d'erreur (ou de compatibilité) des primitives géométriques et les modifications que cela apporte aux algorithmes d'alignement et de hachage. La contrepartie de la prise en compte de l'erreur est la possibilité de 'conspirations', c'est à dire de reconnaissances fantômes. Celles-ci ont différentes causes :

- On considère une borne sur chacun des points qui prend en compte l'erreur systématique sur la transformation, mais de façon indépendante. Or cette erreur est bien évidemment corrélée sur tous les points d'un modèle : ils seront tous décalés 'dans le même sens'. Comme cette corrélation n'est pas vérifiée lors de la reconnaissance, un certain nombre de points dans l'image peuvent tomber par hasard dans les zones compatibles relatives à une base et ainsi donner l'illusion que le modèle est présent. Une étape de vérification après la reconnaissance s'impose donc, non seulement pour affiner la transformation, mais aussi pour vérifier qu'avec cette transformation censée être précise, l'erreur sur les points reconnus ne dépasse pas l'erreur de mesure possible. Les faux positifs proviennent ici du fait que les algorithmes n'exploitent pas, et ne peuvent pas exploiter pour des raisons de complexité, les corrélations entre les points de l'image.

- On travaille souvent avec des points parce que c'est plus facile que de travailler directement sur les primitives réelles. Par exemple, si l'on s'occupe d'objets polygonaux, on utilisera les sommets pour la reconnaissance, et on vérifiera ensuite les arêtes. Il est ainsi possible de reconnaître un hexagone là où il y a une étoile régulière à 6 branches. Seule l'étape de vérification fera la différence. Les faux positifs sont dus dans ce cas à la sous-exploitation des données lors de la reconnaissance.

Le but de ce chapitre est d'étudier d'un point de vue probabiliste la fréquence de ce phénomène. Nous verrons ainsi que si dans certains cas les faux positifs sont extrêmement rares, et l'étape de vérification presque inutile (algorithme idéal), le plus souvent l'alignement ou le hachage géométrique effectue seulement un filtrage de l'arbre d'interprétation. Le nombre d'hypothèses d'appariement à vérifier en sortie est alors crucial pour déterminer la complexité moyenne. Enfin, il se peut que ces algorithmes soient tout simplement inefficaces, et l'on a intérêt dans ce cas soit à considérer des modèles probabilistes ([15, 16, 17]), soit à utiliser d'autres algorithmes plus coûteux au niveau complexité, mais peut être plus précis.

5.1 Etalement statistique moyen

Le principe de l'analyse est le suivant : nous allons tout d'abord calculer la probabilité qu'un point au hasard tombe dans une zone d'erreur relatif à une base, puis la probabilité qu'un certain nombre de points tombent dans des zones toujours relatives à une seule base. Selon le critère d'acceptation choisi (fraction des votes ou nombre de votes supérieurs à un seuil) nous pourrons alors déterminer la probabilité d'acceptation d'une association relativement à une base. Ceci permettra de dériver le nombre moyen d'associations acceptées, et donc le nombre moyen d'hypothèses à vérifier.

Le premier problème qui se pose est **"qu'est-ce qu'un point pris au hasard ?"**

Une hypothèse raisonnable est de considérer une image possédant n points aléatoirement répartis de façon uniforme. Cela convient tout à fait pour l'alignement puisqu'on travaille dans l'espace image. Par contre, pour la hachage il faudrait calculer la loi de probabilité induite dans l'espace des invariants. Nous aurions par exemple besoin de savoir pour les transformations rigides 2D quelle est la loi de la 'longueur d'une base', puis, ayant choisi 2 points (aléatoires suivant la loi uniforme dans l'image) pour former une base, nous devons calculer la loi de probabilité d'un point dans cette base.

Supposons pour l'instant connue la densité de probabilité ρ de cette loi dans l'espace image pour l'alignement (uniforme) ou dans l'espace des invariants pour le hachage simple et le hachage géométrique. Le second problème qui se pose est relatif à la taille des zones de compatibilité. Nous avons vu deux types de formulation pour le rayon de la zone d'erreur :

- Une borne constante du type $\varepsilon_l = 2\,\varepsilon_p$ pour les invariants utilisés en hachage simple.
- Une borne du type $\varepsilon_{al} = \varepsilon_p\ (2 + \mathcal{K}_A\|x\|)$ ou $\varepsilon_{gh} = \max\|\delta x\| \leq \|\bar{A}^{-1}\|\ \|\delta y - \delta A.x\| = \|\bar{A}^{-1}\|$ pour l'alignement et le hachage géométrique.

La probabilité qu'un point aléatoire ayant une densité de probabilité ρ tombe dans la zone Z_x compatible avec le point x est donc définie en notant $\mathbf{I}_{(Z_x)}$ l'indicatrice de cette zone par

$$p(x) = \int \rho(y)\mathbf{I}_{(Z_x)}(y)dy$$

Dans tous les cas, la probabilité p est dépendante de x, c'est à dire que la probabilité qu'un point pris au hasard tombe dans *une* zone relative à *une* base dépend de la position des points du modèle. Afin d'intégrer ceci, il nous faudrait la distribution des points du modèle: **"quelles sont les hypothèses à faire sur les modèles ?"**

La réponse est difficile à donner tant les modèles peuvent varier du tout au tout en fonction des applications. Il serait en fait audacieux de faire des hypothèses génériques sur les modèles. Regardons par exemple en imagerie médicale les modèles de crânes ou autres organes obtenus par extraction des points extrémaux sur les lignes de crête [18] : les points sont à peu près répartis dans un volume qui dépend bien entendu des organes. Si par contre nous faisons de la reconnaissance 3D après reconstruction de la scène à partir d'images 2D, les points modèles seront forcement répartis sur la surface. Il faudrait donc une étude précise de l'application pour pouvoir émettre des hypothèses valables sur les modèles. De plus, nous avons jusqu'à présent éludé le problème du hachage, mais dans ce cas **"les zones pour le hachage dépendent de la discrétisation de la table et le volume de la zone résultante n'est pas connu de façon théorique."**

En résumé, nous pouvons dire que s'il est sans doute possible d'obtenir la distribution des points dans l'espace des invariants induite par une distribution uniforme dans l'espace image, l'étape qui met en jeu les zones d'erreur est trop liée aux modèles pour pouvoir faire les calculs de probabilité de façon théorique. En fait, il est beaucoup plus simple et plus intéressant d'obtenir la probabilité d'un vote dans une zone de manière statistique à partir de la bibliothèque de modèles, ou directement de la table de hachage, et ceci pour une application donnée. On obtient ainsi un paramètre $\bar{\mu}$ que l'on appellera étalement moyen. Il faut bien faire attention que $\bar{\mu}$ ne correspond pas forcement à la moyenne des étalements des modèles : il faut dans le cas du hachage prendre en compte la probabilité ρ dans l'espace des invariants et l'étalement réel après discrétisation. Une étude de ce problème dans le cas des transformations affines 2D est présentée dans [5].

Nous considérerons dans la suite que ce paramètre a été déterminé pour l'application que l'on voudra.

5.2 Probabilité d'acceptation d'une association

5.2.1 Notations

Soit k le nombre de points nécessaires à la définition d'une base. On considère une bibliothèque constituée de M modèles de m points chacun, et on cherche à reconnaître ces objets dans une scène constituée de n points aléatoires (loi uniforme).
On note $\bar{\mu}$ l'étalement moyen obtenu par les statistiques sur la bibliothèque, aussi bien pour l'alignement que pour le hachage. On rappelle que $\bar{\mu}$ est la probabilité qu'un point de la scène tombe dans une zone de compatibilité indéterminée mais unique.

5.2.2 Traitement de l'erreur à l'apprentissage et alignement

Un modèle ayant m points dont k sont utilisés pour former la base, il reste $m-k$ points à mettre en correspondance. Dans le cas de l'alignement il y aura donc $m-k$ zones de recherche dans la scène et pour le hachage $m-k$ zones de compatibilité indexant cette base dans la table de hachage. La probabilité qu'un point de la scène ne tombe dans aucune de ces zones est $(1-\bar{\mu})^{m-k}$, donc la probabilité qu'il tombe dans l'une au moins de ces zones et qu'il vote pour cette base est

$$p = 1 - (1 - \bar{\mu})^{m-k} \tag{5.1}$$

La probabilité calculée ci-dessus est celle qu'il y ait au moins un vote pour une paire dû à un point (par rapport à une base) image. On s'assurera par des méthodes appropriées (compteur dans la paire ou séparation des aires se chevauchant dans la table de hachage) qu'il n'y ait qu'un seul vote effectif par paire, ceci pour préserver la justesse de l'algorithme.

5.2.3 Traitement de l'erreur lors de la reconnaissance

Dans le cas d'une discrétisation de pas constant pour la table de hachage, Lamdan et Wolfson [2] proposent l'analyse suivante :

- Chaque base a $m-k$ entrées dans la table. La probabilité de voter pour une entrée est $\nu = \frac{m-k}{N}$ où N est le nombre de cases de la table.
- La zone avec laquelle on vote a un étalement moyen de $\bar{\mu}$, et recouvre en moyenne $b = \bar{\mu}N$ cases de la table. On vote donc en moyenne b fois.
- La probabilité de ne voter aucune fois sur b pour une case indexant cette paire est $(1-\nu)^b$, et donc celle de voter au moins une fois pour la paire considérée est :

$$p_W = 1 - (1 - \frac{m-k}{N})^b \tag{5.2}$$

5.2.4 Equivalence des méthodes

Toujours dans le cas d'un pas de discrétisation constant, on peut réécrire les probabilités :

$$\begin{cases} p_W & = 1 - (1-\nu)^{N\bar{\mu}} \\ p & = 1 - (1-\bar{\mu})^{N\nu} \end{cases} \tag{5.3}$$

Dans le cadre d'une utilisation normale de l'algorithme, le nombre de cases indexant une base dans la table doit être inférieur au nombre de cases de la table $m - k \ll N$; de même, le volume moyen d'une zone d'erreur doit être inférieur au volume de la table $\bar{\mu} \ll 1$. Dans le cas contraire, il parait difficile de pouvoir reconnaître quoi que ce soit. Ces précisions autorisent les développements limités en ν et $\bar{\mu}$ des formules 5.3 :

$$\begin{cases} p_W & = N\bar{\mu}\nu - (\frac{(N\bar{\mu}\nu)^2}{2} - \frac{N\bar{\mu}\nu}{2}\nu) + O(\nu^3) \\ p & = N\bar{\mu}\nu - (\frac{(N\bar{\mu}\nu)^2}{2} - \frac{N\bar{\mu}\nu}{2}\bar{\mu}) + O(\bar{\mu}^3) \end{cases}$$

La différence relative des probabilités est alors :

$$\frac{p_W - p}{p_W} = \frac{\nu - \bar{\mu}}{2} + O(\nu^2 + \bar{\mu}^2) \tag{5.4}$$

ce qui montre que ce sont quasiment les mêmes probabilités. Dans le cas où $N\nu = m-k \gg 1$ et $N\bar{\mu} = b \gg 1$, on a même identité des probabilités à l'ordre 2

$$p_W = p = (m-k)\bar{\mu}(1 - \frac{(m-k)\bar{\mu}}{2}) + O(\bar{\mu}^2) \tag{5.5}$$

Dans le cas où la table de hachage a un pas de discrétisation non constant, il suffit de remarquer qu'on peut toujours rediscrétiser la table de hachage avec un pas constant bien choisi pour faire notre analyse. On peut donc conclure que
Sous des hypothèses très peu restrictives, la probabilité qu'un point pris au hasard dans la scène vote pour une base est

$$p = 1 - (1-\bar{\mu})^{m-k} = \bar{\mu}(m-k)(1 - \frac{(m-k)\bar{\mu}}{2}) + O(\bar{\mu}^2)$$

5.2.5 Probabilité d'acceptation d'une association

Critère d'acceptation

On accepte l'association d'une base modèle et d'une base image si elle met plus de τ points en correspondance. On peut également écrire ça comme la fraction des votes sur le nombre maximum de votes supérieure à un seuil α fixé ($\alpha = \frac{\tau}{m-k}$), ce qui est plus raisonnable si le nombre de points par modèle n'est pas constant.

Probabilité théorique

Dès qu'on a choisi une base image, il reste $n-k$ points images qui vont voter soit spécifiquement pour l'association avec la base modèle que l'on est en train de tester s'il s'agit d'alignement, soit au travers de la table de hachage. Comme la probabilité qu'un point vote pour une base donnée est p, la probabilité que j points sur $n-k$ votent pour la même base suit la loi binomiale :

$$B_{(n-k,p)}(j) = \begin{pmatrix} j \\ n-k \end{pmatrix} p^j (1-p)^{n-k-j}$$

La probabilité d'acceptation d'une association de bases est donc

$$q = \sum_{j \geq \tau} B_{(n-k,p)}(j) = 1 - \sum_{j=0}^{\tau} \begin{pmatrix} j \\ n-k \end{pmatrix} p^j (1-p)^{n-k-j} = I_p(\tau+1, n-\tau) \tag{5.6}$$

où $I_p(a,b)$ est la fonction Bêta incomplète normalisée.

Approximation

Dans l'hypothèse $p \ll 1$, c'est à dire $\bar{\mu}(m-k) \ll 1$, et $n-k \gg 1$, on peut approcher la binomiale $B_{(n-k,p)}$ par la loi de Poisson de paramètre $\lambda = (n-k)p$. Dans la pratique, les conditions d'application sont toujours vérifiées et on obtient :

$$q \simeq 1 - e^{-\lambda} \sum_{j=0}^{\tau} \frac{\lambda^j}{j!} \tag{5.7}$$

5.3 Complexité moyenne

5.3.1 Nombre moyen de fausses associations par base image

Hachage géométrique

Pour former une base modèle, on choisit k points parmi m. Pour chacun des M modèles de la bibliothèque, il y a donc $\binom{k}{m}$ possibilités, ce qui fait un total de $M\binom{k}{m}$ bases indexées dans la table de hachage. Le nombre de fausses associations suit ainsi la loi binomiale $B_{(M\binom{k}{m},q)}$ et le nombre moyen de fausses associations est

$$\bar{f}_{gh} = M\binom{k}{m}q$$

Hachage simple

Il s'agit de trouver le nombre moyen d'objets qu'un invariant aléatoire indexe. Soit I le nombre d'objets indexés par la table, et $\bar{\mu}_i$ l'étalement statistique moyen de celle-ci. Par définition,la probabilité qu'un point indexe un objet est $\bar{\mu}_i$, et donc la probabilité que ce point indexe j objets est donné par la binomiale $B_{(I,\bar{\mu}_i)}$. Le nombre moyen d'objets indexés est alors :

$$\bar{f}_i = \bar{\mu}_i I$$

Alignement

Avec les modifications que nous lui avons apportés au chapitre 3, l'algorithme d'alignement s'énonce :

Pour chaque base image, déterminer à partir des invariants l'ensemble des bases modèles compatibles. Pour chacune de celles-ci, calculer la transformation du modèle dans l'image et rechercher des correspondants aux $m-k$ points restants du modèle.

Observons l'étape de recherche des bases modèles compatibles : celle-ci s'effectue par un hachage simple, et dans ce cas les objets indexés dans la table sont les bases modèles $I = M\binom{k}{m}$. L'étalement moyen $\bar{\mu}_i$ exprime directement la proportion moyenne d'objets indexés. Par exemple, pour les transformations affines, il n'y a pas d'invariant (donc en réalité pas de hachage) d'où $\bar{\mu}_i = 1$. Pour les transformations rigides 2D ou 3D il y a par contre

respectivement 1 et 3 invariants, et $\bar{\mu}_i$ est calculé selon le principe de la section 5.1.
Pour chaque base image, on a donc $\bar{f}_i = \bar{\mu}_i M\binom{k}{m}$ bases modèles possibles. Pour chacune de celles-ci, on a vu que la probabilité d'acceptation est q_{al}. La probabilité que j bases soit acceptées suit la binomiale $B_{(\bar{f}_i,q_{al})}$ et le nombre moyen de fausses associations est donc

$$\bar{f}_{al} = M\binom{k}{m}\bar{\mu}_i q_{al}$$

5.3.2 Nombre d'hypothèses à vérifier

Dans l'algorithme initial, on effectue la recherche des associations pour chacune des $\binom{k}{n}$ bases images possibles. Le nombre moyen de vérifications s'écrit alors

$$\begin{aligned} H_{gh} &= M\tbinom{k}{m}\tbinom{k}{n}q_{gh} \\ H_{al} &= M\tbinom{k}{m}\tbinom{k}{n}\bar{\mu}_i q_{al} \end{aligned}$$

En étant plus réaliste, on peut se permettre d'arrêter la recherche dès la première bonne solution. On fait bien entendu la vérification des hypothèses retenues par le hachage ou l'alignement dans la foulée.

Nombre moyen de bases image à envisager

Supposons que l'image contienne un objet avec un taux d'occultation c, c'est à dire que l'on ne voit que $(1-c)m$ points de cet objet (sur m définissant le modèle). $\binom{k}{(1-c)m}$ bases sur les $\binom{k}{n}$ possibles dans la scène conduisent donc à une reconnaissance correcte. La probabilité de choisir l'une de ces bases est

$$r = \frac{\binom{k}{(1-c)m}}{\binom{k}{n}}$$

et celle de trouver une bonne base à l'essai t est $r(1-r)^{t-1}$.
Le nombre moyen de bases à consulter est donc :

$$B_{moy} = \frac{1}{r} = \frac{\binom{k}{n}}{\binom{k}{(1-c)m}}$$

La fraction des votes $\alpha = \frac{\tau}{m-k}$ servant de seuil à l'acceptation d'un association doit être inférieure à la fraction visible $(1-c)$ de l'objet à reconnaître. En effet, si $(1-c)m$ points de l'objet sont visibles, on ne peut obtenir que $(1-c)m-k$ votes. Le cas le pire est obtenu pour l'occultation maximum : $(1-c)m = \tau + k$, soit

$$r = \frac{\binom{k}{\tau+k}}{\binom{k}{n}}$$

Nombre de vérifications

Avec l'arrêt de l'algorithme dès la première bonne solution, ce nombre devient

$$\begin{aligned} H_{gh} &= M\binom{k}{m}\binom{k}{n}\frac{q_{gh}}{\binom{k}{\tau+k}} \\ H_{al} &= M\binom{k}{m}\binom{k}{n}\frac{\bar{\mu}_i q_{al}}{\binom{k}{\tau+k}} \end{aligned} \tag{5.8}$$

On remarquera que $\frac{q_{gh}}{\binom{k}{\tau+k}}$ et $\frac{\bar{\mu}_i q_{al}}{\binom{k}{\tau+k}}$ constituent les taux moyen d'élagage de l'arbre de recherche des associations pour le hachage géométrique et l'alignement.

Approximations

k restant faible, on peut approximer $\binom{k}{m}$ par $\frac{m^k}{k!}$ d'où en notant q_{gh} ou $\bar{\mu}_i q_{al}$ par q :

$$H' \simeq Mm^k n^k \frac{q}{(k!)\tau^k} = Mn^k \frac{q}{(k!)\alpha^k} \tag{5.9}$$

5.3.3 Complexité moyenne

Etape de filtrage

Si l'on utilise le hachage géométrique, on doit consulter $B_{moy} \simeq \frac{n^k}{\alpha^k m^k}$ bases images, chacune d'entre elle nécessitant le vote (en temps constant) de $O(n)$ points. L'étape de hachage pur a donc avec ou sans traitement de l'erreur une complexité

$$C^1_{gh} = O\left(\frac{n^{k+1}}{\alpha^k m^k}\right)$$

Avec l'alignement, on doit tester l'appariement de B_{moy} bases images avec en moyenne $\bar{\mu}_i M\binom{k}{m} \simeq \bar{\mu}_i M\frac{m^k}{(k!)}$ bases modèles, chacun de ces test nécessitant la recherche de correspondants pour les $O(m)$ points du modèle. La complexité de cette étape est donc :

$$C^1_{al} = O\left(Mmn^k \frac{\bar{\mu}_i}{(k!)\alpha^k}\right)$$

Etape de vérification

En général, une vérification demande un temps $O(m)$. En effet, s'il n'y a qu'un seul objet à reconnaître , au plus m points sont appariés. Comme la vérification consiste dans la plupart des cas à calculer la transformation aux moindres carrés entre tous les points appariés et vérifier ensuite que l'erreur finale entre points appariés ne dépasse pas l'erreur de mesure, la complexité reste en $O(m)$. La complexité de l'étape pour les deux algorithmes est donc

$$C^2 = O\left(Mmn^k \frac{q}{(k!)\alpha^k}\right)$$

5.4 Utilisation Pratique des formules

Etant donné une application, on calcule statistiquement le paramètre $\bar{\mu}_{gh}$ ou $\bar{\mu}_{al}$. Ceci permet de déterminer la probabilité p qu'un point au hasard vote pour une base. On peut alors tracer la surface définie par le nombre moyen de vérifications en fonction du seuil d'acceptation α et du nombre de points de la scène $H_{gh}(\alpha, n)$ ou $H_{al}(\alpha, n)$, et déterminer graphiquement la zone du plan (α, n) où ce nombre est très inférieur à 1. On peut d'ores et déjà conclure que si l'on travaille dans cette zone, il n'y aura (quasiment) aucun faux positif.
Si cela ne suffit pas, il vaut mieux tracer ensuite le taux moyen d'élagage de l'arbre d'interprétation $\frac{q(\alpha,n)}{(k!)\alpha^k}$, ce qui donne une bonne indication de la qualité du filtrage réalisé. On peut ainsi déterminer le seuil $\alpha(n)$ d'acceptation des associations qui permet d'obtenir un filtrage d'une valeur donnée.
Si toutefois cela ne convient pas, et qu'on ne puisse obtenir qu'un filtrage très médiocre (proche de 1), c'est que l'on a atteint les limites de notre modélisation de l'erreur, et il faut alors soit utiliser des modèles probabilistes ([15, 16, 17]), soit changer d'algorithme.

5.4.1 Exemple d'application

Nous allons nous focaliser ici sur l'algorithme d'alignement proposé par J. P. Thirion dans [18] : il s'agit de recalage rigide 3D entre des images scanner du même patient. Dans un souci de simplicité, nous ne considérerons dans notre analyse aucune restriction sur l'appariement des bases $\bar{\mu}_i = 1$. Le calcul de $\bar{\mu}_{al}$ est lui aussi très simplifié puisque l'auteur considère des zones de recherche d'un rayon constant égal à $\varepsilon = 0.6$ en voxel[1]. Etant donné que l'image a un volume de $\mathcal{V} = 5.10^6$ voxels, on obtient facilement

$$\bar{\mu}_{al} = \frac{4\pi\varepsilon^3}{3\mathcal{V}} = 1.8\ 10^{-7}$$

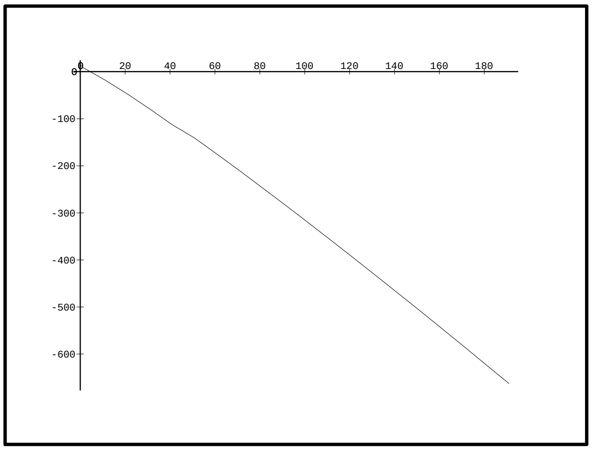
Comme il s'agit de recalage entre deux images, il n'y a qu'un seul modèle ($M = 1$), défini par $m = 350$ points, et on a le même nombre $n = 350$ de points dans la scène. Il faut bien entendu $k = 3$ points pour définir une base.
Le calcul de p ne pose aucun problème : $p = 6.3 10^{-5}$. On trace ensuite directement le nombre de faux positifs en fonction de τ, seuil sur le nombre de points appariés (figure 5.1).

On se rend compte qu'à partir de 7 points appariés en plus de la base, on une chance sur 10^7 environ que ce soit une conspiration ! On voit donc que l'alignement peut être une méthode très efficace.

5.5 Résumé

Dans le processus de calcul du nombre de faux positifs, nous avons tout d'abord vu qu'il était nécessaire de mener une étude statistique particulière à chaque application afin de déterminer

[1] Le voxel est la généralisation à 3 dimensions de la notion de pixel. Il est parfois considéré comme une longueur, c'est alors son côté qu'il faut prendre en compte.



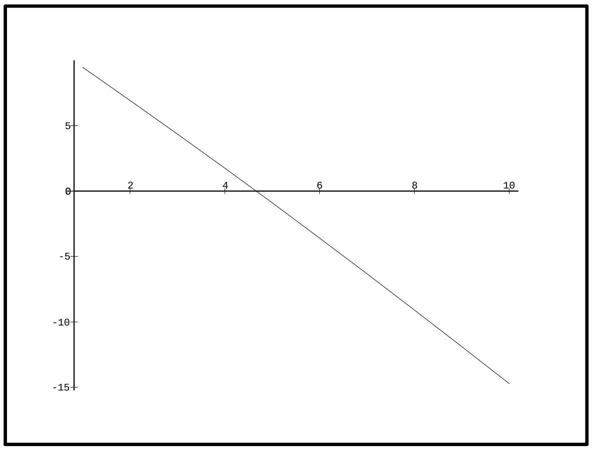


Figure 5.1 : Nombre prévu de faux positifs ($Log_{10}(H_{al})$ en fonction du seuil τ sur le nombre de points appariés)

l'étalement moyen $\bar{\mu}$. A partir de ce paramètre, nous pouvons déterminer la probabilité de vote pour une base en considérant le traitement de l'erreur à l'apprentissage, dont la formalisation est identique à celle de l'alignement, ou lors de la reconnaissance. En fait, nous avons montré que ces deux analyses amènent intrinsèquement au même résultat dans le domaine de validité des algorithmes. De là découle très simplement la probabilité d'acceptation d'une association puis le nombre moyen de faux positifs et la complexité moyenne. On remarquera qu'on calcule au passage le nombre moyen de bases images à consulter avant de trouver l'une des bonnes.

Le nombre de faux positifs peut alors servir de critère pour l'établissement d'une étape de vérification, voire pour l'acceptation ou le refus de l'algorithme. On peut également déterminer pour une application donnée le seuil sur le nombre de votes à partir duquel on a une certaine qualité de filtrage de l'arbre d'interprétation ou nombre maximum de faux positifs. Un exemple développé sur une application en imagerie médicale montre que l'alignement est dans ce cas tout à fait précis.

Si cette fois-ci l'analyse théorique ne mène à aucun algorithme particulier, elle permet toutefois de déterminer de façon automatique certains paramètres comme le seuil d'acceptation sur le nombre de votes. Nous avons donc fait un pas de plus vers des méthodes entièrement automatisées.

Chapitre 6

Conclusion et perspectives

Nous avons obtenu en fin de compte des algorithmes tout à fait utilisables et traitant l'erreur de façon à produire un résultat exact dans le cas des erreurs bornées. Nous avons de plus automatisé le choix d'un certain nombre de paramètres dans ces méthodes.

Nous nous sommes particulièrement attaché aux algorithmes d'alignement et de hachage géométrique pour les transformations rigides en trois dimensions en raison de leur utilité pour le recalage d'images médicales. Ils semblent en effet particulièrement adaptés tant pour le suivi temporel d'un patient que pour le recalage inter-patient ou patient-atlas anatomique. Pour cette dernière application, il reste cependant une étude à mener avant de pouvoir tenter d'appliquer nos algorithmes : il faut déterminer l'erreur initiale que l'on doit prendre en compte. En effet, si l'erreur uniforme et bornée convient bien pour modéliser les erreurs de discrétisation dues à l'image, il est beaucoup moins sûr que cette modélisation convienne pour simuler des transformations presque rigides.

Une autre voie permettant d'aborder le problème est l'utilisation de modèles probabilistes : la même étude serait en quelque sorte à refaire, mais avec une modélisation gaussienne de l'erreur.

Loin de clore le problème, notre étude ouvre donc au contraire de nouvelles voies de recherche et de nouveaux espoirs dans le cadre de l'imagerie médicale.

Annexe A

Quaternions et rotation au moindres carrés

A.1 Les quaternions

A.1.1 Définition

Les quaternions sont les éléments d'une algèbre de dimension 4 sur $\mathbb{R}$ que l'on notera $\mathcal{Q}$. C'est aussi le premier corps non commutatif qui fut trouvé (par Hamilton en 1843). On peut construire cette algèbre de différentes façons, mais celle qui nous intéressera ici considère un quaternion $q \in \mathcal{Q}$ comme un couple $q = (a, v)$, où $a \in \mathbb{R}$ est la partie réelle et $v \in \mathbb{R}^3$ la partie dite pure. Les opérations définies pour former l'algèbre sont :

- L'addition : $(a_1, v_1) + (a_2, v_2) = (a_1 + a_2, v_1 + v_2)$
- La multiplication interne : $(a_1, v_1) * (a_2, v_2) = (a_1 a_2 - < v_1 | v_2 >, v_1 {}_\wedge v_2 + a_1 v_2 + a_2 v_1)$ où '$_\wedge$' et '$< .|. >$' désignent les produits vectoriel et scalaire usuels sur $\mathbb{R}^3$

On définit en plus le conjugué et la norme

- $\overline{(a, v)} = (-a, v)$
- $|q|^2 = \|q\|^2_{\mathcal{Q}} = \bar{q} * q = a^2 + \|v\|^2_{\mathbb{R}^3} = \|q\|^2_{\mathbb{R}^4}$

ce qui permet d'écrire l'inverse très simplement

$$q^{-1} = \frac{\bar{q}}{|q|^2}$$

pour $q \neq 0$. On remarquera que la norme est compatible avec le produit : $|q_1 q_2| = |q_1|.|q_2|$.

On pourra trouver une présentation beaucoup plus complète et exhaustive des quaternions et de leurs propriétés dans [10] et [11].

A.1.2 Rotations de $\mathbb{R}^3$

Parmi les applications linéaires de $\mathbb{R}^3$, les rotations vectorielles sont les isométries positives (elles conservent le produit scalaire et l'orientation). Muni de la loi de composition, cet ensemble forme un groupe non commutatif qui peut être représenté par l'ensemble des matrices 3×3 sur $\mathbb{R}$ vérifiant

$$R^t R = I \quad \text{et} \quad \det(R) = 1$$

Une seconde représentation, toujours non minimale, est fournie par les quaternions (voir paragraphe suivant). On peut encore utiliser les angles d'Euler ou le vecteur rotation [9]. De l'utilisation de ce dernier dérive la formule de Rodrigues : une rotation d'angle θ autour du vecteur n s'écrit

$$R_{(n,\theta)} = I + \sin\theta\tilde{n} + (1 - \cos\theta)\tilde{n}^2$$

où l'on note $\sim$ l'opération qui à un vecteur n associe la matrice (antisymétrique) $\tilde{n}$ telle que pour tout vecteur v on ait $\tilde{n}v = n_\wedge v$. Si le composantes de n sont (n_x, n_y, n_z), la matrice $\tilde{n}$ a la forme

$$\begin{pmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{pmatrix}$$

A.1.3 Quaternions et rotations

Soit q un quaternion unitaire. On vérifie aisément qu'il existe $\theta \in [0, \pi]$ et n vecteur unitaire de $\mathbb{R}$ tels que $q = \left(\cos\left(\frac{\theta}{2}\right), \sin\left(\frac{\theta}{2}\right).n\right)$. L'application

$$\begin{array}{rccc} R_q : & \mathcal{Q} & \longrightarrow & \mathcal{Q} \\ & p & \longmapsto & q * p * \bar{q} \end{array}$$

est un automorphisme intérieur de $\mathcal{Q}$ qui conserve les quaternions purs. Sa restriction à $\mathbb{R}^3$ est la rotation vectorielle de $\mathbb{R}^3$ d'angle θ autour du vecteur n. Inversement, on peut associer à toute rotation de $\mathbb{R}^3$ deux quaternions unitaires q et $-q$.

A.2 Minimisation aux moindres carrés

Possédant un ensemble de points x_i et leur image supposées y_i, on veut trouver la rotation R qui minimise l'erreur quadratique de superposition entre y_i et Rx_i. Le critère est donc

$$C = \sum_i \|y_i - Rx_i\|^2$$

En notant q l'un des deux quaternions unitaires associés à R et en identifiant les x_i et les y_i à des quaternions purs (partie réelle nulle), on obtient

$$\begin{aligned} C &= \sum_i |y_i - q * x_i * \bar{q}|^2 \\ &= \sum_i |y_i * q - q * x_i|^2 \\ &= \sum_i [A_i.q]^t [A_i.q] \\ &= q^t [\sum_i A_i^t A_i] q \\ &= q^t B q \end{aligned}$$

en notant A_i la matrice 4×4 vérifiant $A_i.q = y_i * q - q * x_i$ et $B = \sum_i A_i^t A_i$. L'application simple des lois sur les quaternions fournit en écrivant la matrice par blocs

$$A_i = \begin{pmatrix} 0 & (y_i - x_i)^t \\ & \\ (y_i - x_i) & (\tilde{y_i} + \tilde{x_i}) \end{pmatrix}$$

$$A_i^t A_i = \begin{pmatrix} \|y_i - x_i\|^2 & (y_i - x_i)^t (\tilde{y_i} + \tilde{x_i}) \\ & \\ (\tilde{y_i} + \tilde{x_i})(x_i - y_i) & (y_i - x_i)(y_i - x_i)^t - (\tilde{y_i} + \tilde{x_i})^2 \end{pmatrix}$$

En développant les termes et en simplifiant grâce à la formule de Gibbs[1] on obtient

$$A_i^t A_i = \begin{pmatrix} \|y_i - x_i\|^2 & 2(y_{i\wedge} x_i)^t \\ & \\ 2(y_{i\wedge} x_i) & \|y_i + x_i\|^2 I - 2(x_i y_i^t + y_i x_i^t) \end{pmatrix}$$

Ce qui permet d'écrire la matrice B avec les notations

$$\begin{aligned} \gamma &= \sum (\|x_i\|^2 + \|y_i\|^2) \\ \alpha &= \sum < x_i | y_i > \\ \beta &= \sum x_{i\wedge} y_i \\ \Gamma &= \sum (x_i y_i^t + y_i x_i^t) \end{aligned} \qquad B = \begin{pmatrix} \gamma - 2\alpha & -2\beta^t \\ & \\ -2\beta & (\gamma + 2\alpha) I - 2\Gamma \end{pmatrix} \tag{A.1}$$

Rappelons que le problème est de minimiser $C = q^t B q$ sous la contrainte $|q| = 1$. La solution est donnée par le vecteur propre unitaire associé à la plus petite valeur propre de B. Si cette résolution s'effectue fort bien numériquement, il n'y a pas actuellement de solution explicite de façon formelle.

A.3 Superposer deux triplets de points en repère - barycentrique

On considère trois points x_i en repère barycentrique, donc appartenant à un certain plan $\mathcal{M}$ défini par son vecteur normal $m = (x_{1\wedge} x_2)/\|x_{1\wedge} x_2\|$, et leurs correspondants y_i, également en repère barycentrique, appartenant au plan $\mathcal{S}$ défini par $s = (y_{1\wedge} y_2)/\|y_{1\wedge} y_2\|$.

[1] $\tilde{a}\tilde{b} = ba^t - < a|b > I$

Supposons que les plans $\mathcal{M}$ et $\mathcal{S}$ soient confondus (les x_i et les y_i sont coplanaires), avec une orientation identique : $m = s$. Nous allons trouver explicitement la rotation minimisant l'erreur de superposition. L'analogie avec le cas 2D permet de proposer deux quaternions rotation que l'on vérifiera être vecteurs propres de B. Des considérations de symétrie permettront ensuite de déterminer les autres valeurs propres de B qui s'avéreront être supérieures à la première. Ceci permettra de conclure notre démonstration.

A.3.1 $2 * 3$ points en deux dimensions

Le critère à minimiser pour la recherche de la rotation vectorielle est :

$$C = \sum_i \|y_i - Rx_i\|^2 \quad \text{avec} \quad R = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

Pour obtenir les angles optimaux, on dérive C par rapport à θ

$$\frac{\partial C}{\partial \theta} = 0 \quad \Leftrightarrow \quad \sin\theta \left\{ \sum_i < x_i | y_i > \right\} + \cos\theta \left\{ \sum_i |y_i, x_i| \right\} = 0$$

d'où les solutions

$$\begin{cases} \cos\theta\sqrt{(\sum < x_i|y_i >)^2 + (\sum |y_i, x_i|)^2} = \pm \sum < x_i|y_i > \\ \sin\theta\sqrt{(\sum < x_i|y_i >)^2 + (\sum |y_i, x_i|)^2} = \sum |y_i, x_i| \end{cases}$$

Supposons maintenant que les 6 points soient dans l'espace, mais coplanaires. La rotation proposée en 2D est de direction normale au plan. En appelant k le vecteur normal au plan dans le sens direct, on propose donc comme rotation (à la norme près) les quaternions

$$q_1 = \left| \begin{array}{l} 1 + \cos\theta \\ \sin\theta . k \end{array} \right. \quad \text{et} \quad q_2 = \left| \begin{array}{l} 1 - \cos\theta \\ \sin\theta . k \end{array} \right.$$

ce qui s'écrit avec nos notations

$$q_{1,2} = \left| \begin{array}{l} \sqrt{\alpha^2 + \|\beta\|^2} \pm \alpha \\ \beta \end{array} \right.$$

A.3.2 Deux vecteurs propres de B

Vérifions tout d'abord q_1 :

$$(B - \lambda I)q_1 = 0 \quad \Leftrightarrow \quad \begin{cases} (\gamma - 2\alpha - \lambda)(\sqrt{\alpha^2 + \|\beta\|^2} + \alpha) = 2\|\beta\|^2 \\ \beta(\gamma - 2\sqrt{\alpha^2 + \|\beta\|^2} - \lambda) = 2\Gamma\beta \end{cases}$$

Or $\beta = \sum x_{i\wedge} y_i$ est colinéaire à $m = s$ donc orthogonal aux x_i et aux y_i, ce qui montre que $\Gamma\beta = 0$. On obtient donc

$$(B - \lambda_1 I)q_1 = 0 \quad \Leftrightarrow \quad \lambda_1 = \gamma - 2\sqrt{\alpha^2 + \|\beta\|^2}$$

En ce qui concerne q_2, la démarche est identique et on a

$$(B - \lambda_2 I)q_2 = 0 \quad \Leftrightarrow \quad \lambda_2 = \gamma + 2\sqrt{\alpha^2 + \|\beta\|^2}$$

Donc q_1 et q_2 sont des vecteurs propres de B associés aux valeurs propres λ_1 et λ_2. Nous allons maintenant exprimer ces valeurs propres de façon différente.

$$\begin{aligned}
\alpha^2 + \|\beta\|^2 &= (\textstyle\sum_i < x_i|y_i >)^2 + < \sum_i x_{i\wedge}y_i| \sum_i x_{i\wedge}y_i > \\
&= \textstyle\sum_{i,j} < x_i|y_i >< x_j|y_j > + \sum_{i,j} < x_i|x_j >< y_i|y_j > - \sum_{i,j} < x_i|y_j >< x_j|y_i > \\
&= \textstyle\sum_{i,j} < x_{i\wedge}x_j|y_{i\wedge}y_j > + \sum_{i,j} < x_i|x_j >< y_i|y_j >
\end{aligned}$$

Notons r le second terme . On remarquera qu'il ne dépend que des positions relatives des x_i entre eux et des y_i entre eux. Il ne change donc pas lors d'une rotation : c'est une constante du problème, tout comme γ. Le premier terme se simplifie beaucoup en introduisant les valeurs de x_3 et y_3 en repère barycentrique : $x_3 = -x_1 - x_2$ et $y_3 = -y_1 - y_2$. On obtient

$$\begin{aligned}
\alpha^2 + \|\beta\|^2 &= r + 6 < x_{1\wedge}x_2|y_{1\wedge}y_2 > \\
&= r + 6\|x_{1\wedge}x_2\|.\|y_{1\wedge}y_2\| < m|s >
\end{aligned}$$

L'hypothèse $m = s$ n'étant pas intervenue dans la dérivation de cette expression, celle-ci est donc toujours valable.

$$\alpha^2 + \|\beta\|^2 = r + 6\|x_{1\wedge}x_2\|.\|y_{1\wedge}y_2\| < m|s > \quad \text{avec} \quad r = \sum_{i,j} < x_i|x_j >< y_i|y_j > \qquad \text{(A.2)}$$

Introduisant cette expression dans nos valeurs propres, on trouve une formule qui ne dépend plus de la position initiale des x_i par rapport aux y_i.

$$\lambda_{1,2} = \gamma \mp 2\sqrt{r + 6\|x_{1\wedge}x_2\|.\|y_{1\wedge}y_2\|}$$

A.3.3 Les deux autres valeurs propres

Notons maintenant μ_1 et μ_2 les deux valeurs propres non nulles de Γ en supposant qu'elles existent, et n_1, n_2 les vecteurs propres unitaires associés. Ces derniers sont dans $\mathcal{M} = \mathcal{S}$ d'après la formulation même de Γ. On vérifie aisément que $q_{3,4} = \left| \begin{array}{l} 0 \\ n_{3,4} \end{array} \right.$ est alors vecteur propre de B associé à la valeur propre $\lambda_{3,4} = \gamma + 2\alpha - \mu_{3,4}$.
Or q_3 et q_4 correspondent à des rotations de π autour d'un axe situé dans le plan $\mathcal{M} = \mathcal{S}$; ce sont donc des symétries par rapport à des droites pour ce plan. En particulier, elles amènent m sur $-s$. On peut donc retrouver les valeurs propres λ_3 et λ_4 en considérant les rotations d'axe m (ou s) dans le cas $m = -s$. Le raisonnement utilisé pour trouver λ_1 et λ_2 dans leur première forme ne tenant compte que de $\beta \,//\, m \,//\, s$, on trouve

$$\lambda_{3,4} = \gamma \pm 2\sqrt{\alpha^2 + \|\beta\|^2}$$

ce qui s'écrit en tenant maintenant compte de $< m|s >= -1$ dans l'expression A.2

$$\lambda_{3,4} = \gamma \mp 2\sqrt{r - 6\|x_{1\wedge}x_2\|.\|y_{1\wedge}y_2\|}$$

A.3.4 Conclusion

Quelque soit la position initiale des x_i par rapport aux y_i, on peut toujours décomposer la rotation recherchée A, minimisant l'erreur de superposition aux moindres carrés, en une rotation A_1 qui amène le vecteur m sur s, puis une rotation A_2 qui se situe dans le cadre de ce que nous venons de faire. Les valeurs propres étant écrites sous forme invariante par rotation des x_i, elles sont valables dans tous les cas.

Théoreme 1 *Les valeurs propres de la matrice B sont*

$$\lambda_1 = \gamma - 2\sqrt{r + 6\|x_{1\wedge}x_2\|.\|y_{1\wedge}y_2\|} \qquad \lambda_2 = \gamma + 2\sqrt{r + 6\|x_{1\wedge}x_2\|.\|y_{1\wedge}y_2\|}$$
$$\lambda_3 = \gamma - 2\sqrt{r - 6\|x_{1\wedge}x_2\|.\|y_{1\wedge}y_2\|} \qquad \lambda_4 = \gamma + 2\sqrt{r - 6\|x_{1\wedge}x_2\|.\|y_{1\wedge}y_2\|}$$

avec les notations

$$\gamma = \textstyle\sum_{i=1}^{3} (\|x_i\| + \|y_i\|) \qquad r = \textstyle\sum_{i=1,j=1}^{3,3} < x_i|x_j >< y_i|y_j >$$

L'ordre de ces racine est $\lambda_1 \leq \lambda_3 \leq \lambda_4 \leq \lambda_2$. La valeur du critère au minimum est donc λ_1, et le quaternion recherché est le vecteur propre unitaire associé à cette valeur propre.

Les rotations amenant m sur s s'écrivent comme combinaison linéaire (à la norme près) des deux rotations de base

$$p_1 = \left| \begin{array}{l} 1+ < m|s > \\ m_\wedge s \end{array} \right. \qquad p_2 = \left| \begin{array}{l} 0 \\ m + s \end{array} \right.$$

Il y a toutefois une singularité en $m = -s$, que nous avons exploitée précédemment, et pour laquelle ceci n'est pas valable. Il suffit dans ce cas d'une rotation de π par rapport à n'importe quel vecteur du plan.
En dehors de cette singularité, la rotation d'angle le plus faible est donnée par p_1. Notons R_1 la rotation équivalente, et $x'_i = R_1 x_i$. La rotation R_2 amenant au minimum est alors donnée par

$$q_1 = \left| \begin{array}{l} \sqrt{\alpha'^2 + \|\beta'\|^2} + \alpha' \\ \beta' \end{array} \right. \qquad \text{avec} \qquad \begin{array}{l} \alpha' = \sum_{i=1}^{3} < x'_i|y_i > \\ \beta' = \sum_{i=1}^{3} x'_{i\wedge} y_i \end{array}$$

Corollaire 1 *On peut toujours décomposer la rotation recherchée R en $R = R_2 R_1$ où R_1 est une rotation amenant m sur s, et R_2 est une rotation d'axe s.*

Annexe B

Exemples de recalage en imagerie médicale

A partir d'images scanner X, on extrait automatiquement les lignes de crête et les points extrémaux. L'exemple de deux images du crâne du même patient est présenté ci-dessous. Les modèles sont alors recalés avec la méthode d'alignement ; la superposition est montrée page suivante. Un second exemple d'images recalées montre les points extrémaux, la dernière image étant simplement un zoom. Les images sont grâcieusement prêtées par J.P. Thirion.

Références

[1] Y. LAMDAN, H.J. WOLFSON, 1988, *Geometric Hashing : A General and Efficient Model-Based Recognition Scheme*, Proc. Sec. Int. Conf. Comput. Vision, pp. 238-249.

[2] Y. LAMDAN, H.J. WOLFSON, 1991, *On the error analysis of Geometric Hashing*, IEEE Conf. Comput. Vis. and Patt. Recog. pp. 22-27.

[3] D.P. HUTTENLOCHER, S. ULLMAN, 1987, *Object Recognition using Alignement*, Proc. of ICCV, pp. 72-78.

[4] W.E.L. GRIMSON, D.P. HUTTENLOCHER, 1990, *On the sensitivity of Geometric Hashing*, Proc. Third Int. Conf. Comput. Vision, pp. 334-338.

[5] W.E.L GRIMSON, D.P HUTTENLOCHER, D.W. JACOBS, 1992, *A study of affine matching with bounded sensor error*, to appear.

[6] W.E.L GRIMSON, 1992, *Object Recognition By Computer - The role of Geometric Constraints*, MIT Press.

[7] A. GUEZIEC, N. AYACHE, 1992, *Smoothing and Matching of 3-D-Space Curves*, in IEEE Proc. of Comput. Vision and Pat. Recog., pp. 592-598.

[8] N. AYACHE, O. D. FAUGERAS, 1986, *HYPER: A New Approach for the Recognition and Positionning of Two-Dimensionnal Objects*, IEEE TPAMI, 8(1), pp. 44-54.

[9] N. AYACHE, 1989, *Vision Stéréoscopique et Perception Multisensorielle*, Inter-Editions.

[10] P. CASTELJAU, 1987, *Les quaternions*, Hermes.

[11] L. REYES-AVILA, Oct. 1990, *Les quaternions : une représentation paramétrique systématique des rotations finies*, Rapport de recherche 1303, INRIA.

[12] M. BERGER, 1992, *Géométrie, tome 1*, Nathan.

[13] W. M. WELLS, 1993, *Statistical Object Recognition*, PhD Thesis, MIT.

[14] F. C. TSAI, 1993, *A Probabilistic Approach to Geometric Hashing using Line Features*, Technical Report 640, Courant Institute of Mathematical Science, New York University.

[15] I. RIGOUSTOUS, R. HUMMEL, 1991, *Several Results on Affine Invariant Geometric Hashing*, Proceedings of the 8th Israeli Conference on Artificial Intelligence and Computer Vision.

[16] I. RIGOUSTOUS, R. HUMMEL, 1991, *Robust Similarity Invariant Matching in the Presence of Noise*, Proceedings of the 8th Israeli Conference on Artificial Intelligence and Computer Vision.

[17] I. RIGOUSTOUS, R. HUMMEL, 1992, *A Bayesian approach to model matching with geometric hashing*, Submitted to CVGIP-IU.

[18] J. P. THIRION, 1993, *New Invariant Feature Points for 3D Image Registration*, Rapport de Recherche 1901, INRIA.

[19] J. P. THIRION, A. GOURDON, O. MONGA, A. GUEZIEC, N. AYACHE, 1992, *Automatic Registration of 3D Images Using Surface Curvature*, in : 14th Ann. Int. Conf. IEEE Engineering in Medecine and Biology Society.

[20] J. P. THIRION, O. MONGA, S. BENAYOUN, A. GUEZIEC, N. AYACHE, 1992, *automatic Registration of 3D Images Using Surface Curvature*, In IEEE Int. Symp. on Optical Applied Science and Engineering, San-Diego.

[21] J. P. THIRION, A. GOURDON, 1993, *The 3D Marching Lines Algorithm : New Results and Proofs*, rapport de recherche INRIA 1881, (submitted to CVGIP).

[22] G. MALANDAIN, 1992, *Filtrage, Topologie et mise en correspondance d'images médicales multidimensionnelles*, Thèse de doctorat, Ecole Centrale de Paris.

[23] G. MALANDAIN, J.M. ROCCHISANI, 1992, *3D superposition of CT and SPECT images of the brain by a mechanical based method*, In Europ. Assoc. Nuclear Medecine Congress, Lisbon, Portugal, and In Europ. Journ. of Nuclear Medecine, vol. 19, n. 8, pp 624.