

THESE

présentée

DEVANT L'INSTITUT NATIONAL DES SCIENCES APPLIQUEES DE LYON

pour obtenir

LE GRADE DE DOCTEUR

FORMATION DOCTORALE : Automatique Industrielle

ECOLE DOCTORALE : Sciences pour l'ingénieur :
Electronique, Electrotechnique, Automatique

PAR

Véronique MORON

(Ingénieur INSA de LYON)

MISE EN CORRESPONDANCE DE DONNEES 3D AVEC UN MODELE CAO : APPLICATION A L'INSPECTION AUTOMATIQUE

Soutenue le 9 Décembre 1996 devant la commission d'examen

Jury :

MM. Ph. BOLON	Professeur à l'Université de Savoie / ESIA	Examineur
P. BOULANGER	Agent de Recherche du CNRC (Ottawa - Canada)	Co-directeur
M. DHOME	Directeur de Recherche CNRS au LASMEA de Clermont-Ferrand	Rapporteur
A. JUTARD	Professeur à l'INSA de Lyon	Président
D. PYZAK	Directeur-Marketing Matra Datavision	Examineur
H.T. REDARCE	Maître de conférences HdR à l'INSA de Lyon	Co-directeur
S. TICHKIEWITCH	Professeur à l'INP de Grenoble	Rapporteur

*Cette thèse a été préparée au Laboratoire d'Automatique Industrielle de l'INSA de LYON
et à l'Institut des Technologies de l'Information du Conseil National de Recherche du Canada*

**. Ecole Doctorale Mat riaux de Lyon =
INSA - ECL - UCBL - U. Chambéry - ENS**

Responsable : Pr. A. HOAREAU

Formations doctorales :

- Génie des matériaux (Pr. GUENIN)
 - Sciences des matériaux (Pr. BARRAT)
 - Matériaux macromoléculaires et composites (Pr. SAUTEREAU)
-

**.Ecole Doctorale des Sciences pour l'Ingénieur de Lyon :
Mécanique, Énergétique, Génie Civil, Acoustique (MEGA) =
ECL - INSA - UCBL**

Responsable : Pr. J. BATAILLE

Formations doctorales :

- Acoustique (Pr. GUYADER)
 - Génie civil : Sols, Matériaux, Structures
physique du bâtiment (Pr. LAREAL)
 - Mécanique (Pr. BATAILLE)
 - Thermique et Énergétique (Pr. LANCE)
-

**.Ecole Doctorale des Sciences pour l'Ingénieur de Lyon :
Électronique, Electrotechnique, Automatique (EEA) =
INSA - ECL - UCBL - U. Chambéry - U. St Etienne**

Responsable : Pr. G. GIMENEZ

Formations doctorales :

- Acoustique (Pr. GUYADER)
- Automatique industrielle (Pr. BOLON)
- Dispositifs de l'électronique intégrée (Pr. PINARD)
- Génie biologique et médical (Pr. COLLOMBEL)
- Génie Électrique (Pr. AURIOL)
- Signal, image, parole (Pr. LACOUME)

LISTE DES DEA ou FORMATIONS DOCTORALES

<i>FORMATIONS DOCTORALES</i>	<i>RESPONSABLES INSA</i>	<i>ADRESSES INSA</i>
Acoustique	<u>GUYADER Jean Louis</u>	Bât 303 Tél 80 80
Analyse et modélisation de systèmes biologiques	NARDON Paul	Bât 406 Tél 80 86
Automatique industrielle	SCAVARDA Serge	Bât 303 Tél 83 41
Biochimie	LAGARDE Michel	Bât 406 Tél 82 40
Chimie Inorganique	GONNARD Paul	Bât 504 Tél 81 58
Conception en bâtiment et techniques urbaines	<u>MIRAMOND Marcel</u>	Bât 304 Tél 82 09
DEA Informatique de Lyon	KOULOUMDJIAN Jacques	Bât 501 Tél 80 99
Dispositifs de l'électronique intégrée	<u>PINARD Pierre</u>	Bât 502 Tél 82 47
Génie biologique et médical	MAGNIN Isabelle	Bât 502 Tél 85 63
Génie civil : sols, matériaux, structures, physique du bâtiment	<u>LAREAL Pierre</u>	Bât 304 Tél 82 16
Génie Electrique	CHANTE Jean-Pierre	Bât 401 Tél 87 26
Matériaux polymères et ccomposites	<u>SAUTEREAU Henri</u>	Bât 403 Tél 81 78
Mécanique	DALMAZ Gérard	Bât 113 Tél 83 03
Microstructure et comportement mécanique et macroscopique des matériaux - génie des matériaux	<u>GUENIN Gérard</u>	Bât 502 Tél 82 45
Productique : organisation et conduite des systèmes de production	<u>FAVREL Joël</u>	Bât 502 Tél 83 63
Sciences des matériaux et des surfaces	LAUGIER André	Bât 502 Tél 82 33
Sciences et techniques du déchet	<u>NAVARRO Alain</u>	Bât 404 Tél 84 30
Signal, Image, Parole	GIMENEZ Gérard	Bât 502 Tél 83 32
Thermique et énergétique	LALLEMAND Monique	Bât 404 Tél 81 54

Les responsables soulignés sont également responsables généraux

INSTITUT NATIONAL DES SCIENCES APPLIQUEES DE LYON

Directeur : J.ROCHAT

Professeurs :

S.	AUDISIO	PHYSICOCHEMIE INDUSTRIELLE
J.C.	BABOUX	G.E.M.P.P.M.
J.	BAHUAUD	MECANIQUE DES SOLIDES
B.	BALLAND	PHYSIQUE DE LA MATIERE
M.	BARBIER	PHYSIQUE DE LA MATIERE
G.	BAYADA	MOD. MAT. CAL. SCIEN../L.M.C.
C.	BERGER (Melle)	PHYSIQUE DE LA MATIERE
M.	BETEMPS	AUTOMATIQUE INDUSTRIELLE
J.M.	BLANCHARD	L.A.E.P.S.I.
C.	BOISSON	VIBRATIONS ACOUSTIQUES
M.	BOIVIN	MECANIQUE DES SOLIDES
H.	BOTTA	EQUIPE DEVELOPPEMENT URBAIN
G.	BOULAYE	I.F.
J.	BRAU	CETHIL/THERMIQUE DU BATIMENT
M.	BRISAUD	GENIE ELECTRIQUE ET FERROELECTRICITE
M.	BRUNET	MECANIQUE DES SOLIDES
J.C.	BUREAU	THERMODYNAMIQUE APPLIQUEE
J.P.	CHANTE	COMPOSANTS DE PUISSANCE ET APPLICATIONS
B.	CHOCAT	U.R.G.C. HYDROLOGIE URBAINE
B.	CLAUDEL	L.A.E.P.S.I.
M.	COUSIN	U.R.G.C. STRUCTURE
M.	DIOT	THERMODYNAMIQUE APPLIQUEE
A.	DOUTHEAU	CHIMIE ORGANIQUE
J.C.	DUPUY	PHYSIQUE DE LA MATIERE
H.	EMPTOZ	R.F.V.
C.	ESNOUF	G.E.M.P.P.M.*
G.	FANTOZZI	G.E.M.P.P.M.*
J.	FAVREL	P.R.I.S.Ma.
G.	FERRARIS-BESSON	MECANIQUE DES STRUCTURES
Y.	FETIVEAU	GENIE ELECTRIQUE ET FERROELECTRICITE
L.	FLAMAND	MECANIQUE DES CONTACTS
P.	FLEISCHMANN	G.E.M.P.P.M.*
A.	FLORY	L.I.S.I.
R.	FOUGERES	G.E.M.P.P.M.*
F.	FOUQUET	G.E.M.P.P.M.*
L.	FRECON	I.F.
R.	GAUTHIER	PHYSIQUE DE LA MATIERE
M.	GERY	CETHIL/EQ. THERMIQUE DU BATIMENT
G.	GIMENEZ	C.R.E.A.T.I.S.
P.	GONNARD	GENIE ELECTRIQUE ET FERROELECTRICITE
M.	GONTRAND	L.C.P.A.
G.	GRANGE	GENIE ELECTRIQUE ET FERROELECTRICITE
G.	GUENIN	G.E.M.P.P.M.*
M.	GUICHARDANT	BIOCHIMIE ET PHARMACOLOGIE
G.	GUILLOT	PHYSIQUE DE LA MATIERE
A.	GUINET	PRISMa
J.L.	GUYADER	VIBRATIONS-ACOUSTIQUE
D.	GUYOMAR	GENIE ELECTRIQUE ET FERROELECTRICITE
J.M.	JOLION	R.F.V.
J.F.	JULLIEN	U.R.G.C. STRUCTURES
A.	JUTARD	AUTOMATIQUE INDUSTRIELLE
R.	KASTNER	U.R.G.C. GEOTECHNIQUE
H.	KLEIMANN	GENIE ELECTRIQUE ET FERROELECTRICITE
J.	KOULOUMDJIAN	L.I.S.I.
M.	LAGARDE	BIOCHIMIE ET PHARMACOLOGIE
M.	LALANNE	MECANIQUE DES STRUCTURES
A.	LALLEMAND	CETHIL/EQ. ENERGETIQUE ET THERMIQUE
M.	LALLEMAND (Mme)	CETHIL/EQ. ENERGETIQUE ET THERMIQUE
P.	LAREAL	U.R.G.C. GEOTECHNIQUE
A.	LAUGIER	PHYSIQUE DE LA MATIERE
CH.	LAUGIER	BIOCHIMIE ET PHARMACOLOGIE
P.	LEJEUNE	G.M.M.I.C.
A.	LUBRECHT	MECANIQUE DES CONTACTS

Y.	MARTINEZ	L.3I.
H.	MAZILLE	PHYSICOCHEMIE INDUSTRIELLE
J.	MERLE	G.E.M.P.P.M.*
J.	MERLIN	G.E.M.P.P.M.*
J.P.	MILLET	PHYSICOCHEMIE INDUSTRIELLE
M.	MIRAMOND	U.R.G.C. HYDROLOGIE URBAINE
N.	MONGEREAU	U.R.G.C. GEOTECHNIQUE
R.	MOREL	MECANIQUE DES FLUIDES
P.	NARDON	BIOLOGIE APPLIQUEE
A.	NAVARRO	L.A.E.P.S.I.
M.	OTTERBEIN	L.A.E.P.S.I.
J.P.	PASCAULT	MATERIAUX MACROMOLECULAIRES
J.	PERA	U.R.G.C. MATERIAUX
G.	PERACHON	THERMODYNAMIQUE APPLIQUEE
J.	PEREZ	G.E.M.P.P.M.*
P.	PINARD	PHYSIQUE DE LA MATIERE
J.M.	PINON	L.I.S.I.
D.	PLAY	C.A.S.M.
J.	POUSIN	MODEL. MATH. ET CALCUL SCIEN.
P.	PREVOT	G.R.A.C.I.M.P.
R.	PROST	C.R.E.A.T.I.S.
M.	RAYNAUD	CETHIL/EQ. TRANSFERT INTERFACES MATERIAUX
J.M.	REYNOUARD	U.R.G.C. STRUCTURES
J.	ROBERT-BAUDOUY (Mme)	G.M.M.I.C.
D.	ROUBY	G.E.M.P.P.M.*
P.	RUBEL	L.I.S.I.
C.	RUMELHART	MECANIQUE DES SOLIDES
J.F.	SACADURA	CETHIL/EQ. TRANSFERTS INTERFACES MATERIAUX
H.	SAUTEREAU	MATERIAUX MACROMOLECULAIRES
S.	SCAVARDA	AUTOMATIQUE INDUSTRIELLE
D.	THOMASSET	AUTOMATIQUE INDUSTRIELLE
M.	TROCCAZ	GENIE ELECTRIQUE ET FERROELECTRICITE
R.	UNTERREINER	C.R.E.A.T.I.S.
J.	VERON	L.A.E.P.S.I.
G.	VIGIER	G.E.M.P.P.M.*
A.	VINCENT	G.E.M.P.P.M.*
P.	VUILLERMOZ	PHYSIQUE DE LA MATIERE

Directeurs de recherche C.N.R.S. :

A.	ANKER	CHIMIE ORGANIQUE
Y.	BERTHIER	MECANIQUE DES CONTACTS
P.	CLAUDY	THERMODYNAMIQUE APPLIQUEE
N.	COTTE-PATTAT (Mme)	G.M.M.I.C.
P.	FRANCIOSI	G.E.M.P.P.M.*
M.A.	MANDRAND-BERTHELOT (Mme)	G.M.M.I.C.
M.	MURAT	G.E.M.P.P.M.*
J.F.	QUINSON	G.E.M.P.P.M.*
A.	ROCHE	MATERIAUX MACROMOLECULAIRES

Directeurs de recherche I.N.R.A. :

G.	BONNOT	BIOLOGIE APPLIQUEE
G.	FEBVAY	BIOLOGIE APPLIQUEE
S.	GRENIER	BIOLOGIE APPLIQUEE

Directeurs de recherche I.N.S.E.R.M. :

A-F.	PRIGENT (Mme)	BIOLOGIE ET PHARMACOLOGIE
I.	MAGNIN (Mme)	C.R.E.A.T.I.S.

REMERCIEMENTS

Ce travail a été effectué dans le cadre d'une collaboration franco-canadienne, dans l'équipe vision industrielle du Laboratoire d'Automatique Industrielle de l'INSA de Lyon pendant deux ans, ainsi qu'à l'Institut des Technologies de l'Information du Conseil National de Recherche du Canada situé à Ottawa lors de plusieurs séjours d'une durée globale d'un an.

Je voudrais exprimer ma reconnaissance au Pr. Alain Jutard, directeur du LAI et président du jury de cette thèse, pour l'accueil qu'il m'a réservé dans son laboratoire, pour ses encouragements, et pour les remarques judicieuses dont il a fait part lors de la relecture de ce manuscrit.

Je remercie vivement Jacques Domey, directeur du groupe des Technologies de l'Information Visuelle du CNRC, pour m'avoir permis de m'intégrer à son laboratoire et pour l'accueil chaleureux qu'il m'a prodigué. Que Marc Rioux soit également remercié de m'avoir intégrée dans son équipe.

Je tiens à exprimer toute ma gratitude à Hervé-Tanneguy Redarce, Maître de Conférence HdR à l'INSA de LYON, directeur de l'équipe vision industrielle du LAI et co-directeur de thèse, pour sa contribution scientifique, sa grande disponibilité et ses encouragements tout au long du déroulement de cette thèse. Que Flavio Prieto, étudiant de DEA, soit également remercié pour son aide lors de la phase expérimentale finale de ce travail.

Mes vifs remerciements vont également à Pierre Boulanger, agent de recherche du CNRC et co-directeur de thèse, pour son importante contribution scientifique et ses judicieux conseils.

Je remercie vivement messieurs Michel Dhome, directeur de recherche CNRS au LASMEA de Clermont-Ferrand, et Serge Tichkiewitch, professeur à l'Institut National Polytechnique de Grenoble, d'avoir accepté de participer à ce jury en tant que rapporteurs, et d'avoir examiner mon mémoire.

Je remercie également les autres membres du jury, Mr Philippe Bolon, professeur à l'Université de Savoie, et Mr Daniel Pyzak, directeur marketing Matra-Datavision, pour l'intérêt dont ils ont témoigné en examinant mes travaux.

Des remerciements particuliers vont à Takeshi Masuda chercheur de l'ElectroTechnical Laboratory (ETL) de Tsukuba au Japon, et à Guy Godin du CNRC, pour leur précieuse aide scientifique et leur grande sympathie.

Je remercie Mr Joel Favrel, directeur de l'Atelier Interétablissement de Productique Rhône Alpes Ouest (AIP-RAO), de m'avoir permis l'accès aux ressources de l'AIP, et Mr Achille Valisolalao, ingénieur système de l'AIP, pour ses conseils autour du système de CFAO Unigraphics.

Que l'AIP RAO, et l'IIT soient également remerciés pour nous avoir fourni les pièces mécaniques utilisées dans cette étude, et la société Hymarc (Ottawa), l'IIT, et l'équipe Métrologie du CAST de l'INSA de Lyon, pour nous avoir fourni les fichiers de points de mesure 3D de ces pièces.

Que trouve ici mes sincères remerciements pour leur aide technique et leur disponibilité, Pierre Gariepy et Luc Cournoyer du CNRC. Merci également à Patrick Clarysse du CREATIS pour ses conseils en optimisation non-linéaire, à Bernard Barbier du CREATIS pour son "prêt" de cpu, à Jean Pierre Taboy pour l'utilisation de l'imprimante laser couleur d'Eurinsa, et à Mireille Rubin-Delanchy pour ses nombreux conseils graphiques.

Je n'oublie pas le personnel du LAI et de l'IIT : Professeurs, enseignants, secrétaires, chercheurs, stagiaires et techniciens; je leur exprime ma reconnaissance pour leur sympathie, leur disponibilité et leur accueil. Je tiens aussi à remercier tout particulièrement mes camarades thésards du LAI pour l'amitié et le soutien qu'ils m'ont témoignés.

Merci enfin à la Conférence des Grandes Ecoles pour sa participation financière sans laquelle cette collaboration franco-canadienne n'aurait été possible.

TABLE DES MATIERES

Introduction Générale.....	13
Chapitre I : Modélisation 3D.....	17
I. Généralités.....	17
I.1. Construction d'un modèle.....	18
I.1.1. Adaptation d'un modèle préexistant.....	18
I.1.2. Création d'un modèle.....	18
I.2. Evaluation d'un modèle.....	19
II. Courbes et surfaces pour la modélisation.....	19
II.1. Courbes et surfaces polynomiales.....	19
II.2. Courbes splines.....	21
II.2.1. Fonctions splines polynomiales.....	21
II.2.2. Fonctions splines sous tension.....	21
II.3. Courbes et surfaces polynomiales de Bernstein-Bézier.....	22
II.4. Courbes et surfaces B-Spline.....	24
III. Modélisation géométrique.....	25
III.1. Modélisation filaire.....	25
III.1.1. Modélisation filaire.....	26
III.1.2. Représentation filaire.....	28
III.1.3. Limitations de la représentation filaire.....	28
III.2. Modélisation surfacique.....	29
III.3. Modélisation volumique.....	31
III.3.1. Enumération de l'occupation spatiale.....	32
III.3.2. Représentation par volumes élémentaires.....	33
III.3.3. Géométrie Solide Constructive (CSG).....	34
III.3.4. Représentation par balayage.....	35
IV. Modèles vision.....	35
IV.1. Points de surface, Normales, Courbures.....	36
IV.2. Modèle général de représentation multi-vues.....	37
IV.2.1. Sphère de vues.....	38
IV.2.2. Graphes d'aspect.....	38
IV.2.3. Travaux récents.....	39
IV.3. Image Gaussienne Etendue.....	40
IV.4. Polygonisation de Voronoï, triangulation de Delaunay.....	41
IV.5. Décomposition en primitives.....	41

V. Modèles déformables.....	42
V.1. Surfaces Superquadriques, Hyperquadriques et GEONS	42
V.1.1. Superquadriques.....	42
V.1.2. Superquadriques déformées	42
V.1.3. Hyperquadriques et hyperquadriques hybrides.....	44
V.1.4. GEONS.....	46
V.2. Approche physique de modélisation	47
V.2.1. Maillage adaptatif.....	48
V.2.2. Modèles de Terzopoulos/ Metaxas	49
V.2.3. Représentation modale (Sclaroff).....	49
V.2.4. Snakes	50
V.2.5. Déformations libres (Free Form Deformations - FFD).....	52
Conclusion.....	52
Chapitre II : Mise en Correspondance	54
I. Introduction	54
II. Mise en correspondance 2D.....	55
II.1. Mise en correspondance multi-modale	55
II.2. Mise en correspondance de primitives	56
II.3. Mise en correspondance d'images prises de différents points de vue	56
II.4. Mise en correspondance temporelle.....	56
II.5. Types de différences entre images	57
III. Mise en correspondance 3D.....	57
III.1. Représentation de la transformation rigide.....	58
III.2. Choix des caractéristiques à mettre en correspondance.....	59
III.3. Calcul de distance inter-primitives.....	61
III.4. Réduction du temps de calcul de distance.....	61
III.4.1. Octrees.....	62
III.4.2. <i>k-D-Trees (K-Dimensional binary Trees) [Zhang 94]</i>	62
III.4.3. <i>Octree-spline [Lavallée 95]</i>	62
III.4.4. <i>Points cachés [Simon 94]</i>	63
III.5. Méthodes de mise en correspondance 3D	63
III.6. Estimateurs plus spécifiques.....	64
III.7. Choix de la transformation initiale	64
IV. Méthode de mise en correspondance.....	66
IV.1. Principe de la méthode de mise en correspondance	66
IV.2. Algorithme $LMS(R_k^I, R^II)$	68
IV.3. Algorithme $ICP(P, \bar{T}, R^II)$	68
IV.4. Estimation de la transformation rigide utilisant les quaternions	70
Conclusion.....	71

Chapitre III : Données 3D et Modèles CAO 72

I. Données 3D.....	72
I.1. Système d'acquisition des images 3D.....	73
I.2. Mise en correspondance des images 3D.....	75
I.3. Bruit de mesure sur les points 3D.....	76
II. Modèles CAO.....	77
II.1. Modèle triangulé STL.....	77
II.1.1. Contraintes de la modélisation STL.....	78
II.1.2. Exemples de triangulation STL.....	78
II.1.3. Calcul de distance point/modèle STL.....	80
II.2. Modèle surface NURBS découpée.....	82
II.2.1. Définition.....	82
II.2.2. Calcul de distance point/surface NURBS découpée.....	84
II.3. Compromis modèle STL/modèle NURBS.....	88
II.4. Complémentarité modèle STL/modèle NURBS.....	88
Conclusion.....	89

Chapitre IV : Evaluation de la Méthode de Mise en correspondance..... 90

Introduction.....	90
I. Justesse et fidélité.....	91
I.1. Premier essai.....	92
I.2. Deuxième essai.....	93
II. Sources des erreurs de calcul.....	95
II.1. Erreur de la méthode de mise en correspondance seule.....	95
II.1.1. Objet Support.....	95
II.1.2. Objet Cylindre.....	96
II.2. Erreur de modélisation STL.....	97
II.2.1. Objet "Cylindre".....	97
II.2.2. Objet "Support".....	100
III. Comportement de la méthode en fonction de ses paramètres internes.....	101
III.1. Paramètres N_s et N_t	102
III.1.1. Etude théorique.....	102
III.1.2. Vérification expérimentale.....	103
III.2. Diminution du temps de calcul.....	106
IV. Robustesse au bruit.....	108
Conclusions.....	109

Chapitre V : Inspection Automatique 111

I. Cotes et tolérances.....	112
I.1. Pièces de forme non-complexe.....	112
I.1.1. Tolérances de forme.....	113
I.1.2. Tolérances d'orientation.....	114
I.1.3. Tolérances de position.....	115
I.1.4. Tolérances de battement.....	115

I.2. Pièces de forme complexe.....	115
I.2.1. <i>Matrice de dimension</i>	115
I.2.2. <i>Matrice de tolérance</i>	116
I.2.3. <i>Tolérance de forme des repères de référence</i>	117
I.2.4. <i>Tolérance de forme générale</i>	118
I.3. Intégration des tolérances en CAO.....	119
I.3.1. <i>Espace vectoriel de variables modèles</i>	120
I.3.2. <i>Modèle Offset</i>	120
II. Inspection Visuelle Automatique.....	121
II.1. Définition de l'inspection visuelle automatique.....	122
II.2. Types d'images utilisées en inspection visuelle.....	123
II.2.1. <i>Images binaires</i>	123
II.2.2. <i>Images à niveaux de gris</i>	124
II.2.3. <i>Images couleurs</i>	124
II.2.4. <i>Images de profondeur</i>	124
II.3. Utilisation de données CAO pour l'inspection.....	124
II.4. Exemples d'inspection automatique visuelle.....	126
III. Méthode d'inspection d'objets de forme complexe.....	127
III.1. Tolérances globales.....	128
III.1.1. <i>Tolérances de positionnement et d'orientation de chaque surface</i>	128
III.1.2. <i>Tolérances de forme globale de chaque surface</i>	129
III.2. Tolérances de forme locales.....	130
III.2.1. <i>Calcul des paramètres de la surface</i>	131
III.2.2. <i>Calcul des propriétés géométriques</i>	132
III.2.3. <i>Vérification des tolérances</i>	133
IV. Méthode d'inspection d'objets de formes non-complexes.....	134
IV.1. Défauts d'usinage courants.....	134
IV.2. Raisonnement point par point.....	135
IV.2.1. <i>Etude des défauts de la pièce étudiée</i>	136
IV.2.2. <i>Comparaison acquisition par machine à mesurer et capteur laser</i>	139
IV.3. Raisonnement par caractéristique.....	139
Conclusion.....	139
Chapitre VI : Applications	141
I. Reconnaissance de formes	141
I.1. Polyèdres.....	142
I.2. Supports.....	142
II. Inspection d'objets industriels	143
II.1. Procédés de visualisation.....	143
II.1.1. <i>Coloration des triangles</i>	143
II.1.2. <i>Carte de texture</i>	144
II.1.3. <i>Tracé des segments d'erreur</i>	145
II.2. Exemples.....	146
II.2.1. <i>Polyèdre</i>	146
II.2.2. <i>Plaque</i>	146
II.2.3. <i>Support</i>	146
II.2.4. <i>Téléphone</i>	152
II.2.5. <i>IMS</i>	153
II.2.6. <i>Pièce1</i>	154

III. Préparation à la finition d'une pièce ébauchée	155
Conclusion	155
Conclusion Générale	157
Annexe A	160
Annexe B	172
Annexe C	182
Annexe D	193
Annexe E.....	195
Références Bibliographiques.....	205

Introduction Générale

La production croissante d'objets présentant des surfaces gauches, que ce soit pour des raisons fonctionnelles ou de design, et l'évolution des techniques de fabrication, ont créé un besoin dans le domaine de l'inspection automatique de pièces complexes. Ce type de contrôle nécessite une définition géométrique très fidèle de l'objet inspecté, de nombreux points de mesure suffisamment précis, et des règles d'inspection de ces surfaces clairement définies.

L'utilisation de machines à mesurer tridimensionnelles, dont la précision de mesure atteint le μm , et l'avènement récent des capteurs télémétriques alliant précision de mesure et rapidité d'acquisition, permettent d'obtenir des mesures 3D constituant une description explicite des surfaces de l'objet. Par ailleurs, la connaissance du modèle CAO ayant servi à la réalisation de l'objet fournit une description exacte et complète de sa géométrie. Nous proposons donc une méthode d'inspection automatique de pièces comprenant des surfaces complexes, à partir d'un modèle CAO (format IGES ou STL) et de données 3D fournies par un capteur télémétrique ou par une machine à mesurer.

Nous choisissons dans un premier temps un modèle géométrique permettant de représenter l'objet étudié. Pour cela, nous établissons un état de l'art complet sur les modèles géométriques déformables ou non-déformables existants, pour des applications spécifiques telles que la Vision et la CAO. A partir de cette classification, nous sélectionnons deux modèles surfaciques : un modèle exact et complexe (surface NURBS découpées) et un modèle triangulé simple, facile à manipuler (STL). D'un autre côté, nous disposons d'un ensemble de points 3D, fourni soit par une machine à mesurer tridimensionnelle, soit par la mise en correspondance mécanique de différentes vues 3D acquises par un capteur télémétrique.

Le modèle géométrique et les données 3D sont exprimés initialement dans des repères indépendants (repère de modélisation et repère de mesure). Afin de comparer ces deux formes, il est nécessaire de les "superposer", c'est-à-dire de minimiser la distance globale points 3D/modèle géométrique, opération appelée "mise en correspondance". D'un point de

vue mathématique, la mise en correspondance des points 3D avec le modèle géométrique consiste à déterminer les paramètres de la transformation rigide (trois paramètres pour la translation et trois paramètres pour la rotation) à appliquer sur les points 3D, de telle sorte que l'on minimise un critère de distance entre les deux formes ; un état de l'art concernant ces techniques est présenté. Nous proposons une méthode de mise en correspondance générale automatique, robuste (jusqu'à 50% de points aberrants) et capable de mettre en correspondance deux formes quelconques dans des positions initiales relatives quelconques. En fonction du type de formes manipulées, cette méthode nécessite la connaissance du calcul de distances interprimitives. Pour cela, nous définissons un calcul de distance point/ensemble de triangles et point/surface NURBS découpées.

Une fois établie la mise en correspondance entre les données 3D et le modèle CAO, en fonction de l'erreur constatée ϵ , de multiples conclusions et actions sont possibles (figure 0.1).

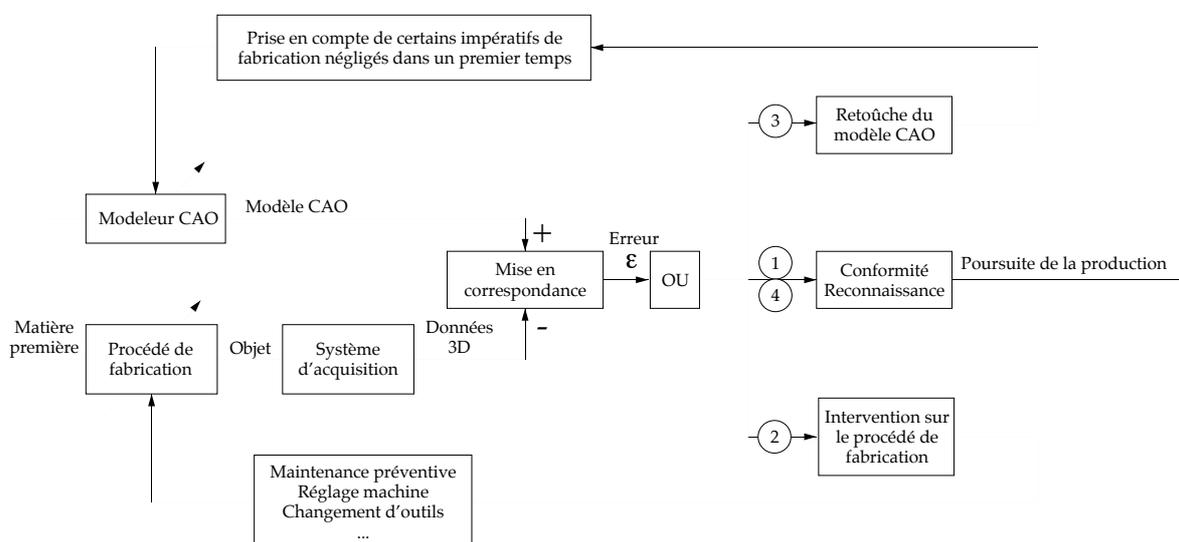


Figure 0.1 : Comparaison modèle CAO/objet usiné et rétroaction sur le procédé de fabrication ou sur le modèle CAO.

En effet, on peut envisager plusieurs stratégies d'exploitation des résultats de la mise en correspondance, à savoir :

1. La pièce est conforme ; elle est donc acceptée comme bonne et exploitée dans la suite du processus.
2. Des erreurs sont constatées, indiquant une dégradation du procédé de fabrication (usure des outils, dérive,...). Le procédé de fabrication est rectifié en conséquence.
3. Non-conformité de la pièce due à une non-prise en compte d'impératifs de fabrication (retrait des pièces moulées,...). Le modèle CAO est modifié, pour que les futures pièces usinées soient conformes.

4. La pièce est reconnue parmi une famille d'objets (reconnaissance de formes).

Nous nous sommes limités dans notre travail aux conclusions issues de la mise en correspondance, sans rétroaction sur les paramètres du procédé ou sur le modèle CAO. En particulier, une application importante de la mise en correspondance réside dans l'inspection automatique d'objets présentant des surfaces complexes ou non-complexes. Nous établissons, dans un premier temps, un état de l'art dans les domaines de la vérification des tolérances et de l'inspection visuelle automatique.

Nous présentons ensuite notre propre méthode d'inspection d'objets présentant des surfaces complexes. L'étude bibliographique a montré que les méthodes existantes se contentent le plus souvent de tester que la distance entre chaque point et sa surface nominale, contrairement à notre méthode qui réalise une inspection complète des objets. Cette phase d'inspection se décompose en une inspection globale, vérifiant les tolérances du positionnement, de l'orientation et de la forme des surfaces de l'objet, et en une inspection locale, comparant les courbures et normales en chaque point 3D avec sa surface nominale. Nous présentons également une méthode d'inspection des objets comportant des surfaces simples (plans, portions de cylindre), basée sur le respect des cotes et des différentes tolérances (forme, position, orientation). Un rapport d'inspection est finalement engendré pour chacune des pièces inspectées; on dispose également de la possibilité de visualiser graphiquement la localisation des erreurs.

A partir de la mise en correspondance des deux formes, d'autres applications sont possibles. L'une des plus simples est la reconnaissance de formes : disposant des données 3D d'un objet à reconnaître parmi une famille d'objets, nous réalisons leur mise en correspondance avec chacun des modèles géométriques disponibles. Le classifieur de type distance euclidienne, qui donne la plus petite distance, permet de déterminer l'objet à reconnaître.

L'intégration de ces différentes parties dans le processus d'inspection automatique industrielle, ou de reconnaissance de formes est illustrée par la figure 0.2 ; les rectangles grisés correspondent aux contributions de notre travail de thèse.

Ainsi, après le chapitre I qui expose l'état de l'art sur la modélisation 3D, le chapitre II est consacré à la présentation de la méthode générale de mise en correspondance, tandis que le chapitre III est relatif aux données 3D, aux modèles CAO manipulés et aux calculs de distances associés. Nous évaluons au chapitre IV les capacités de la méthode de mise en correspondance proposée, en fonction de ses paramètres internes et du type de données manipulées. Au chapitre V, après un état de l'art concernant l'inspection visuelle automatique, nous proposons notre propre méthode d'inspection pour des objets présentant des surfaces complexes. Le chapitre VI est essentiellement consacré aux applications que sont la Reconnaissance de formes et l'Inspection.

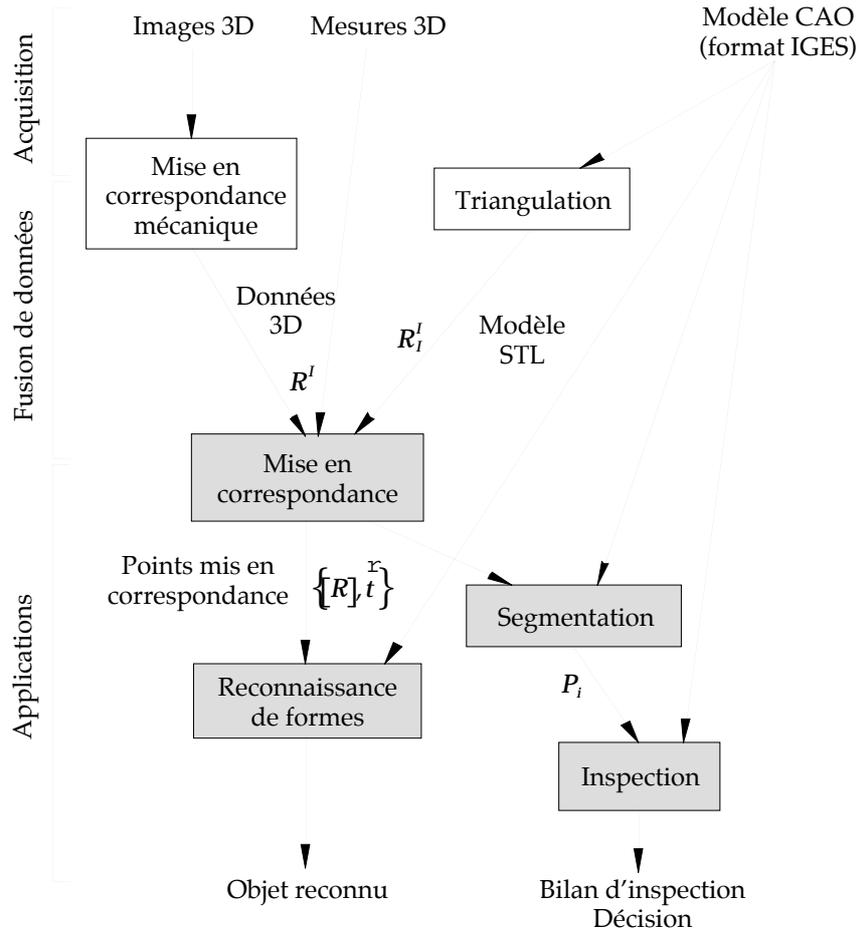


Figure 0.2 : Processus d'inspection automatique industrielle.

Chapitre I

Modélisation 3D

L'objectif de ce chapitre est de faire un tour d'horizon des modèles 3D existants afin de sélectionner le mieux adapté pour la mise en correspondance. Dans un premier paragraphe nous donnerons quelques généralités sur la création et l'évaluation des modèles 3D. Nous introduirons ensuite les différents types de courbes et de surfaces utilisées pour la modélisation, puis nous présenterons les modèles géométriques les plus utilisés, ainsi que les modèles générés par et pour la Vision par Ordinateur. Finalement nous présenterons des développements récents sur les modèles déformables.

I. Généralités

Les domaines d'application de la vision par ordinateur sont divers : reconnaissance de formes, robotique mobile, imagerie médicale, réalité virtuelle, inspection d'objets industriels, poursuite d'objets en temps réel, etc... Toutes ces applications nécessitent à la fois une bonne représentation interne de la tâche à effectuer ainsi que des données à manipuler, d'où la nécessité de définir un modèle de référence. Ce modèle peut être : soit de type descriptif, dont les informations complètes permettent aussi bien d'effectuer l'inspection d'objets industriels que de générer des images de synthèse, soit de type discriminatoire, où les informations du modèle sont utilisées pour des applications de type Reconnaissance de Formes.

Nous allons nous intéresser maintenant aux modes de création des modèles 3D, ainsi qu'aux critères utilisés pour leur évaluation.

I.1. Construction d'un modèle

Pour construire un modèle approprié à l'application [Flynn 91] [Givert 88], il existe deux possibilités. La plus simple est d'adapter un modèle préexistant, l'autre consiste à créer le modèle à partir des données disponibles.

I.1.1. Adaptation d'un modèle préexistant

Dans beaucoup d'applications (essentiellement industrielles) les objets à traiter sont déjà prédéfinis et disponibles dans une base de données (base de données CAO, par exemple). Ces bases de données qui ont l'avantage d'être complètes et exactes peuvent être ainsi exploitées dans leur intégralité ou dégradées, adaptées, améliorées ou restructurées en fonction de l'application.

I.1.2. Création d'un modèle

Au contraire, lorsqu'il n'existe pas de modèle de l'objet traité, il est nécessaire d'en construire un, en exploitant les informations données par des capteurs de vision 2D ou 3D.

Vision 2D

Cette construction du modèle d'un objet, décrite par Givert [Givert 88], s'appuie sur l'utilisation de caractéristiques extraites d'images 2D de l'objet, prises à des points de vue différents (coins, arêtes, ...), entre lesquelles sont établies des relations structurelles. A partir de ces caractéristiques, est bâti un modèle homogène aux données de l'image de la scène à traiter.

Vision 3D

La vision 3D fournit pour un objet un ensemble de points 3D décrivant ses surfaces. A partir de ces points 3D, peut être construite une triangulation de l'objet, le nombre de triangles créés étant optimisé en fonction de la précision du modèle souhaitée [Hoppe 92]. On peut également engendrer des modèles surfaciques plus complexes, en segmentant l'ensemble de points 3D. Chacun des sous-ensembles obtenus est interpolé par une surface plus ou moins complexe (plan, ..., surface NURBS). Cette méthode, appelée Ingénierie inverse, permet à partir de la vision 3D, de produire un modèle géométrique de l'objet numérisé, au format des logiciels de CAO [Boulanger 93a] [Boulanger 93b].

Le modèle ainsi constitué, en vision 2D ou 3D, est une approximation de l'objet réel. L'efficacité de l'application utilisant le modèle dépendra de la richesse et de la précision des détails de la description ainsi obtenue, étroitement liées à la structure et à la qualité de mesure du capteur utilisé. Cette technique permet la constitution de modèles descriptifs d'objets non manufacturés, ou de forme complexe difficilement modélisables par d'autres

méthodes. Cependant la lourdeur de construction, la cohérence et la précision de description de l'objet limitent parfois l'utilisation de tels modèles.

I.2. Evaluation d'un modèle

Il existe un certain nombre de critères [Bhanu 87] [Kholadi 91] [Brown 81] permettant d'évaluer les propriétés d'un modèle, en voici quelques-uns :

- Capacité : Quels types d'objets peut-on représenter ?
- Accessibilité : Peut-on obtenir facilement un modèle à partir d'une image ?
- Concision : Peut-on décrire les objets efficacement ?
- Unicité : La description de l'objet est-elle unique ?
- Stabilité : Peut-on calculer les paramètres du modèle avec une méthode numérique stable ?
- Sensibilité : Le modèle permet-il de distinguer deux objets proches ?

Avant d'aborder une description des différents types de modèles 3D existants, il est nécessaire d'introduire quelques définitions mathématiques concernant les courbes et surfaces utilisées pour la modélisation.

II. Courbes et surfaces pour la modélisation

La création d'un modèle 3D nécessite le choix d'entités géométriques (courbes et/ou surfaces) plus ou moins complexes. Nous allons faire le tour d'horizon dans un ordre de complexité croissante des courbes et surfaces les plus utilisées en modélisation, en soulignant pour chacune d'entre elles ses avantages et inconvénients.

II.1. Courbes et surfaces polynomiales

Ces courbes et surfaces ont été décrites par [Besl 88] [Farin 92b] [Faux 81] [Peroche 90]. Il est vrai que les polynômes de haut degré peuvent décrire des courbes complexes, mais ils demandent un grand nombre de coefficients dont la signification physique est parfois difficile à maîtriser. D'autre part, plus le degré des polynômes est élevé plus il apparaît d'instabilités numériques. De ce fait, les courbes et surfaces cubiques (degré 3 maximum dans chaque direction), ont été reconnues comme étant un bon compromis de modélisation dans la plupart des applications. D'une manière générale, les courbes de degré maximum 2 sont

appelées courbes quadratiques et les courbes de degré maximum 3 sont appelées courbes cubiques.

Fergusson (1963) a été le premier à introduire l'utilisation des courbes cubiques paramétriques dans le domaine de l'aéronautique (Boeing Co).

Les segments de ces courbes paramétriques sont définis par des équations de la forme :

$$\vec{r} = \vec{r}(u) = \sum_{i=0}^n u^i \vec{a}_i$$

où n est le degré de la courbe, où le vecteur \vec{r} décrit l'ensemble des points de la courbe lorsque u varie de 0 à 1. Les vecteurs \vec{a}_i , sont les paramètres de la courbe.

Les surfaces paramétriques sont exprimées comme suit :

$$\vec{r} = \vec{r}(u, v) = \sum_{i=0}^n \sum_{j=0}^m u^i v^j \vec{a}_{ij}$$

Afin de mieux maîtriser localement la forme des courbes et surfaces, les points \vec{a}_i peuvent être pondérés. Des poids w_i sont introduits dans les polynômes de description qui deviennent rationnels, offrant aux courbes et surfaces des capacités descriptives supérieures aux polynômes non-rationnels. Un des avantages importants est que toutes les coniques peuvent être exactement représentées avec des polynômes rationnels de degré 2. Un autre avantage, pour les applications Vision et Graphique, est que la projection d'une courbe rationnelle est également une courbe rationnelle.

L'équation d'une courbe polynomiale rationnelle de degré n est :

$$\vec{r} = \vec{r}(u) = \frac{\sum_{i=0}^n w_i u^i \vec{a}_i}{\sum_{i=0}^n w_i u^i}$$

où les w_i sont les poids qui pondèrent chaque point de contrôle du polygone de contrôle.

En 2D une surface polynomiale rationnelle de degrés m et n s'exprime ainsi :

$$\vec{r} = \vec{r}(u, v) = \frac{\sum_{i=0}^n \sum_{j=0}^m w_{ij} u^i v^j \vec{a}_{ij}}{\sum_{i=0}^n \sum_{j=0}^m w_{ij} u^i v^j}$$

II.2. Courbes splines

Ces courbes paramétriques [Besl 88] [Farin 92b] ont été introduites en CAO par Ferguson en 1963. C'est Schoenberg (1946) puis Whitney (1959) qui présentent les premières idées sur les fonctions splines, qui connurent un succès sous l'appellation "spline cubique". Ces splines (ou lattes élastiques) peuvent être considérées comme des courbes contraintes à passer par des points arbitraires (noeuds) et leur équation du troisième degré est donnée par les équations de la mécanique des milieux continus.

II.2.1. Fonctions splines polynomiales

Soit un intervalle d'abscisses $[a, t_0, \dots, t_n, b]$, une fonction spline polynomiale f de degré q relative aux n abscisses t_i est définie comme suit :

- f est un polynôme de degré $(2q - 1)$ dans chaque intervalle $]t_i, t_{i+1}[$.
- ◻ f est un polynôme de degré $(q - 1)$ dans les intervalles $[a, t_1[$ et $]t_n, b]$.
- La dérivée d'ordre $(2q - 2)$ est continue.

Connaissant un jeu de données z_i , la fonction f doit vérifier : $f(t_i) = z_i$

et minimiser le critère suivant : $\int_a^b (f^{(q)}(t))^2 dt$

II.2.2. Fonctions splines sous tension

Les fonctions splines sous tension [Besl 88] sont très utilisées en CAO. Dans ce cas, le critère à minimiser devient, après ajout d'une contrainte de forme :

$$\int_a^b [f''(t)]^2 + u^2 [f'(t)]^2 dt$$

et les fonctions-solutions sont de la forme :

$$\alpha + \beta \cdot x + \gamma \cdot sh(px) + \delta \cdot ch(qx)$$

II.3. Courbes et surfaces polynomiales de Bernstein-Bézier

Bézier (1970) a recombinaé les termes de la paramétrisation cubique de Fergusson afin que la signification physique des coefficients-vecteurs soit plus apparente [Bézier 87] [Bézier 88]. Les courbes de Bernstein-Bézier [Faux 81] [Farin 92a] [Farin 92b] de degré n ont la forme suivante :

$$\vec{r} = \vec{r}(u) = \sum_{i=0}^n B_i^n(u) \vec{r}_i$$

où n est le degré de la courbe et où les \vec{r}_i sont les vecteurs de positionnement des $(n+1)$ sommets P_0, P_1, \dots, P_n d'un polygone appelé polygone caractéristique généralisé (figure I.1).

Les polynômes $B_i^n(u) = \frac{n!}{(n-i)!i!} u^i (1-u)^{n-i}$ sont appelés polynômes de Bernstein.

L'équation d'une surface de Bézier est de la forme suivante :

$$\vec{r} = \vec{r}(u, v) = \sum_{i=0}^n \sum_{j=0}^m B_i^n(u) B_j^m(v) \vec{r}_{ij}$$

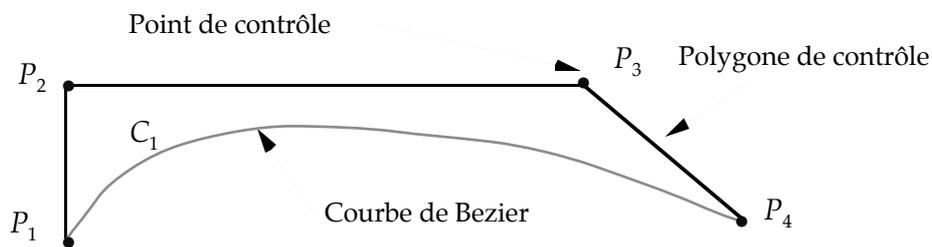


Figure I.1 : Courbe de Bézier, Points de contrôle et polygone de contrôle.

De même que pour les courbes et surfaces polynomiales rationnelles, on peut décrire des courbes et surfaces de Bézier rationnelles :

Courbe de Bézier rationnelle :

$$\vec{r} = \vec{r}(u) = \frac{\sum_{i=0}^n w_i B_i^n(u) \vec{r}_i}{\sum_{i=0}^n w_i B_i^n(u)}$$

Surface de Bézier rationnelle :

$$\vec{r} = \vec{r}(u, v) = \frac{\sum_{i=0}^n \sum_{j=0}^m w_{ij} B_i^n(u) B_j^m(v) \vec{r}_{ij}}{\sum_{i=0}^n \sum_{j=0}^m w_{ij} B_i^n(u) B_j^m(v)}$$

Avantages des courbes de Bézier :

- + Chaque point de la courbe est une combinaison convexe de tous les points de contrôle,
- + Pas de problème de non-continuité aux raccords,
- + Grande maniabilité des courbes,
- + Elles ne sont pas modifiées par un changement d'axe,
- + Les polynômes de Bernstein s'expriment de façon explicite.

Inconvénients de l'utilisation des courbes et surfaces de Bézier :

- Le déplacement d'un point de contrôle implique une modification complète de la courbe (figure I.2).
- Il est coûteux de rajouter un point.
- Le degré du polynôme augmente avec le nombre de points de contrôle. Ainsi, lorsque le nombre de points de contrôle est important, le calcul des fonctions de Bézier devient délicat.

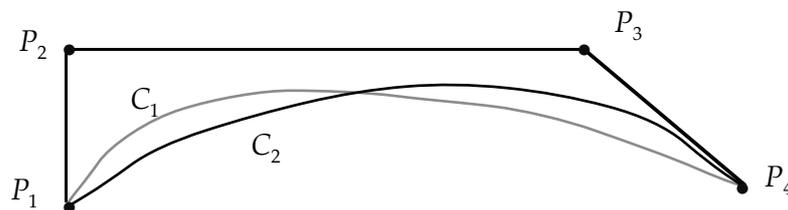


Figure I.2 : La courbe C_2 résulte de la déformation globale de la courbe C_1 lorsqu'on augmente le poids du point P_3 .

II.4. Courbes et surfaces B-Spline

Les courbes polynomiales segmentées "Basis Splines" (BSpline) étudiées en 1972 par De Boor offrent l'avantage de pouvoir changer de forme localement.

Equation d'une courbe B-Spline [Besl 88] [Farin 92a] [Farin 92b] :

$$\vec{r} = \vec{r}(u) = \sum_{i=0}^n N_i^n(u) \vec{r}_i$$

où les polynômes N_i^n sont définis de manière récursive :

$$N_i^n(u) = \frac{u - u_{i-1}}{u_{i+n-1} - u_{i-1}} N_i^{n-1}(u) + \frac{u_{i+k} - u}{u_{i+k} - u_i} N_{i+1}^{n-1}(u) \quad N_i^0(u) = \begin{cases} 1 & \text{si } u_{i-1} \leq u < u_i \\ 0 & \text{sinon} \end{cases}$$

et où le vecteur de paramètres $\{u_0, u_1, \dots, u_{N-1}, u_N\}$ est le vecteur de noeuds de la courbe.

Surface B-Spline :

$$\vec{r} = \vec{r}(u, v) = \sum_{i=0}^n \sum_{j=0}^m N_i^n(u) N_j^m(v) \vec{r}_{ij}$$

On appelle B-Splines uniformes, les courbes et surfaces pour lesquelles les vecteurs de noeuds sont décrits par une suite arithmétique. A l'opposé, on appelle B-Splines non uniformes, les fonctions de base et les courbes construites à l'aide d'un vecteur nodal où l'intervalle entre deux noeuds successifs n'est pas constant.

Avantages des courbes et surfaces B-Spline :

- + Leur capacité descriptive,
- + Les propriétés d'enveloppe convexe (Les points de contrôle de la courbe se situent toujours du même côté de la courbe),
- + l'invariance affine,
- + le contrôle local,
- + le degré indépendant du nombre de points de contrôle,
- + la définition récursive des N_i^n .

Un seul Inconvénient :

- Le nombre de paramètres.

Les courbes et surfaces B-Spline peuvent également être pondérées pour augmenter leur pouvoir descriptif; on obtient alors les courbes et surfaces B-Spline rationnelles.

Courbe B-Spline rationnelle :

$$\vec{r} = \vec{r}(u) = \frac{\sum_{i=0}^n w_i N_i^n(u) \vec{r}_i}{\sum_{i=0}^n w_i N_i^n(u)}$$

Surface B-Spline rationnelle :

$$\vec{r} = \vec{r}(u, v) = \frac{\sum_{i=0}^n \sum_{j=0}^m w_{ij} N_i^n(u) N_j^m(v) \vec{r}_{ij}}{\sum_{i=0}^n \sum_{j=0}^m w_{ij} N_i^n(u) N_j^m(v)}$$

Lorsque des vecteurs de noeuds non-uniformes sont utilisés, ces B-Splines rationnelles sont connues sous le nom de NURBS (Non-Uniform knot vector Rational B-Spline). Les NURBS peuvent représenter de façon exacte toutes les formes polygonales, toutes les coniques, toutes les courbes polynomiales paramétriques par morceaux ainsi que toute courbe complexe. Ces courbes et surfaces d'une grande capacité de description sont utilisées par les modeleurs des logiciels de CAO, et choisies comme format d'échange standard (IGES) entre systèmes de CAO depuis 1983. Nous les utiliserons donc comme modèles géométriques de surfaces.

Nous présentons maintenant trois catégories de modèles 3D : les modèles géométriques, les modèles vision ainsi que les modèles déformables.

III. Modélisation géométrique

Dans ce paragraphe, nous allons présenter les modèles géométriques les plus connus. Par nature, les objets sont à deux dimensions (surfaces), ou à trois dimensions (volumes). Un modèle géométrique est la représentation numérique d'un objet spatial. Il existe plusieurs approches de modélisation tridimensionnelle des objets spatiaux : la modélisation filaire, la modélisation surfacique, et la modélisation volumique.

III.1. Modélisation filaire

Le modèle filaire [Brady 82] [Brown 81] [Gigus 90] [Nguyen 92] [Shirai 87] a été historiquement le premier à être mis en oeuvre. Le modèle filaire d'un objet est un graphe

dont les points sont des points 3D de la surface de l'objet (typiquement des coins ou d'autres points remarquables) et les traits du graphe représentent les arêtes physiques de l'objet. Ces représentations ne contiennent pas d'informations sur les surfaces et ne sont pas suffisamment complètes.

Avantages :

- + Simple
- + Rapidité de visualisation
- + Nécessite un minimum de puissance de calcul

Inconvénients :

- Ambigu (on ne distingue pas le vide du plein)
- Possibilité de créer des solides sans aucun sens physique
- Impossibilité d'effectuer certains calculs (calcul de la masse, par exemple)
- Difficulté de résolution du problème de l'élimination des parties cachées

La représentation filaire reste encore très utilisée surtout pour la manipulation d'objets polyédriques et par extension pour certains objets non-polyédriques de forme simple.

III.1.1. Modélisation filaire

La modélisation filaire polyédrique [Filbois 93] est largement utilisée en vision par ordinateur, à cause de sa simplicité et comme bon support pour les informations géométriques et topologiques. Le modèle polyédrique peut être construit à partir des coins, des arêtes, des faces; où les coins sont des points 3D, les arêtes sont les segments linéaires entre les coins, et les faces sont les polygones plats délimités par une liste ordonnée d'arêtes ou de coins (figure I.3).

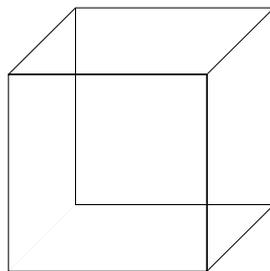


Figure I.3 : Modélisation filaire d'un cube.

A partir de cette modélisation, on distingue deux types de paramètres : ceux qui ne dépendent que de la primitive qu'ils caractérisent (paramètres intrinsèques) et ceux qui dépendent du contexte dans lequel se trouve la primitive (paramètres extrinsèques).

Paramètres intrinsèques :

- Pour les segments :
 - Longueur de chaque segment.
- Pour les facettes :
 - Surface,
 - Périmètre,
 - Normes des axes d'inertie,
 - Angles entre les segments constituant la facette,
 - Couples de segments parallèles ou perpendiculaires,
 - Longueurs des segments de la facette,
 - Nombre de segments.

Paramètres extrinsèques :

- Pour les segments :
 - Angles entre les segments,
 - Distances entre les segments.
- Pour les facettes :
 - Angles entre les normales des facettes,
 - Distance entre les centres de gravité des facettes,
 - Facettes voisines et nombre de facettes voisines.

Chaque arête de l'objet polyédrique peut être classée comme convexe ou concave suivant l'angle formé par les deux faces partageant cette arête. Une arête convexe peut être aussi considérée comme une arête occluse si d'un certain point de vue les faces qui se rencontrent le long d'une arête sont du même côté de l'arête.

III.1.2. Représentation filaire

Il existe une extension à la représentation filaire, appelée représentation filaire étiquetée [Gigus 90] (figure I.4), une arête convexe est alors représentée par un "+", une arête concave par un "-" et une arête ocluse par une flèche (lorsqu'on se déplace sur l'arête dans la direction indiquée par la flèche, les faces qui se rencontrent sur l'arête sont sur la droite de celle-ci).

Les arêtes se rencontrent aux jonctions. Une jonction est soit une projection d'un sommet en 3D, soit un point de rencontre de projection d'arête non concourantes appelée jonctions en T.

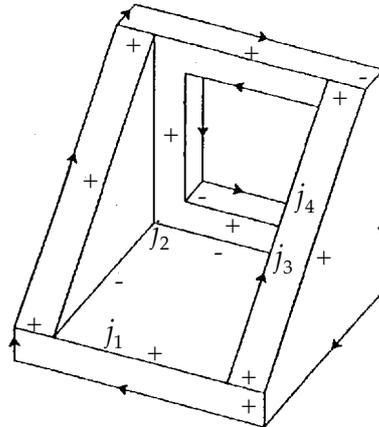


Figure I.4 : Une représentation filaire, ou j_1 et j_2 sont des jonctions d'arêtes et j_3 est une jonction en T.

III.1.3. Limitations de la représentation filaire

La représentation filaire s'applique surtout aux objets polyédriques, mais peut aussi s'appliquer à des objets constitués de portions de cônes, de sphères et de cylindres. Néanmoins l'addition de nouvelles étiquettes augmente considérablement le nombre de types de jonctions possibles.

Les techniques d'étiquetage représentent un outil puissant pour l'interprétation des représentations filaires, mais certaines représentations ne sont pas réalisables, ni interprétables (figure I.5).

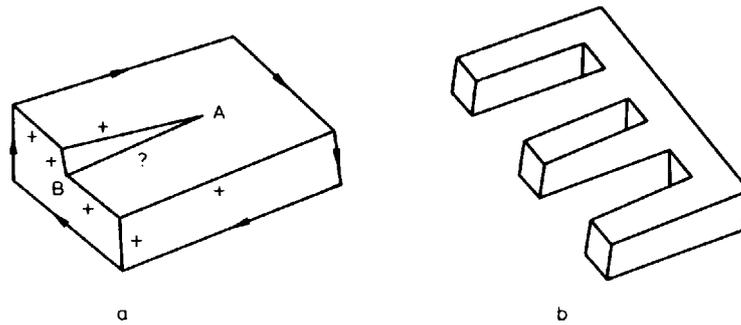


Figure I.5 : Représentations filaires qui ne peuvent pas être étiquetées (a) ou d'un objet impossible à réaliser (b).

Qamhiyah [Qamhiyah 93] a établi à partir de la représentation filaire un modèle dédié à la classification d'objets 3D. L'unité de description choisie est le contour, aussi bien de solide que de trou. Cette représentation a l'avantage d'être à la fois invariante en rotation et en translation. Basés sur la classification des contours entourant un trou ou un solide, les éléments de description définissent une relation entre deux contours de solides, un contour de solide et un contour de trou, deux contours de trous. La représentation de l'objet est alors une matrice où chaque ligne et chaque colonne représentent un contour. Chaque terme de la matrice donne le code de l'élément de description entre le contour i et le contour j , auquel est attribué un ordre de priorité.

III.2. Modélisation surfacique

Il faut distinguer deux types de modélisation surfacique; ou bien l'objet modélisé est décrit par une seule surface aussi complexe que nécessaire pour le représenter fidèlement, ou bien l'objet est représenté par un assemblage de plusieurs surfaces plus simples, mais auxquelles on devra adjoindre des informations de topologie. Les types de surfaces utilisées sont très divers, des surfaces planes jusqu'aux surfaces NURBS découpées. Le modèle surfacique comble une partie des défauts de la représentation filaire et très utile dans la conception de formes. Cependant, des problèmes d'intégrité et de cohérence du volume décrit peuvent se poser puisque chaque surface est décrite indépendamment des autres.

Avec les représentations par frontières ou BREP (Boundary REPresentation) [Bhanu 87] [Flynn 91] [Kholladi 91] [Peroche 90], la frontière de l'objet est explicitement décrite : Cette information est très utile en CAO pour l'usinage des surfaces, pour le calcul rapide des masses, des centres de gravité...; elle est aussi utile en infographie, où la plupart des algorithmes de visualisation supposent connues les surfaces des objets. Ce modèle doit comprendre deux types d'informations : des informations géométriques (coordonnées des points appelés noeuds et équations des faces), ainsi que des informations topologiques (façon dont sont liées entre elles les informations géométriques) (figures I.6,I.7).

Pour valider un modèle surfacique, chaque face ou morceau de surface doit vérifier un ensemble de propriétés, qui assurent la reconnaissance de l'objet en temps que volume occupant une portion de l'espace ainsi que son homogénéité :

- Une face ou un morceau de surface doit posséder une aire et ne pas comporter d'arêtes pendantes ni de points isolés,
- Toute face ou morceau de surface est un sous-ensemble d'une surface de base,
- Une face ou un morceau de surface d'un objet spatial est un sous-ensemble de la frontière de l'objet,
- L'union de toutes les faces ou de tous les morceaux de surfaces d'un objet doit être égale à la frontière de l'objet.

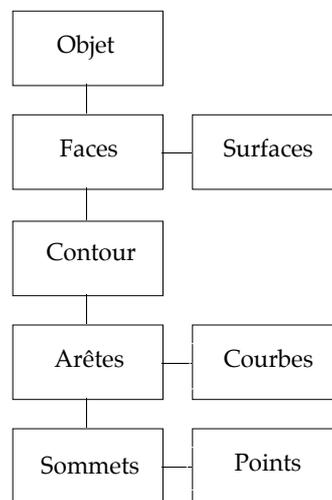


Figure I.6 : Modèle de représentation par les frontières, relations entre les entités de différentes dimensions.

Lorsqu'on utilise des surfaces planes (triangles, polygones, ...) pour modéliser un objet, une approximation polygonale est nécessaire ; on obtient un ensemble de faces représentées par leurs bords et leurs coins. L'inconvénient de cette méthode est sa faible puissance de représentation. Les objets présentant des surfaces non-planes ne peuvent pas être représentés exactement, leur approximation polygonale n'est pas unique et le nombre d'entités planaires créées est directement lié à la précision de l'interpolation souhaitée.

Pour décrire des surfaces non-planes ou sculptées, les NURBS découpées (IGES144) sont utilisées dans de nombreux systèmes de CAO, bien que leur manipulation soit plus compliquée. Par exemple, trouver les points d'intersection d'une courbe NURBS avec une surface NURBS ou l'intersection de deux surfaces NURBS revient à résoudre un problème mathématique complexe [Krishnan 94].

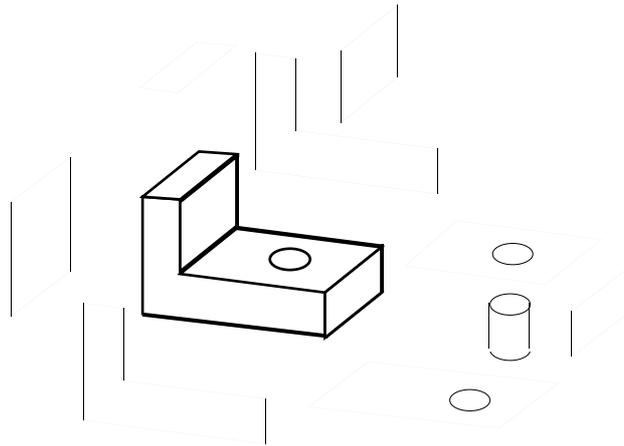


Figure I.7 : Exemple de représentation B-Rep (traits fins) de l'objet représenté en traits épais.

[Crocker 91] a étendu cette représentation surfacique aux représentations "non-manifolds", où le terme "manifold" signifie qu'une arête délimite deux faces. Le terme "non manifold" fait alors référence à des situations topologiques où une arête peut ne partager aucune face (wire edge), une seule face (lamina edge), ou plus que deux faces (non-manifold edge).

Avantage de la modélisation surfacique :

- + Facilités de description, de visualisation et de transformations géométriques

Inconvénients :

- Difficultés de réalisation des opérations logiques (dus au grand nombre de faces constituant l'objet initial),
- Difficultés de validation (les faces ne constituent pas toujours un volume),
- Absence d'algorithmes généraux (la plupart des algorithmes associés à ce modèle traitent des volumes dont les frontières sont composées de facettes planes, elles mêmes délimitées par des segments de droite pour faciliter les opérations logiques).

III.3. Modélisation volumique

Ce type de modélisation où l'objet est défini directement comme un volume, est devenue un standard de modélisation dans le domaine des modeleurs CAO (Unigraphics Version 10, ASCIS...).

III.3.1. Énumération de l'occupation spatiale

Dans ce modèle [Bhanu 87] [Givert 88] [Kholadi 91] [Peroche 90], un objet spatial est représenté par un ensemble de cellules occupées par le volume de l'objet dans l'espace. Celles-ci peuvent être des volumes quelconques mais l'usage des cubes de taille fixe est très fréquent. Ces cubes appelés "voxels" (volumes éléments) sont localisés le plus souvent par les coordonnées d'un point, par exemple le pied du cube, c'est à dire le sommet du cube le plus proche de l'origine du système d'axes (figure I.8).

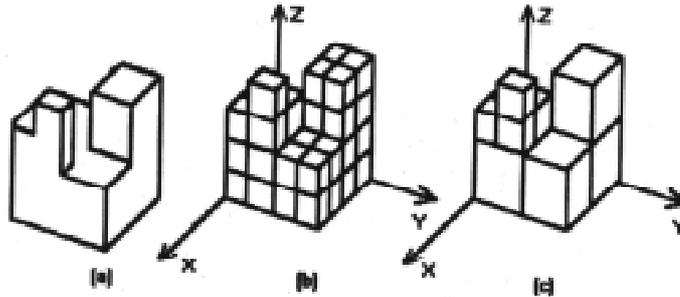


Figure I.8: Exemple de modélisation par énumération spatiale et arbre octal, (a) Objet spatial à modéliser, (b) Approche par énumération spatiale, (c) Approche par arbre octal (octrees).

Un objet est représenté par un tableau de triplets (x,y,z) appelé tableau spatial, habituellement ordonné et dont l'ordre correspond à l'ordre de la construction de l'objet.

Cette approche est simple à mettre en oeuvre. Elle facilite le calcul des propriétés physiques d'un objet et la mise en oeuvre des opérations booléennes (union, intersection et différence) et relationnelles (jointure, etc. ...). Elle reste invariante en translation, mais ne l'est pas en rotation surtout si les faces des cubes sont parallèles à l'un des axes du référentiel. Elle ne donne qu'une approximation plus ou moins grossière de l'objet spatial, selon le niveau de résolution choisi, dont dépend également l'encombrement mémoire.

La représentation par énumération spatiale regroupe les représentations par tableau à trois dimensions de voxels, appelé PEARY (Picture Element ARraY), ou par arbre octal (octrees).

Représentation PEARY

La représentation peary la plus simple est la représentation peary-binaire; elle consacre un bit à chaque pixel et ne permet seulement de différencier le plein du vide. Le peary permet de résoudre trivialement tout un ensemble de requêtes : Un point est-il à l'intérieur d'un solide ? quel est le volume d'un solide ? où est son centre de gravité ? Les opérations booléennes sont effectuées voxel par voxel.

Arbre octal

Dans une représentation peary, les voxels voisins ont très souvent la même valeur, ce qui permet d'envisager une forte compression des données. Un objet est décrit par un arbre octal qui est basé sur une subdivision successive et récursive d'un cube en huit cubes homogènes plus petits jusqu'à l'obtention d'un cube minimal homogène, qui est représenté par un octant noir (pour un voxel allumé si le cube est à l'intérieur de l'objet) et blanc (pour un voxel éteint si le cube est à l'extérieur du volume de l'objet).

Avantages :

- + Facilité de composition à l'aide d'opérateurs logiques de base,
- + Facilité de visualisation,
- + Facilité d'obtention à partir d'un autre modèle,
- + Facilité de calcul des propriétés géométriques et physiques d'un objet spatial.

Inconvénients :

- Représentation approximative des objets,
- Difficulté d'effectuer des opérations géométriques, qui nécessitent la reconstruction de l'arbre octal.

III.3.2. Représentation par volumes élémentaires

Le domaine de ce type [Peroche 90] [Kholadi 91] de représentation est limité à une famille d'objets, les objets étant indexés à l'intérieur d'une famille par un petit nombre de paramètres. Un objet est ainsi identifié par un n-uplet qui d'une part indique à quelle famille appartient l'objet et d'autre part donne la liste des paramètres permettant de le sélectionner. Par exemple ("cube",3) permet de désigner l'élément de taille trois unités dans la famille des cubes, l'élément ("prisme",5,2,8) désigne un objet de la famille prisme avec comme paramètres (nombre de cotés = 5, rayon du cylindre englobant = 2, hauteur du cylindre englobant = 8).

Par leur principe, ces schémas sont uniques, non-ambigus et triviaux à valider. L'absence de moyens de composition des familles d'objets pour former des objets plus complexes et l'inexistence d'algorithmes généraux de calcul des propriétés des objets représentés (car les familles d'objets doivent être traitées comme des cas particuliers) en sont les principaux inconvénients. Ce modèle ne se prête donc pas à la description et au traitement des objets complexes.

III.3.3. Géométrie Solide Constructive (CSG)

Dans ce modèle très utilisé en CAO, on dispose d'un ensemble de volumes élémentaires, tels que des parallélépipèdes, des sphères, des cônes, des tores, etc... On paramètre leurs dimensions, telles que largeur, longueur, hauteur, rayon, etc... On dispose de transformations géométriques (translations, rotations, etc.) et d'un ensemble d'opérateurs de composition (union, intersection et différence), qui permettent de combiner les objets de base afin de construire des objets spatiaux complexes.

La représentation de l'intérieur d'un objet généré selon ce modèle est un graphe acyclique binaire appelé souvent arbre de construction CSG, où chaque noeud non-terminal est un opérateur de composition et où les feuilles sont des objets élémentaires (figure I.9).

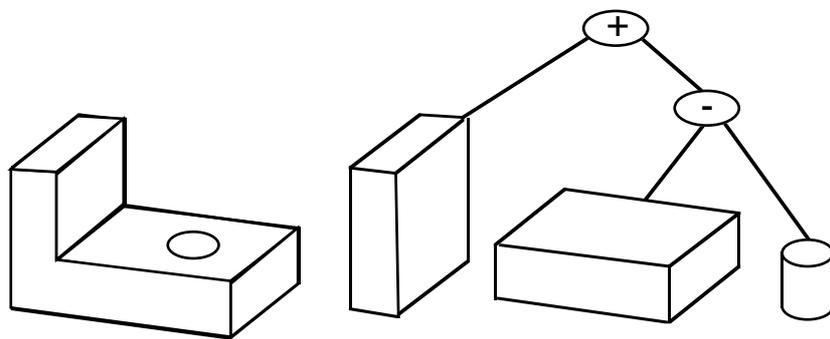


Figure I.9 : Exemple de construction d'un objet et de son arbre CSG associé.

Le résultat d'une opération booléenne doit être un objet régulier topologiquement clos, l'intersection de deux objets ne devant pas donner d'arêtes ou de faces pendantes.

Avantages :

- + Facilité de création,
- + Facilité de stockage,
- + Facilité de validation.

Inconvénients :

- Difficultés de calcul des propriétés géométriques et physiques d'un objet complexe,
- Difficultés pour effectuer certaines conversions (arbre CSG vers le modèle de représentation par les frontières par exemple).

Une extension de cette représentation a été appliquée au cas particulier de l'assemblage [Arbab 90]. En effet, La plupart des modeleurs basés sur la représentation CSG ne prennent

pas en compte les opérations booléennes sur les assemblages. En représentation CSG simple, chaque solide élémentaire est défini comme étant un demi-espace, topologiquement fermé et régulier. La représentation CSG étendue consiste à représenter les assemblages comme étant un ensemble de demi-espaces, les opérateurs booléens étant étendus pour manipuler ces demi-espaces. Elle utilise également un modèle théorique d'ensembles, basé sur les ensembles ouverts, pour représenter les assemblages et connecter ensemble les pièces semblables.

III.3.4. Représentation par balayage

Un balayage généralisé, un cylindre généralisé, ou un cône généralisé sont définis comme étant le volume décrit par le déplacement d'un ensemble de sections le long d'un axe suivant des règles de balayage. La particularité de la représentation par balayage [Bhanu 87] [Besl 88] [Chin 86b] [Flynn 91] [Kholadi 91] [Ponce 88] est d'utiliser des règles de génération au lieu de paramètres numériques. Un balayage généralisé est défini de la façon suivante : Soit une courbe spatiale appelée axe de la forme; en chaque point de cet axe, suivant un certain angle, il y a une section plane; sur chaque section plane, il y a une courbe plane qui constitue la section de l'objet dans ce plan. Il reste alors à définir une règle de transformation qui spécifie la façon dont évolue la section le long de l'axe.

Avantages :

- + Facilité de description (de l'objet en mouvement et de sa trajectoire),
- + Facilité de stockage,
- + Facilité de transformation géométrique.

Inconvénients :

- Manque d'outils algorithmiques pour le calcul des propriétés géométriques,
- Difficulté de mise en oeuvre des opérateurs booléens,
- Difficulté de visualisation.

IV. Modèles vision

Les modèles Vision sont des modèles qui exploitent directement les connaissances sur l'objet fournies par la vision 2D ou 3D et procurent des caractéristiques immédiatement compatibles avec l'application visée. L'une des phases importantes consiste à déterminer quelles caractéristiques il faut utiliser ou extraire. Pour l'analyse d'images industrielles, celles-

ci sont le plus souvent les frontières et les mesures géométriques dérivées de ces frontières; pour la reconnaissance de formes quelques caractéristiques simples peuvent être suffisantes.

Ces caractéristiques peuvent être classées en trois catégories :

Caractéristiques globales :

- Périmètre,
- Centre de gravité,
- Distance des points du contour au centre de gravité,
- Courbure,
- Aire,
- Moments d'inertie.

Caractéristiques locales :

- Segments droits,
- Arc avec courbure constante,
- Coins.

Caractéristique relationnelle :

- Distances et mesures d'orientation relatives entre les structures et les régions d'un objet.

IV.1. Points de surface, Normales, Courbures

Le plus simple des modèles Vision est obtenu directement par numérisation d'un objet avec un capteur 3D. On obtient un ensemble de points 3D permettant de décrire de façon universelle les frontières d'un objet, à condition d'avoir un nombre suffisant de points. Cette représentation n'apporte pas d'informations supplémentaires par rapport à un modèle CAO dégradé, mais présente l'avantage de fournir des informations directement exploitables par les opérateurs de vision de description élevée (algorithmes d'accroissement de région et de détection de contour).

En chaque point de surface peuvent être déterminées une normale ainsi que des courbures [Bhanu 87] [Boulangier 94b] décrivant les propriétés locales de la surface. Sur une surface

3D, chaque point a une valeur de courbure différente dans chaque direction. Parmi ces valeurs, on distingue les courbures principales, (le produit de ces courbures est appelé courbure de GAUSS); et les directions qui leur correspondent sont appelées directions principales. En chaque point retenu, on calcule les courbures minimale, maximale, moyenne et gaussienne. Les informations locales de points, de normales et de courbures constituent des caractéristiques directement exploitables par des algorithmes de vision.

IV.2. Modèle général de représentation multi-vues

Contrairement aux représentations "du point de vue de l'objet" qui contiennent une description tridimensionnelle explicite de l'objet étudié dans un repère quelconque, les représentations multi-vues [Chin 86b] [Flynn 91] [Gigus 90] consistent en un ensemble d'images de l'objet réalisant une description implicite de sa forme. Les graphes d'aspect s'appuyant sur la géométrie de l'image sont un exemple classique de ce type de représentation. Les nouvelles approches dans ce domaine incluent des informations de photométrie, de couleur, de texture, de positionnement de l'objet ou de variation d'illumination de la scène [Nayar 94].

Ces représentations, pour lesquelles la connaissance explicite de l'objet n'est pas nécessaire, peuvent être classées en deux grandes catégories en fonction du degré des caractéristiques extraites des informations-images :

- Certaines techniques stockent comme modèle de l'objet, un ensemble explicite d'images de l'objet. Elles utilisent cette représentation pour reconnaître les objets, soit en superposant directement l'image de l'objet sur les images modèles, soit en comparant l'image de l'objet avec les composantes principales de l'ensemble de données-modèle [Nayar 94].
- A l'opposé, d'autres techniques demandent plus de structure dans les données, en extrayant d'abord des caractéristiques des images et en les organisant en "aspects" ou en identifiant explicitement les invariants de ces caractéristiques. La reconnaissance de formes s'effectue alors en retrouvant le modèle de l'objet à partir des caractéristiques extraites de son image. Les approches diffèrent en fonction des critères utilisés pour grouper ces caractéristiques en ensembles invariants [Stockman 94] [Gros 94]. Construire de tels modèles s'effectue en deux phases : A partir des images 2D, isoler l'objet du fond de l'image puis extraire ses caractéristiques et les regrouper de manière significative (segmentation).

Ce type de modèle est utilisé en Reconnaissance de visages par Pentland et Sclaroff au MIT-Boston , en Robotique mobile [Huttenlocher 94], [Nayar 94],...

L'inconvénient de ces méthodes est la taille des bases de données à manipuler, chaque objet étant décrit par un grand nombre d'images 2D redondantes. Des améliorations sont donc recherchées pour stocker et organiser efficacement toutes ces informations, la description hiérarchique étant une des solutions à ce problème. Ces méthodes sont très liées au type de capteur utilisé et aux caractéristiques qu'il peut détecter.

IV.2.1. Sphère de vues

On construit un modèle multi-vues d'un objet en construisant une liste des caractéristiques de l'objet et des conditions sous lesquelles elles sont visibles. La caractéristique la plus simple utilisée est le segment droit représentant une portion de la surface de l'objet à laquelle, on peut faire correspondre une normale à la surface ou une discontinuité de réflectivité. L'ensemble des différentes positions de vue, est représenté en partitionnant la surface d'une sphère de vue unité en carreaux élémentaires de petite taille, plutôt uniformes. Pour chacun de ces points de vue, on stocke les caractéristiques qui nous serviront pour la reconnaissance, par exemple les coordonnées des extrémités des arêtes visibles.

Avantages :

- + Partition facile à calculer,
- + Partition fixe pour tous les objets.

La difficulté de cette méthode est de choisir une bonne taille de partition. Avec un partitionnement fin, les vues de régions proches sont similaires qualitativement, et ne fournissent aucune information supplémentaire pour mettre en correspondance les vues de l'image. D'un autre côté, avec une partition plus large, des vues importantes sont négligées.

IV.2.2. Graphes d'aspect

Dans cette approche, la sphère de vues est partitionnée en fonction de la structure qualitative des vues : l'aspect. celui-ci peut être défini de différentes façons, qui dépendent de l'application : Ensemble des faces visibles (figure I.10), ensemble des arêtes concourantes, ou autres mesures topologiques sur les vues. La partition est définie par une relation d'équivalence entre les points de vue, ou deux points de vue sont équivalents, si l'aspect des deux points de vue est le même.

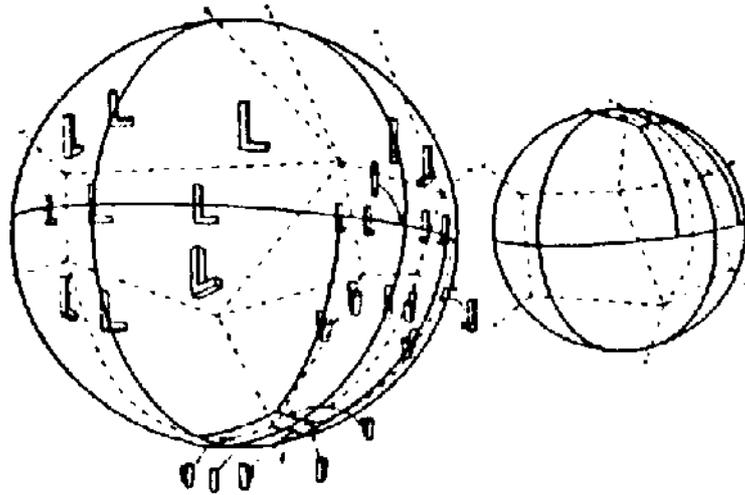


Figure I.10 : Graphe d'aspect d'un objet en forme de L (traits pointillés), les traits continus correspondent aux limites des partitions dans lesquelles figure une vue représentative.

IV.2.3. Travaux récents

Travaux de Nayar [Nayar 94]

Pour Nayar, les images 2D n'apportent pas seulement des informations permettant de reconstruire la forme d'un objet, mais également des informations permettant de construire une représentation d'apparence de l'objet. Dans cette représentation d'apparence, on trouve les variations de clarté dues à la forme 3D de l'objet, les propriétés de réflectance de la surface, les paramètres du capteur, et les conditions d'illumination. La représentation de forme et la représentation d'apparence s'avèrent être complémentaires, leur combinaison offre une plus grande puissance de représentation qu'une représentation de forme seule.

Travaux de Shapiro [Shapiro 94]

Un objet industriel est représenté par un petit ensemble de vues représentatives, où chaque vue a son ensemble de caractéristiques propres. Ces caractéristiques sont : arcs circulaires coaxiaux, ellipses, triplets de segments (formes en U ou en Z), jonctions (en V, en T, en Y ou en \rightarrow) ou lignes parallèles, proches ou éloignées. Elles présentent l'avantage d'offrir un meilleur pouvoir descriptif que les caractéristiques classiques que sont les points et segments de droite. Ces caractéristiques sont reliées entre elles par un graphe relationnel.

Travaux de Gros [Gros 94]

La méthode est basée sur l'utilisation de quasi-invariants, une catégorie de caractéristiques invariantes aux transformations géométriques infinitésimales. Ils sont utilisés

pour déterminer un déplacement de caractéristiques entre deux images par simple transformation géométrique (identité ou transformation affine). Deux images représentent alors le même point de vue, lorsque le plus grand nombre de caractéristiques des deux images sont mises en correspondance. Ces travaux visent à automatiser la phase de construction d'un modèle et de pouvoir utiliser celui-ci pour reconnaître un objet dans une image.

IV.3. Image Gaussienne Etendue

L'image gaussienne d'un objet représente la distribution des normales aux surfaces, un point sur la sphère gaussienne correspondant à la normale à la surface en ce point. Dans une IGE [Bhanu 87] [Okada 92] [Shiraï 87], les flèches montrent la direction de la normale à la surface et la longueur de chaque flèche représente l'aire des surfaces de même direction (un cylindre est approximé par un prisme à plusieurs faces) (figure I.11).

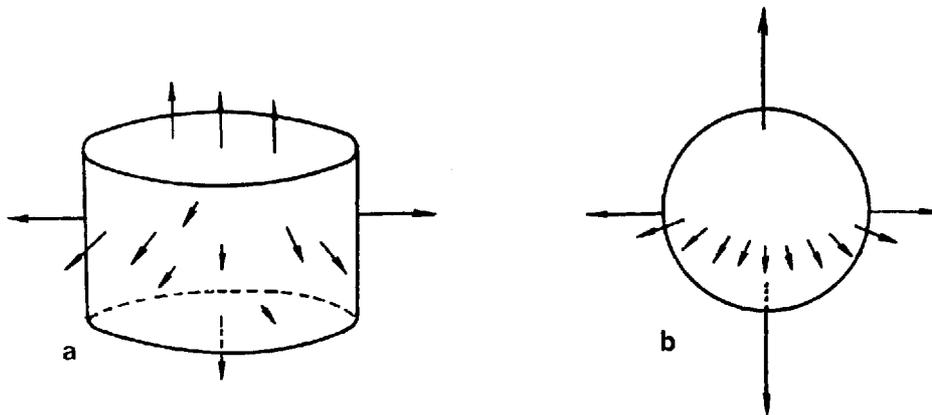


Figure I.11 : Exemples d'EGI : (a) Cylindre (b) EGI d'un Cylindre.

En pratique, on obtient une Image Gaussienne Etendue en divisant la surface d'un objet en carreaux d'aire unitaire et en calculant leurs normales. Lorsque les vecteurs normaux sont représentés par des vecteurs-unitaires, l'IGE peut être exprimée par un ensemble de vecteurs unitaires ou une distribution de points sur la sphère unitaire.

L'IGE est indépendante de la position des surfaces. Pour un polyèdre convexe, il y a une IGE unique représentant l'objet. Mais une IGE correspond à une infinité d'objets concaves. L'IGE est ainsi un modèle abstrait représentant une infinité d'objets.

Ikeuchi et Hebert [Ikeuchi 94] ont apporté des extensions aux IGE, qui ne pouvaient manipuler que des images d'objets convexes en rotation. Les Images Gaussiennes Etendues Distribuées (IGED) peuvent manipuler des objets non-convexes et également en translation. Finalement, les Images d'Attributs Sphériques (IAS) autorisent la manipulation d'objets non-convexes, avec des transformations générales.

IV.4. Polygonisation de Voronoï, triangulation de Delaunay

Cette modélisation volumique s'applique à des objets décrits par un ensemble de points 3D appartenant à leur enveloppe et consiste à placer ces points sur des polyèdres qui doivent satisfaire à certaines conditions de géométrie. Pour la modélisation de Delaunay [Bertin 92] [Besl 88] [Soucy 92] les polyèdres ont nécessairement 4 faces, alors que pour la modélisation de Voronoï [Besl 88] [Bertin 92], le nombre de faces peut varier de 5 à 24 offrant ainsi une plus grande flexibilité de modélisation (figure I.12).

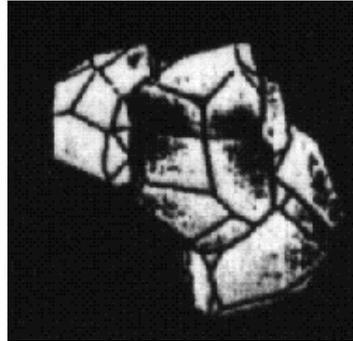


Figure I.12 : Polyèdre de Voronoï.

Les inconvénients de cette modélisation sont que sa qualité est dépendante de la quantité et de la densité de points disponibles, et qu'elle est moins puissante que la décomposition en primitives de par la simplicité de ses entités.

IV.5. Décomposition en primitives

Cette représentation consiste à partir de points 3D à décomposer la forme décrite en primitives surfaciques simples. Ces primitives surfaciques simples peuvent être des triangles construits d'abord à partir de points 3D puis transformés ("split and merge") pour optimiser le maillage [Hoppe 92] [Hoppe 93] [Soucy 92]. Les carreaux peuvent être aussi des polygones ou des surfaces quadriques. Des surfaces plus compliquées comme des surfaces NURBS [Boulanger 93a] [Boulanger 93b] peuvent également être extraites à partir des points et après segmentation pour générer un modèle surfacique complexe, proche de celui qui serait généré pour le même objet avec un logiciel de CAO (ingénierie inverse).

Pour Besl [Besl 94], la meilleure des représentations reste la représentation par triangles, en effet les représentations plus complexes comme celles utilisant les surfaces NURBS, bien que plus compactes, très puissantes d'un point de vue descriptif et plus exactes nécessitent des algorithmes et moyens de calculs importants. La représentation par triangles, dont l'erreur de représentation peut être paramétrée est beaucoup plus souple, de plus elle est bien implantée dans le milieu industriel.

V. Modèles déformables

Les modèles déformables sont utilisés dans de nombreux domaines. En imagerie médicale, ils permettent une modélisation efficace dans l'espace et dans le temps de certains organes comme le coeur (superquadriques, hyperquadriques). En animation, ils permettent de simuler le comportement complexe d'objets rigides ou non-rigides, en interaction avec leur environnement (modèles physiques). En vision 2D ou 3D, ils permettent de suivre dans le temps et l'espace le déplacement d'objets (snakes, maillages adaptatifs), ou de faire de la Reconnaissance de formes (geons),...

V.1. Surfaces Superquadriques, Hyperquadriques et GEONS

Ce type de modèle correspond à la définition d'une ou de plusieurs surfaces délimitant un volume. La plus simple de ces surfaces est la superquadrique, ou une seule de ces surfaces est suffisante pour modéliser un volume.

V.1.1. Superquadriques

Les superquadriques sont définies par l'équation implicite suivante [Cohen 94] [Yokoya 92] :

$$\left(\left(\frac{x}{a_1} \right)^{\frac{2}{\varepsilon_2}} + \left(\frac{y}{a_2} \right)^{\frac{2}{\varepsilon_2}} \right)^{\frac{\varepsilon_2}{\varepsilon_1}} + \left(\frac{z}{a_3} \right)^{\frac{2}{\varepsilon_1}} = 1$$

où a_1, a_2, a_3 sont des paramètres de taille et ε_1 et ε_2 sont des paramètres de forme.

Les superquadriques à 5 paramètres sont surtout utilisées dans le domaine médical, mais ont également été utilisées pour la construction de modèles à partir de données 3D [Ferrie 93]. Dans ce cas malgré l'utilisation d'un ensemble de superquadriques, le pouvoir descriptif reste faible.

La limitation du modèle superquadrique est due au faible ensemble de formes permises, car le modèle est contraint par quelques paramètres globaux (minimum 5) et le nombre de degrés de liberté reste faible.

V.1.2. Superquadriques déformées

Afin d'augmenter la flexibilité du modèle superquadrique à 5 paramètres, Solina et Bajcsy [Solina 90] utilisent des paramètres de déformation globaux. Ces paramètres de déformation globaux ont été introduits par Barr, en 1984 [Barr 84].

Compression (2 paramètres)

La déformation suivant l'axe z est définie par :

$$\begin{cases} X = f_x(z).x \\ Y = f_y(z).y \\ Z = z \end{cases}$$

où X, Y, Z sont les composants de la surface déformée, f_x et f_y sont les fonctions de compression suivant x et y de l'objet dans son système de coordonnées, et où x, y, z sont les composants du vecteur d'origine.

Pliage (2 paramètres)

Le plan de pliage est défini par l'axe z et le vecteur r dans le plan $x-y$, dont la direction est définie par l'angle α . La déformation de pliage est d'abord définie par la projection des composantes x et y de tous les points sur le plan de pliage, on effectue ensuite la déformation de pliage, puis les points sont reprojétés sur leur plan d'origine.

La projection du point (x,y) sur le plan de pliage est :

$$r = \cos(\alpha - \beta)\sqrt{x^2 + y^2} \quad \text{et} \quad \beta = \arctan \frac{y}{x}$$

Le pliage transforme r en $R = \frac{1}{k} - \cos(\gamma)\left(\frac{1}{k} - r\right)$ où γ est l'angle de pliage, calculé à partir du paramètre de courbure k : $\gamma = \frac{z}{k}$ (figure I.13).

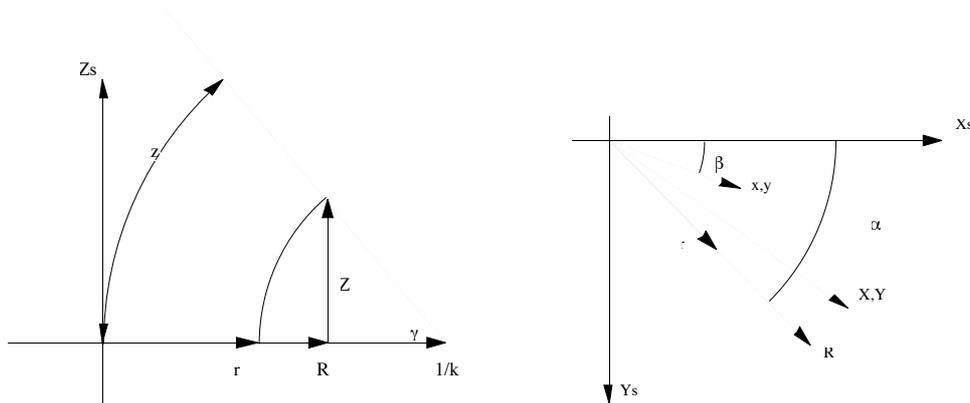


Figure I.13 : Définition des paramètres de pliage, à gauche dans le plan z,r , à droite dans le plan x,y .

Le vecteur des nouvelles coordonnées est donc :

$$\begin{cases} X = x + \cos(\alpha)(R - r) \\ Y = y + \sin(\alpha)(R - r) \\ Z = \sin(\gamma)\left(\frac{1}{k} - r\right) \end{cases}$$

Exemple de modélisation par superquadrique dans le domaine médical (figure I.14) :

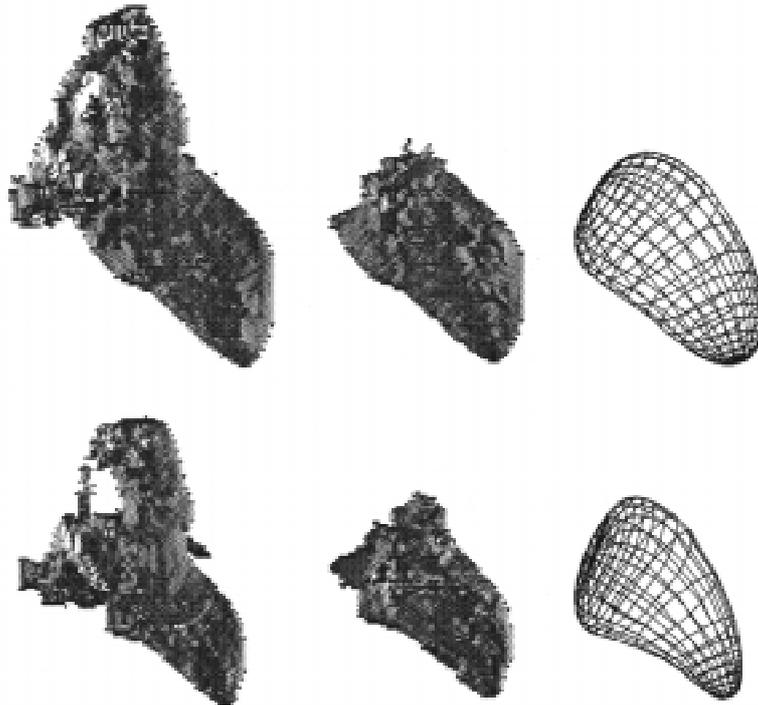


Figure I.14 : Ajustement d'une superquadrique 3D sur un ventricule gauche de chien, images binaires (à gauche et au milieu), superquadrique estimée (à droite) [Robert 95].

V.1.3. Hyperquadriques et hyperquadriques hybrides

Hyperquadriques

Les hyperquadriques [Cohen 94] sont définies comme étant l'ensemble de points satisfaisant :

$$H(x, y, z) = \sum_{i=1}^N |H_i(x, y, z)|^{\gamma_i}, \text{ ou } H_i \text{ est une forme affine 3D}$$

$$H_i(x, y, z) = (a_i x + b_i y + c_i z + d_i), \text{ ou } a_i, b_i, c_i, d_i \text{ et } \gamma_i \text{ sont des constantes } (\gamma_i > 0)$$

Le modèle hyperquadrique se distingue du modèle superquadrique par la non-symétrie de sa représentation et son grand pouvoir descriptif. Par contre, la définition de formes non-convexes est difficile; de plus on ne peut pas suffisamment contrôler la forme et l'emplacement d'éventuelles concavités modifiant la forme globale de la surface.

Hyperquadriques hybrides

Afin de définir des propriétés plus locales, les hyperquadriques hybrides ont été introduites [Cohen 94]. Cette méthode de modélisation permet de définir des propriétés locales implicites sur un modèle de forme globale, en intégrant au modèle hyperquadrique un nombre arbitraire de concavités aux endroits désirés.

$$H(x, y, z) = \sum_{i=1}^N |H_i(x, y, z)|^{\gamma_i} + \sum_{j=1}^M c_j e^{-\sum_{l=0}^{L_j} |K_{jl}(x, y, z)|^{\gamma_{jl}}} = 1$$

où les H_i et les K_{jl} sont les formes linéaires 3D, où N représente le nombre de bandes définissant l'hyperquadrique, où M est le nombre de concavités utilisées, et où les L_j sont les nombres de bandes définissant chaque concavité.

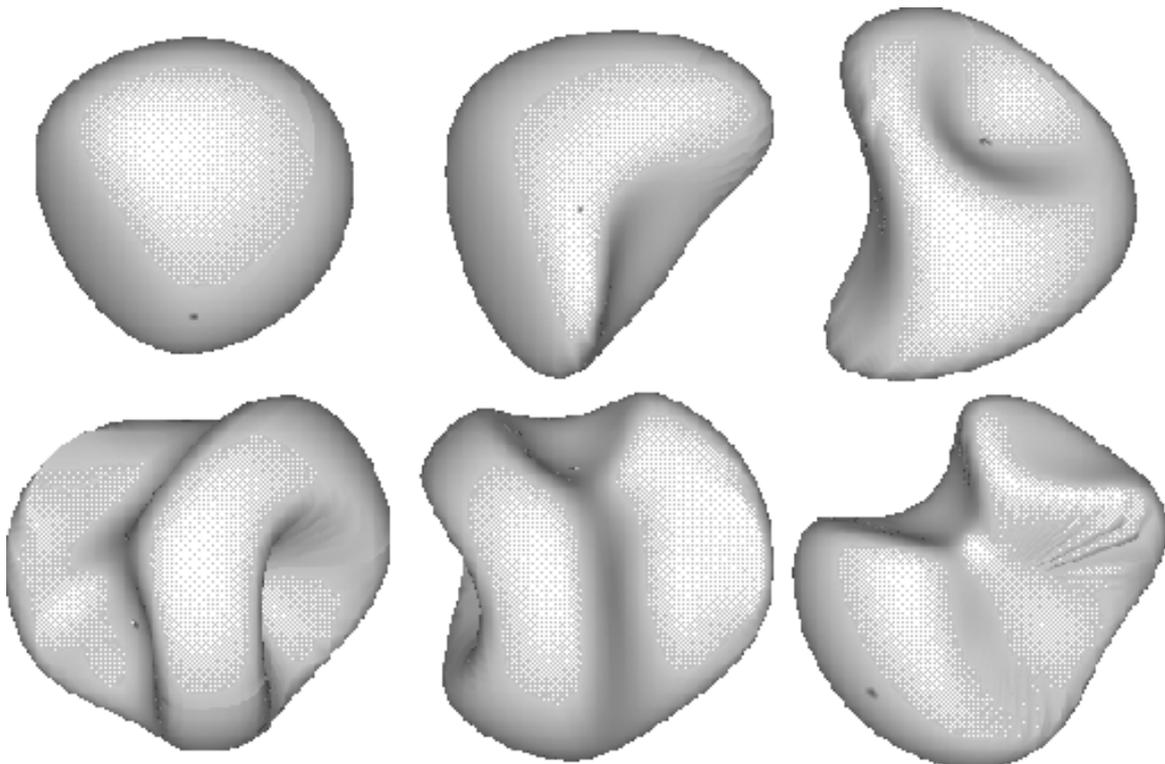


Figure I.15 : En haut hyperquadriques hybrides initiales auxquelles sont ajoutées deux concavités (en bas).

Comme on peut le voir figure I.15, on peut générer un certain nombre de concavités, la définition de ces concavités restant locale et ne modifiant pas la forme globale de l'hyperquadrique. L'avantage de cette approche est que la description de la forme dépend

d'une équation implicite définie par un petit nombre de paramètres; son inconvénient est que les concavités doivent toujours être générées par paire.

A l'heure actuelle, l'utilisation de surfaces à base de superquadriques reste limitée au domaine médical (détermination du volume cardiaque, modélisation des déformations du coeur , ...), pour lequel leur pouvoir descriptif est particulièrement bien adapté.

V.1.4. GEONS

Contrairement à la modélisation utilisant des superquadriques où hyperquadriques, où une seule surface complexe est suffisante à la modélisation d'un objet, on peut être amené à modéliser un objet en décomposant sa forme en primitives simples.

Geons de Biederman

Une idée de représentation qualitative a été proposée par Biederman [Okada 92] [Flynn 91], qui a développé un catalogue de 36 geons (GEometric iONS), où chaque entité du catalogue est une combinaison unique de quatre caractéristiques qualitatives :

- La forme d'une section est droite ou courbe,
- Une section est symétrique ou asymétrique,
- La section est constante, croissante , croissante puis décroissante lorsqu'elle se déplace le long d'un axe,
- L'axe est droit ou courbé.

Geons paramétriques

Les geons paramétriques [Wu 94] sont au nombre de sept, définis par des équations paramétriques qui contrôlent leur taille et leur degré de compression et de pliage; ils sont un sous-ensemble des geons de Biederman. La principale différence des geons paramétriques par rapport aux geons traditionnels est que les précédents sont caractérisés par leurs contraintes de forme globale, et restreints à la description d'un type de formes spécifiques, ce qui facilite l'approximation de forme.

Les sept formes de geons sont : Ellipsoïde, Cylindre, Cuboïde, Cylindre comprimé, Cuboïde comprimé, Cylindre courbé et Cuboïde courbé (figure I.16) :

Ces formes sont définies à partir de l'équation classique de la superquadrique à 5 paramètres définie ci-dessus.

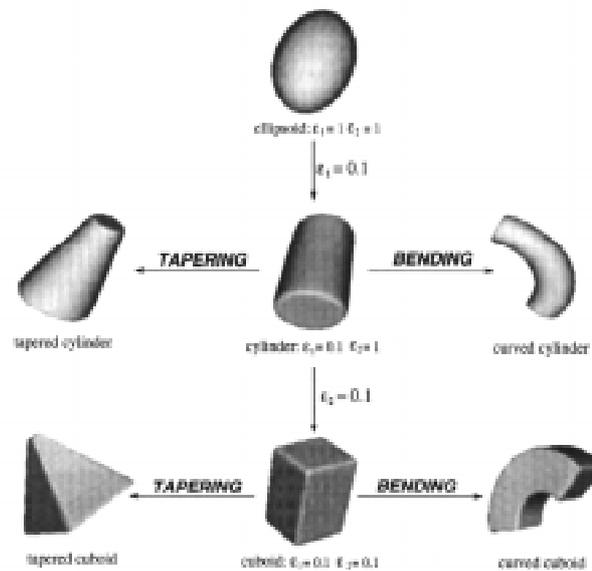


Figure I.16 : Les 7 geons paramétriques.

Dans le cas des geons de Biederman ou des geons paramétriques, chaque objet 3D est composé d'un certain nombre de geons connectés entre eux.

V.2. Approche physique de modélisation

Les objets ne sont plus modélisés selon leur forme uniquement, mais ils sont considérés comme des systèmes mécaniques soumis à des forces, respectant les propriétés du matériau utilisé. Cette méthode de modélisation fonctionne très bien pour des objets dont la forme (qui peut être très complexe) évolue en fonction du temps et en fonction des interactions avec leur environnement.

Ce type de modèle est en général représenté par une structure discrète (maillage) dont le comportement est régi par des équations différentielles complexes. Le grand nombre de paramètres et la complexité des relations entre ces paramètres font que ce type de modèle ne peut être utilisé pour la comparaison. Cette modélisation est surtout utilisée dans le domaine de l'imagerie médicale, pour la modélisation d'objets naturels, en imagerie de synthèse, en réalité virtuelle ...

Une technique de calcul telle que la méthode des éléments finis, développée pour résoudre des problèmes variationnels en modélisation physique, peut alors être appliquée à la vision par ordinateur.

V.2.1. Maillage adaptatif

Les maillages adaptatifs sont souvent utilisés en reconstruction de surfaces 3D ou pour l'étude du mouvement d'objets rigides ou non-rigides évoluant dans le temps.

Maillage triangulé

A partir d'un ensemble de points 3D, on peut construire un ensemble de triangles dont les sommets sont les points 3D, puis réduire le nombre de triangles afin d'alléger le modèle [Hoppe 93][Hoppe 94]. Pour garder une signification physique au modèle, les noeuds du maillage peuvent être reliés entre eux et aux points 3D par des ressorts imaginaires [Huang 93] [Terzopoulos 91].

Maillage simplex

Ce type de maillage [Delingette 94] (figure I.17) est le dual de la triangulation. Le comportement élastique de la surface est modélisé par des fonctionnelles de stabilisation locales, qui contrôlent la courbure moyenne de la surface, en chaque point du maillage. Les maillages simplex sont des structures très adaptatives, dont la résolution peut localement augmenter pour mieux s'adapter.



Figure I.17 : Déformation d'un maillage simplex pour représenter une main.

Maillage Surfaces B-Spline :

Au lieu de choisir comme points de masse des points 3D appartenant aux surfaces de l'objet, il est possible de décomposer les surfaces de l'objet en surfaces B-Spline dont les points de contrôle sont les points de masse reliés entre eux [Thingvold 90].

Maillage B-Spline bicubiques [Clarysse 95]

Le problème général de l'interpolation de surfaces à partir de coupes successives d'un objet est très important en imagerie médicale. La reconstruction 3D permet de mieux appréhender la forme d'un organe (souvent le coeur) ou d'une structure biologique ainsi que son positionnement dans l'environnement. L'utilisation de surfaces splines permet d'assurer un aspect lisse aux surfaces. Trouver la surface B-Spline bicubique idéale consiste à déterminer la surface qui minimise une fonctionnelle J . Cette fonctionnelle J est composée d'un terme de pénalisation qui traduit l'écart surface/données ainsi que d'un terme de stabilisation qui contrôle le degré de lissage de la solution, tout en réduisant l'espace des solutions possibles.

V.2.2. Modèles de Terzopoulos/ Metaxas

La modélisation physique a incorporé des principes physiques dans les modèles géométriques conventionnels. Par exemple, un seul modèle d'objet contient à la fois sa géométrie (forme de l'objet) ainsi que ses déplacements (dynamique). Ces principes physiques contrôlent la création et le comportement des modèles aux forces, couples, énergies de contraintes internes, et autres quantités physiques. Ces modèles sont capables d'avoir un comportement simplement élastique, ou plus généralement non-élastique, incluant la visco-élasticité, la plasticité et les fractures [Terzopoulos 88]. Mathématiquement, cette approche demande l'utilisation d'équations différentielles (ordinaires ou partielles) dynamiques, qu'il faut résoudre numériquement, pour régir le comportement du modèle en fonction de ses interactions avec l'environnement.

En utilisant la mécanique Lagrangienne, la géométrie de surfaces sculptées ou conventionnelles est décomposée en primitives dynamiques, directement utilisables en vision. Ces primitives peuvent également être connectées entre elles par des contraintes physiques, pour former des modèles composés.

V.2.3. Représentation modale (Sclaroff)

Toute forme peut être représentée comme étant la déformation d'une forme prototype. Les modes sont les vecteurs propres de la matrice de raideur de la forme prototype, correspondant aux axes de symétrie généralisés de l'objet et permettent d'avoir une représentation de forme fréquentielle et robuste (figure I.18). Des méthodes robustes de modélisation de formes 3D, de reconnaissance de formes, de suivi d'objets 3D, ont été mises au point en utilisant cette représentation.

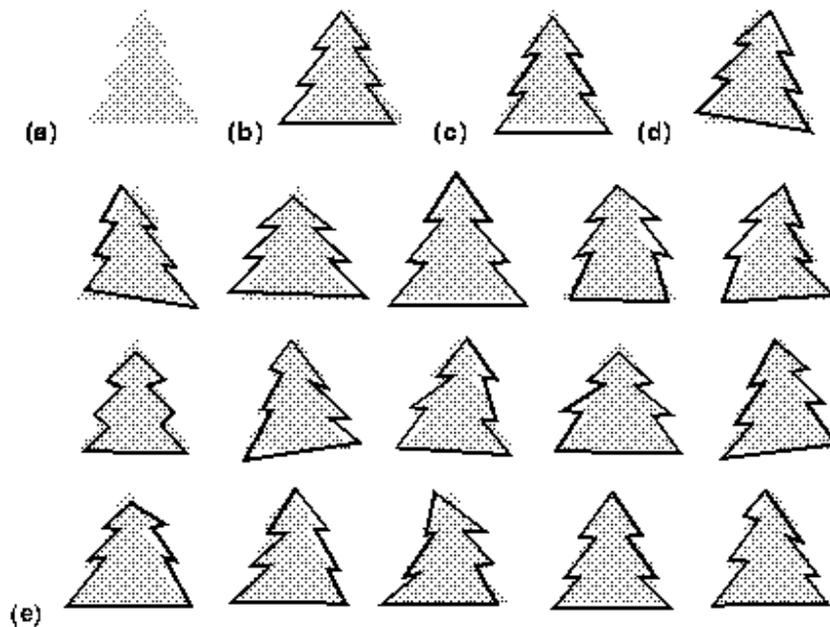


Figure I.18 : Les 18 modes les plus bas calculés pour la forme (a), modes de translation (b) et (c), mode de rotation (d), les autres modes sont non-rigides.

Cette représentation [Pentland 91] [Sclaroff 95] n'est pas adaptée à la mise en correspondance entre des données et des modèles, car chaque objet est décrit comme des déformations à partir d'une forme prototype et donc connue. La seule mise en correspondance possible est entre deux ensembles de caractéristiques d'objets ayant la même forme prototype.

V.2.4. Snakes

Afin de pouvoir modéliser des objets en mouvement à partir des contours et de les poursuivre, on utilise le concept récent de contours actifs déformables ou "snakes" [Bascle 92] [Fujimura 92] [Selsis 93].

Un contour actif est un ensemble de points ordonnés définissant une courbe. L'évolution de la courbe est régie par la minimisation constante de son énergie associée E_{snake} . Cette énergie est définie de sorte à présenter un minimum local sur le contour extérieur d'objets d'une image. Ainsi un contour actif placé à proximité d'un contour, évoluera jusqu'à épouser sa forme.

L'énergie E_{snake} est composée de deux termes : $E_{snake} = E_{int} + E_{ext}$

L'énergie interne E_{int} régit les interactions entre les points du snake, alors que l'énergie externe ou énergie image E_{ext} régit l'interaction entre les points de l'image et les points du snake.

A l'instant t , $I(x,y,t)$ représente l'intensité de l'image au point (x,y) et $v(s,t)=(x(s,t),y(s,t))$ représente la position sur le modèle du contour, où s est le paramètre qui correspond à la longueur de l'arc. Les paramètres w_i sont des poids.

Energie interne

$$E_{\text{int}} = E_{\text{spline}} + E_{\text{aire}} + E_{\text{dist}} + E_{\text{continuité}}$$

$$E_{\text{spline}} = \frac{\sqrt{w_1 \|v_s(s,t)\|^2 + w_2 \|v_{ss}(s,t)\|^2}}{2}$$

$$E_{\text{aire}} = \frac{w_s}{2} \{x(s,t)y_s(s,t) - x_s(s,t)y(s,t)\}$$

$$E_{\text{dist}} = w_4 (\bar{d} - \|v_s\|)^2$$

$$E_{\text{continuité}} = w_5 (x_{ts}^2 + y_{ts}^2)^2$$

où les indices représentent les dérivées partielles, et où d représente la distance moyenne entre les points du modèle adjacent.

E_{aire} régit la surface délimitée par la courbe.

Les autres termes d'énergie sont analogues aux forces intrinsèques de torsion et de tension qui régissent le comportement d'un matériau. Le terme de tension régit la distance entre les points du snake tandis que le terme de torsion régit les courbures du snake.

Energie externe

$$E_{\text{ext}} = E_{\text{arête}} + E_{\text{term}} + E_{\text{intens}}$$

$$E_{\text{arête}} = -w_6 \|\nabla I(x(s,t), y(s,t), t)\|^2$$

$$E_{\text{term}} = -w_7 |I_{yy}I_x^2 - 2I_{xy}I_xI_y + I_{xx}I_y^2|$$

$$E_{\text{intens}} = w_8 \left\| \frac{dI}{dt} \right\|^2$$

L'énergie externe est composée de termes qui attirent les points du snake vers les extrémums locaux de l'image et vers les extrémums locaux d'une fonction de courbure de l'image.

La limitation majeure de l'utilisation des contours actifs reste le problème posé par leur initialisation. En effet, le snake doit être initialisé à proximité de la forme que l'on désire modéliser, si l'on veut obtenir une convergence rapide de l'algorithme.

En deux dimensions, le modèle utilisé est une grille imaginaire élastique [Sebahi 95]. Cette grille élastique est pliée, tordue, compressée dans toutes les formes 3D désirées. La reconstruction de forme est guidée par un ensemble de ressorts imaginaires qui forcent la forme en position, orientation et/ou en courbure. Néanmoins les surfaces snakes sont très peu utilisées, car le problème d'initialisation du snake déjà présent en une dimension devient difficile à résoudre en deux dimensions.

V.2.5. Déformations libres (*Free Form Deformations - FFD*)

Ce type de déformations est utilisée en animation graphique. Elles consistent à enfermer l'objet en représentation surfacique ou volumique, dans un parallélépipède 3D subdivisé en parallélépipèdes élémentaires, puis à déformer l'ensemble de barres 3D ainsi créé, et à constater les déformations impliquées sur l'objet [Sedberg 86] (figure I.19). Un ensemble de barres plus complexes ont aussi été utilisées, barres cylindriques [Coquillart 90] ou barres en courbes NURBS [Lamousin 94].

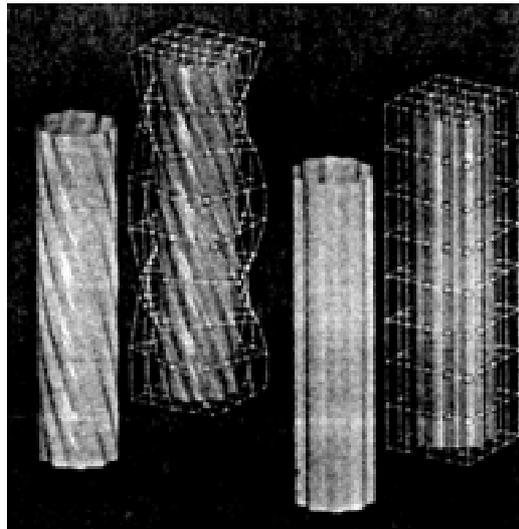


Figure I.19 : Déformation d'un cylindre utilisant les FFD.

Conclusion

L'objectif de notre travail est de mettre en correspondance un ensemble de points 3D décrivant les surfaces d'un objet, avec le modèle géométrique de celui-ci obtenu en CAO. Afin de simplifier la méthode de mise en correspondance, il est nécessaire d'avoir des entités similaires aussi bien dans l'image que dans le modèle. Nous avons présenté dans ce chapitre différents types de modèles (modèles géométriques, modèles vision, et modèles

déformables), leurs applications, ainsi que leurs avantages et leurs inconvénients. L'ensemble de points 3D représentant des surfaces, nous avons choisi d'utiliser des modèles CAO décrivant également ce type d'entités. Pour cela, nous avons opté pour l'utilisation de deux types de surfaces, l'objectif de ce travail n'étant pas de créer notre propre modèle, nous avons utilisé deux formats couramment utilisés dans le milieu industriel :

- les NURBS découpées : celles-ci sont très utilisées par les modelleurs CAO pour leur grand pouvoir descriptif, et choisies comme standard de modélisation entre logiciels de CAO (format IGES). Celles-ci bien que complexes permettent d'avoir une description exacte des objets manipulés.
- Triangulation STL : Les surfaces approximées par des triangles sont bien plus simples à manipuler d'un point de vue calculatoire, mais l'approximation introduit des erreurs de modélisation. Disposant d'un logiciel de conversion IGES/STL, nous avons utilisé la triangulation STL dédiée à la stéréolithographie.

Ces deux types de modèles surfaciques sont présentés dans les détails au chapitre III. Nous utiliserons alors soit un seul de ces modèles, soit les deux pour une représentation géométrique optimisée.

Chapitre II

Mise en Correspondance

I. Introduction

Parmi les traitements classiques de la vision par ordinateur, on peut distinguer deux types de mises en correspondance. Le premier consiste à fusionner les données provenant de différents types de capteurs, prises à différents moments ou à partir de plusieurs points de vue, afin de construire un ensemble d'entités décrivant au mieux l'objet ou la scène observés. L'autre type de mise en correspondance suppose connu le modèle de la scène observée et permet de superposer ce modèle avec les données fournies par les capteurs. Les applications sont diverses et nombreuses : Reconnaissance et localisation d'objets 3D, suivi d'objets 2D ou 3D, navigation, mesure dimensionnelle, étude de déformations, ... Pour notre part, nous nous intéresserons à cette deuxième définition de la mise en correspondance.

D'une manière générale, on dispose de deux formes : une forme-modèle M connue ; une forme-données D résultant des informations fournies par des capteurs 2D ou 3D. Ces deux formes étant décrites dans des systèmes de coordonnées différents (système de coordonnées du capteur et système de coordonnées de modélisation), il est nécessaire, afin de pouvoir les exploiter, de les exprimer dans un système de coordonnées commun. La phase de mise en correspondance consiste donc, après choix d'un mode de représentation, à trouver la transformation rigide ou non-rigide à appliquer sur D afin de minimiser un critère de distance entre D et M .

Les méthodes de mise en correspondance peuvent être classées suivant les dimensions des données et du modèle :

- Données 2D/Modèle 2D : Simples à manipuler,
- Données 2D/Modèle 3D : Très utilisé grâce à la combinaison du pouvoir descriptif des modèles 3D, avec la stratégie rapide et peu coûteuse des capteurs 2D,
- Données 3D/Modèle 2D : Perte d'informations au niveau des données,
- Données 3D/Modèle 3D : Allie la puissance de description des modèles 3D, avec celle des informations 3D.

Nous présenterons, dans un premier temps une revue des méthodes de mise en correspondance 2D. Nous détaillerons ensuite les méthodes de mise en correspondance 3D, puis présenterons la méthode de mise en correspondance originale que nous avons développée.

II. Mise en correspondance 2D

Une revue de toutes les méthodes de mise en correspondance 2D peut être trouvée dans l'article de synthèse de L.G. Brown [Brown 92] ; cette synthèse a servi de base à l'établissement de ce paragraphe.

II.1. Mise en correspondance multi-modale

Problème : Mise en correspondance d'images de la même scène, mais acquises avec différents capteurs.

Type d'applications : Intégration d'informations permettant de réaliser une segmentation.

Caractéristiques d'une telle méthode :

- Utilise souvent les modèles des capteurs,
- Nécessité de réaliser un première mise en correspondance des intensités,
- L'acquisition d'images contenant des marqueurs peut simplifier le problème de mise en correspondance.

II.2. Mise en correspondance de primitives

Problème : Trouver la correspondance d'un motif dans une image.

Type d'applications : Localiser et reconnaître un motif tel qu'un atlas, une carte, un modèle d'objet dans une image.

Caractéristiques d'une telle méthode :

- Approche basée sur les modèles,
- Caractéristiques présélectionnées,
- Propriétés connues des objets,
- Correspondance de haut niveau.

II.3. Mise en correspondance d'images prises de différents points de vue

Problème : Mise en correspondance d'images prises de différents points de vue.

Type d'applications : Reconstruction de profondeur ou de formes.

Caractéristiques d'une telle méthode :

- Transformations locales pour la prise en compte des distorsions de perspective,
- utilise les connaissances sur la géométrie du capteur et sur les propriétés des surfaces,
- Problème des occlusions.

II.4. Mise en correspondance temporelle

Problème : Mise en correspondance d'images de la même scène, prises à différents instants ou dans des conditions différentes.

Type d'applications : Détection de changement de formes de l'objet ou de déplacements.

Caractéristiques d'une telle méthode :

- La mise en correspondance doit tolérer des distorsions dues aux changements,
- Les résultats sont meilleurs, s'il existe déjà un modèle de capteur ainsi qu'un modèle de changement de point de vue,
- Utilisation des méthodes de Fourier pour modéliser les sensibilités aux dissemblances.

II.5. Types de différences entre images

Le choix d'une méthode de mise en correspondance dépend du type de différences constatées entre les images :

- Différences dues à l'acquisition, qui peuvent être corrigées :

Ces distorsions (changement de point de vue ou bruit du capteur) peuvent être modélisées, ce qui permet de déterminer le type de transformation à rechercher.

- Différences dues à l'acquisition, mais qui ne peuvent pas être corrigées :

Ces distorsions sont difficiles à modéliser, car dépendantes de la scène (éclairage, conditions atmosphériques, ombrage, effet de perspective, certains types de bruit de capteur, ...).

- Différences dans les images qui représentent leur intérêt :

Mouvements, grossissement ou déformation d'objets, utilisation de capteurs de sensibilités différentes, ...

Les deux derniers types de différence rendent cette opération plus difficile du fait que la mise en correspondance exacte n'est plus possible.

III. Mise en correspondance 3D

Les nouveaux développements, dans le domaine de l'acquisition d'images de profondeur, permettent d'envisager de façon réaliste l'utilisation de séquences de données 3D pour l'estimation de transformations rigides. L'avantage par rapport à la mise en correspondance 2D est de disposer directement d'informations 3D, évitant ainsi la phase de reconstruction 3D source d'erreurs.

D'une manière générale, le problème de mise en correspondance 3D s'exprime ainsi : Après avoir choisi une représentation de données, de modèles et de transformation, à partir d'une sélection de primitives dans D et M , trouver l'ensemble des paramètres de la transformation qui minimise un critère de distance entre l'ensemble de données transformées et le modèle. D'un point de vue mathématique, le problème étant fortement non-linéaire, il s'agit de trouver les paramètres de la transformation qui minimisent le critère de distance globalement, ce qui nécessite un choix convenable de transformations initiales.

Pour définir complètement un problème de mise en correspondance, il est donc nécessaire de déterminer :

- la représentation de la transformation rigide recherchée,
- les primitives de D et M à utiliser pour la mise en correspondance,
- le calcul de distance associé aux primitives sélectionnées pour D et M ,
- la méthode d'optimisation,
- le critère à minimiser,
- le choix de la transformation initiale.

III.1. Représentation de la transformation rigide

Une transformation rigide peut être décomposée en une composante matrice de rotation R et une composante vecteur de translation T . Le point p se transforme alors en un point p' et la relation qui relie p' à p est :

$$p' = R.p + T$$

La matrice R doit être orthonormale et son déterminant doit être égal à un pour que celle-ci représente une rotation. Ces contraintes imposent que la matrice R n'ait que trois paramètres indépendants, mais provoquent des singularités dans la représentation. Ainsi une transformation rigide est décrite par six paramètres indépendants, trois pour la rotation, trois pour la translation. Ces représentations sont très utilisées pour leur simplicité, et les calculs sont alors prévus pour éviter les points singuliers.

On peut trouver en annexe A.I les représentations des transformations rigides les plus utilisées dans le domaine de la vision 3D :

- Matrice de rotation et Vecteur translation,
- Axe et angle de rotation,
- Angles d'Euler,
- Quaternions,
- Décomposition en paramètres de vissage.

Les représentations qui font intervenir des axes et angles de rotation constituent les représentations les plus intuitives d'une transformation rigide. Par contre, elles sont souvent plus complexes à manipuler lorsqu'il s'agit de composer des transformations et présentent des indéterminations.

La composition de transformations représentées par des quaternions ou des matrices est beaucoup plus simple à calculer, c'est pourquoi nous avons choisi ces deux représentations pour nos calculs.

III.2. Choix des caractéristiques à mettre en correspondance

Les primitives locales (points, lignes) sont sensibles au bruit et aux erreurs de quantification, ce qui peut apporter des incertitudes dans l'estimation de la transformation contrairement aux caractéristiques globales (surfaces) plus concises et plus robustes, mais plus difficiles à extraire (dans un milieu naturel en particulier). Du choix des primitives dépend donc la complexité et l'exactitude de la méthode de mise en correspondance.

Données D

Le type de données fournies directement par des capteurs 3D est souvent un ensemble de points 3D, qui a l'avantage de constituer une description exacte bien que discrétisée de la forme numérisée.

Les données, fournies ou reconstruites à partir de capteurs 2D, peuvent être des points 3D, des contours 3D et peuvent toujours être interpolées par des courbes ou des surfaces de plus haut niveau :

- Points 3D : Points 3D fournis par un capteur télémétrique, coins ou centres de symétrie obtenus à partir de données 2D, sous ensemble de points 3D (Points spéciaux dont la courbure est localement maximum et plus élevée qu'une valeur-seuil),

- Courbes 3D : Contours extraits d'images 2D [Lavallée 95] [Zhang 94], frontières internes ou axes de symétrie, courbes de niveaux [Kamgar Parsi 91],
- Surfaces 3D : Segmentation des points 3D, interpolation par des surfaces plus ou moins complexes (carreaux plans, triangles, quadriques, B-Splines NURBS,...)..

Modèle M

Soit, le modèle est construit à partir de données provenant de capteurs 2D ou 3D, auquel cas il est de même type que les données D. Soit, la structure géométrique de l'objet est supposée connue, on dispose alors d'un modèle complet et exact (modèle CAO, par exemple) pouvant être éventuellement dégradé pour le traitement.

Une méthode générale de mise en correspondance de formes 3D a été proposée par P. Besl en 1992 [Besl 92] ; dans ce cas, ces formes peuvent être de différents types :

- Ensemble de points,
- Ensemble de segments,
- Ensemble de courbes implicites $\bar{g}(x, y, z) = 0$,
- Ensemble de courbes paramétriques $(x(u), y(u), z(u))$,
- Ensemble de triangles,
- Ensemble de surfaces implicites $\bar{g}(x, y, z) = 0$,
- Ensemble de surfaces paramétriques $(x(u, v), y(u, v), z(u, v))$.

Cette méthode se limite au fait que l'ensemble de données D doit être un sous-ensemble de M et ne peut donc pas traiter les cas de recouvrement partiel de D et M.

Aux primitives géométriques données, peuvent s'ajouter d'autres informations favorisant la convergence de la mise en correspondance :

- Utilisation de marqueurs dans le domaine médical [Collignon 94],
- Utilisation d'informations d'intensité ou de courbure [Godin 94] [Godin 95],
- Contraintes de normales [Chen 92].

III.3. Calcul de distance inter-primitives

Dans ce type de problème d'optimisation non-linéaire, le critère à minimiser est un critère de distance global entre M et D , pouvant éventuellement être localement pondéré.

A la base, il est nécessaire de définir un calcul de distance local entre une primitive de D et une primitive de M , afin d'établir le calcul de distance global entre D et M . Dans le cas où D est un sous-ensemble de primitives de M [Besl 92], ou bien lorsque les primitives de D et celles de M sont déjà appairées, tous les calculs de distances locales à partir des primitives de D sont pris en compte. Dans le cas contraire, il faut prendre en compte dans le calcul de distance global le fait que chaque point/primitive de D n'a pas nécessairement de point/primitive associé dans M , et éliminer certaines des distances locales. Ce cas se présente en particulier lorsqu'on veut fusionner les informations de plusieurs images prises à des instants ou selon des points de vue différents. Parfois, seules quelques primitives particulières de D sont utilisées pour le calcul de distance (points de contrôle).

Nous présentons en annexe A.II les calculs de distances inter-primitives figurant ci-dessous, correspondant aux cas de figures les plus souvent rencontrés, ainsi que des solutions au problème de calcul de distance dans le cas d'un recouvrement partiel, et l'utilisation de d'une sélection de points particuliers appelés points de contrôle pour la mise en correspondance :

- Calcul de distance entre deux ensembles de points dans le cas de recouvrement partiel
- Calcul de distance signée entre un ensemble de segments et un ensemble de points
- Calcul de distance entre un ensemble de points et un ensemble de plans
- Calcul de distance entre deux ensembles de triangles

Dans notre cas, le modèle M étant complet il n'y a pas de recouvrement partiel entre M et D , et chaque point de D a un point correspondant dans M . Ainsi toutes les distances locales point 3D/primitive de M sont prises en considération. Les calculs de distance relatifs aux types de modèles que nous avons utilisés seront détaillés plus loin.

III.4. Réduction du temps de calcul de distance

La phase la plus longue dans les algorithmes de mise en correspondance est d'établir la correspondance des primitives mais surtout les calculs de distance inter-primitives. S'il y a N_D entités dans l'ensemble de données et N_M entités (points, lignes, triangles) dans le modèle, la complexité de l'algorithme de calcul de distance est $N_M \times N_D$.

Quelques techniques sont proposées maintenant permettant de réduire efficacement la durée des calculs de distances.

III.4.1. Octrees

Cette méthode utilisée en modélisation volumique (voir Chapitre I, paragraphe II.1.1) n'est pas très adaptée, car nombre de voxels seront vides.

III.4.2. *k-D-Trees (K-Dimensional binary Trees) [Zhang 94]*

Les *k-D-trees* représentent des bisections successives d'un espace de dimension-*k* et la complexité du calcul de distance peut être réduite à $N_D \times \log N_M$.

Exemple de construction d'un 3D-Tree :

Choisir un plan parallèle au plan *yz* passant par le point de donnée *P*, pour couper l'espace des points en deux ensembles de même taille. Chaque partie obtenue est alors coupée par un plan parallèle au plan *xz* pour qu'il y ait le même nombre de points de chaque côté. On fait de même avec des plan parallèles au plan *xy* et ainsi de suite en choisissant à chaque fois un plan parmi les plans (*yz*, *xz*, *xy*). Le processus s'arrête lorsqu'on arrive sur un parallélépipède vide.

Illustration d'un 2D-Tree (figure II.1) :

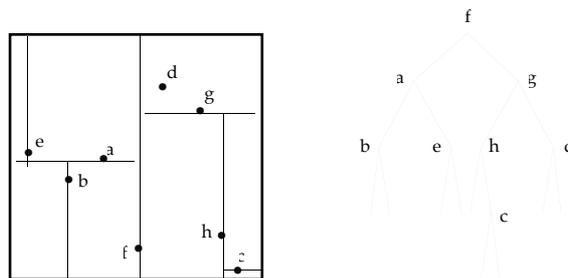


Figure II.1 : Espace 2D et 2D-Tree associé.

III.4.3. *Octree-spline [Lavallée 95]*

L'idée de ce type d'octree est d'obtenir plus de précision lorsqu'on se rapproche de la surface d'intérêt. Cette représentation combine l'avantage des fonctions splines adaptatives et des structures hiérarchiques octree.

La construction d'un octree-spline se fait de la façon suivante :

- a. Construction d'un octree avec les points du modèle,
- b. Subdivision : Homogénéisation de la taille des cubes proches de la surface,
- c. Calcul des distances signées pour chaque coin de cube octree,
- d. Elimination des discontinuités.

Le calcul de la distance d'un point P à M s'effectue ainsi : Chercher le noeud de l'octree le plus proche de P , puis utiliser une interpolation trilinéaire sur les 8 sommets du cube contenant P pour calculer la distance de P à M .

Dans l'espace du cube :

$$P(u, v, w) \text{ avec } (u, v, w) \in [0, 1] \times [0, 1] \times [0, 1]$$

$$d(P) = \sum_{i=0}^1 \sum_{j=0}^1 \sum_{k=0}^1 b_i(u) b_j(v) b_k(w) \vec{d}_{ijk} \quad \text{où } b_l(t) = \delta_l t + (1 - \delta_l)(1 - t)$$

III.4.4. Points cachés [Simon 94]

Cette amélioration du temps de calcul s'appuie sur le fait que les points de M et D qui sont proches à l'instant k le sont sûrement encore à l'instant $(k+1)$. Plutôt que de chercher dans M le point le plus proche du point D_i à l'instant k , on cherche les n points les plus proches de D_i et on les associe à D_i . Ainsi à l'instant $(k+1)$, il suffit de s'intéresser uniquement aux points cachés à l'étape précédente, plutôt que de reprendre tous les points de M en considération. Un test à chaque étape permet de vérifier que la distance converge bien, avec le point caché retenu.

III.5. Méthodes de mise en correspondance 3D

Sabata [Sabata 91] a proposé en 1991 une classification des méthodes de mise en correspondance 3D suivant le type de primitives utilisées et l'existence de correspondance points/primitives entre les formes. L'annexe B présente un état de l'art de ce domaine à partir de quelques méthodes proposées par Sabata et des développements récents dans le domaine de la mise en correspondance 3D, parmi lesquelles :

- Méthodes avec correspondance de points 3D

Ce cas est le plus courant, et de nombreuses méthodes existent pour trouver la bonne transformation, parmi lesquelles nous présentons des méthodes simples, des méthodes d'optimisation non-linéaire plus complexes et des méthodes adaptées au problème de la mise en correspondance.

- Méthodes sans correspondance de points

Ces méthodes partent du principe que l'on dispose de deux ensembles de points non appariés, en supposant que les mêmes points sont visibles dans les deux ensembles.

- Méthodes avec correspondance de primitives

Ces méthodes utilisent des primitives, ce qui permet d'éviter les problèmes liés à l'utilisation des points (erreurs de correspondance, recouvrement partiel, ...)

- Méthode ICP (Iterative Closest Point)

Cette méthode présentée séparément des autres est très utilisée dans le domaine de la mise en correspondance 3D. Elle permet d'appairer les points ou primitives de D avec les points ou primitives de M ; nous la présentons en détail, ainsi que les adaptations ou modifications récentes qui lui ont été apportées. C'est à partir de cet algorithme que nous avons développée notre méthode de mise en correspondance, que nous exposerons au paragraphe IV.

III.6. Estimateurs plus spécifiques

En plus des critères de distance présentés, il existe des estimateurs plus appropriés pour la mise en correspondance et pour la prise en compte des erreurs de mesure. Celles-ci peuvent être d'origines différentes : Erreur du capteur télémétrique, distorsion de caméra, quantification spatiale, ... Elles introduisent des erreurs de mise en correspondance et affectent les performances des applications. La prise en compte de ces erreurs s'effectue généralement en introduisant un septième paramètre (la variance de l'erreur) en plus des paramètres de transformation.

Les techniques robustes se caractérisent par le pourcentage maximum de points aberrants qui peut être toléré sans corrompre le résultat de la transformation recherchée, appelé point d'arrêt. Nous présentons en annexe A.III quelques estimateurs, dont l'estimateur aux moindres carrés (point d'arrêt à 0%), et l'estimateur médian (point d'arrêt à 50%). Nous présentons également un estimateur prenant en compte les bruits de mesure, ainsi que deux estimateurs garantissant un bon résultat dans des conditions particulières même avec plus de 50% de points aberrants. Nous avons choisi comme estimateur dans notre méthode, l'estimateur médian qui garantit une robustesse à 50% de points aberrants, quelque soit les conditions.

III.7. Choix de la transformation initiale

L'un des points délicats dans l'utilisation des méthodes non-linéaires est que, en fonction de la transformation initiale choisie, la méthode peut converger vers un minimum local qui n'est pas nécessairement le minimum global recherché, sauf pour certaines méthodes complètes et complexes telles que les méthodes de recuit simulé (Simulated Annealing, Very

Fast Simulated Annealing, ...). Dans la plupart des méthodes présentées précédemment, la transformation initiale est supposée proche de la transformation recherchée.

Voici quelques solutions pour choisir la transformation initiale :

- Utilisation des moments de l'image pour approcher la translation [Lavallée 94],
- Alignement interactif grossier [Lavallée 94],
- Rotation et translation approximativement estimées à partir des axes principaux de l'objet dans chaque vue,
- Navigation visuelle : Entre deux images le déplacement est faible (vitesse maximale de déplacement de l'objet limitée et fréquence d'échantillonnage des images suffisamment élevée) et connu avec une certaine précision (les véhicules mobiles sont aussi couramment équipés de différents instruments de mesure comme des odomètres ou des systèmes inertiels),
- Raisonnement de P. Besl [Besl 92] : La seule solution pour être sûr de trouver le minimum global est de rechercher le minimum de tous les minimums locaux. La répartition des minimums locaux dépendant de la forme de l'objet observé, il n'existe donc pas de méthode générale de partitionnement de l'espace de recherche en pôles d'attraction.

Le cas le plus simple est de prendre comme transformations initiales la transformation identité et les rotations de 180° autour des trois axes principaux de l'objet. Ces transformations initiales fonctionnent bien pour la plupart des objets.

Une autre possibilité est de choisir les rotations réparties uniformément sur un polyèdre régulier. Besl propose de choisir des combinaisons normalisées de quaternions :

$$q_0 = \{1, 0\}, q_1 = \{1, 0, -1\}, q_2 = \{1, 0, -1\}, q_3 = \{1, 0, -1\}, \text{ pour 40 transformations}$$

$$q_0 = \{1, 0.5, 0\}, q_1 = \{1, 0.5, 0, -0.5, -1\}, q_2 = \{1, 0.5, 0, -0.5, -1\}, q_3 = \{1, 0.5, 0, -0.5, -1\}$$

pour 312 transformations

Pour chacune de ces rotations initiales, peuvent s'ajouter des vecteurs translations uniformément répartis dans l'espace 3D, afin de décrire le plus complètement possible l'espace des minimums locaux.

S'il n'y a pas de connaissances a priori sur la transformation initiale, il est nécessaire d'exploiter le maximum de transformations initiales, afin d'éviter les minimums locaux pour

ne conserver que le minimum global. L'utilisation de transformations initiales est très coûteuse en temps de calcul, la complexité étant alors de l'ordre du nombre de rotations multiplié par le nombre de translations. La plupart des techniques présentées fonctionnent très bien à partir du moment où on est proche du minimum global.

A travers ces diverses possibilités de choix de transformations initiales, nous déduisons qu'il est possible d'éviter les minimums locaux dans le problème d'optimisation non-linéaire, mais que les solutions proposées sont très coûteuses en temps de calcul. Ainsi il peut être suffisant de demander une mise en correspondance initiale interactive grossière à un manipulateur et de concentrer les calculs sur la mise en correspondance finale.

IV. Méthode de mise en correspondance

Nous allons présenter maintenant la méthode générale de mise en correspondance, que nous avons développée. Cette méthode permet la mise en correspondance d'un ensemble de points R^I avec un modèle géométrique R^{II} de nature quelconque (ensemble de points, de courbes, de surfaces plus ou moins complexes) ; seule la notion de distance point/entité de R^{II} est à définir. Nous utilisons un estimateur LMS pour garantir à la méthode sa robustesse (robuste à 50% de points aberrants) [Masuda 95] et une adaptation de l'algorithme ICP de Besl (pour l'appariement des primitives de R^I avec celles de R^{II}) avec un échantillonnage aléatoire pour déterminer la transformation rigide. Nous présenterons donc successivement le principe général de la méthode, l'algorithme LMS, l'algorithme ICP et la détermination de la transformation rigide utilisant la méthode des quaternions.

IV.1. Principe de la méthode de mise en correspondance

Il a été discuté précédemment que l'inconvénient des méthodes d'optimisation non-linéaires pour la mise en correspondance est de ne pas garantir que le minimum local trouvé soit le minimum global recherché. Dans notre cas, ceci est souvent occasionné par la présence d'axes ou de plans de symétrie dans la représentation géométrique des objets. Pour cela, nous proposons de transformer préalablement l'ensemble R^I en N ensembles initiaux R_k^I sur lesquels va être effectuée une première mise en correspondance avec R^{II} . Ces N transformations initiales (T_k) sont choisies selon la méthode proposée par Besl [Besl 92] et sont représentées par des quaternions unitaires uniformément répartis sur la demi-sphère unitaire de \mathfrak{R}^3 .

Par exemple, pour 40 transformations, les quaternions initiaux sont obtenus par les combinaisons suivantes :

$$q_0 = \{1, 0\} \quad q_1 = \{1, 0, -1\} \quad q_2 = \{1, 0, -1\} \quad q_3 = \{1, 0, -1\}$$

L'algorithme de mise en correspondance peut être décomposé en quatre parties. Dans la première, les transformations rigides à partir des N ensembles initiaux sont déterminées en utilisant un échantillonnage aléatoire. La meilleure de ces transformations est conservée (partie 2), pour segmenter l'ensemble initial de points 3D en points aberrants et non-aberrants (partie 3). Celle-ci donne une valeur de distance proche du minimum global recherché, mais comme elle n'utilise pas tous les points (échantillonnage aléatoire), elle n'est pas exacte. Afin de déterminer la transformation exacte nous ajoutons une étape supplémentaire à l'algorithme de mise en correspondance. Dans cette quatrième partie, tous les points non-aberrants sont utilisés par un algorithme ICP simple, afin de déterminer la transformation exacte.

Principe de la méthode de mise en correspondance :

a. Pour R_k^I , $k=1$ à N , $T_k(LMS) = LMS(R_k^I, R^II)$

L'algorithme LMS, qui détermine la transformation rigide entre chaque R_k^I et R^II , est décrit paragraphe IV.2.

b. Choix de la meilleure transformation $T^*(LMS)$ parmi les N calculées.

La transformation $T^*(LMS)$, correspondant à la distance médiane minimum n'est qu'une approximation de la transformation recherchée, car elle n'utilise à chaque fois qu'une partie des points disponibles.

c. Sélection de l'ensemble PNA de points non-aberrants associés à $T^*(LMS)$.

Afin d'obtenir une transformation plus exacte, on extrait de l'image 3D initiale, après leur avoir appliqué $T^*(LMS)$, les points qui sont jugés aberrants, c'est-à-dire trop éloignés du modèle CAO.

D'après P. J. Rousseuw et A. M. Leroy [Rousseuw 87], la meilleure sélection des points 3D utilisant la distance médiane est obtenue avec le critère suivant :

$$PNA = \left\{ p_i \in R^I / d(T^*(p_i), R^II) < 2.5s \right\} \quad \text{avec} \quad s = 1.4826 \left(1 + \frac{5}{N_s} \right) \sigma_{med}$$

où σ_{med} est la distance médiane obtenue par la transformation $T^*(LMS)$; la distribution des résidus est supposée gaussienne.

d. $T_{ICP} \leftarrow ICP(PNA, T^*(LMS), R^II)$

Calcul de la transformation finale exacte par un algorithme ICP en utilisant PNA.

IV.2. Algorithme $LMS(R_k^I, R^II)$

Sur chacun des ensembles de points R_k^I est appliqué un algorithme LMS. Celui-ci extrait aléatoirement N_T sous-ensembles différents de N_S points de R_k^I et calcule, en utilisant un algorithme ICP, la transformation rigide entre ce sous-ensemble de points et R^II . La qualité de cette transformation est alors évaluée par un estimateur d'erreur médiane non-linéaire, théoriquement robuste à 50 % de bruit; seule la meilleure de ces transformations est conservée. L'échantillonnage aléatoire permet de trouver une bonne estimation de la transformation rigide, à partir de sous-ensembles de points extraits aléatoirement en souhaitant, lors des tirages, sélectionner un minimum de points aberrants.

Description de l'algorithme $LMS(R_k^I, R^II)$:

a. Répéter b. à e. N_t fois, N_t étant le nombre d'essais

b. $P_{RS}^I \leftarrow RS(R_k^I, N_S)$

Extraction aléatoire de N_S points de R_k^I

c. $T_{ICP} \leftarrow ICP(P_{RS}^I, T_k(LMS), R^II)$

Evaluation de T_{ICP}

L'ensemble de points P_{RS}^I est utilisé par l'algorithme ICP (présenté paragraphe IV.3) avec l'ensemble d'entités géométriques R^II et avec la meilleure transformation $T_k(LMS)$ pour estimer les paramètres de la transformation T_{ICP} .

d. $LMedS(R_k^I, T_{ICP}, R^II)$

Evaluation de T_{ICP} par $LMedS(R_k^I, T, R^II) = \sqrt{\text{med}_{r_k^I \in R_k^I} \left(d(T(r_k^I), R^II)^2 \right)}$,

d étant la distance Point/Modèle géométrique.

e. Si $LMedS(R_k^I, T_{ICP}, R^II) < LMedS(R_k^I, T_k(LMS), R^II)$

alors $T_k(LMS) \leftarrow T_{ICP}$.

IV.3. Algorithme $ICP(P, \bar{T}, R^II)$

L'algorithme ICP adapté permet d'estimer efficacement la transformation rigide entre un ensemble de points P et R^II connaissant une estimée de la transformation recherchée (\bar{T}). Cet algorithme peut être divisé en deux parties : La première phase consiste à trouver pour

chaque point de P transformé par \bar{T} , son point le plus proche dans R^H . Ces points de R^H constituent alors un nouvel ensemble de points P^H . La seconde phase sera alors de déterminer la transformation rigide entre ces deux ensembles de points associés en utilisant la représentation par quaternions. Ces deux opérations seront répétées jusqu'à ce que P et R^H soient suffisamment proches par la transformation rigide estimée. Ce procédé itératif s'arrêtera lorsque $d_{l-1} - d_l < \tau \cdot \sigma$ où $d_l = d(P^l, P^H)$ est la somme des carrés des distances entre les points de P^l et P^H . La valeur de seuil τ représente la tolérance de la convergence. Le facteur d'échelle σ permet d'obtenir une condition de convergence sans dimension. Il représente la taille approximative de P et est défini comme étant la racine carrée de la trace de la matrice de covariance de P. Pour l'algorithme ICP, il a été prouvé que la séquence des d_k décroît d'une façon monotone [Besl 92].

L'algorithme ICP peut être décrit ainsi :

a. $l \leftarrow 1 ; P^l = \bar{T}(P) ; d_0 \leftarrow \infty.$

\bar{T} est appliquée sur P

b. $P^H \leftarrow C(P^l, R^H)$

Établir la correspondance des points

c. $T_{ICP} \leftarrow Q(P^l, P^H)$

La transformation entre les ensembles de points P^l et P^H est estimée par la méthode des quaternions (voir paragraphe IV.4).

d. $P^l = T_{ICP}(P)$

On applique la transformation estimée à l'ensemble de points P

e. $d_l = d(P^l, P^H)$

f. Les étapes b. à e. sont répétées jusqu'à ce que $d_{l-1} - d_l < \tau \cdot \sigma$ avec $l \leftarrow l + 1$.

L'algorithme ICP est utilisé deux fois dans l'algorithme de mise en correspondance :

- Dans LMS : $T_{ICP} \leftarrow ICP(P_{RS}^l, T_k(LMS), R^H)$

- Partie d. de l'algorithme de mise en correspondance :

$$T_{ICP} \leftarrow ICP(PNA, T^*(LMS), R^H)$$

IV.4. Estimation de la transformation rigide utilisant les quaternions

Les paramètres de la transformation rigide entre les ensembles de points P^I et P^{II} doivent minimiser la fonction de coût suivante :

$$\frac{1}{N_s} \sum_{i=1}^{N_s} \|p_i^{II} - T(p_i^I)\|^2$$

p_i^I est un point de P^I et p_i^{II} est le point de P^{II} qui lui est associé.

Une technique de résolution du problème de minimisation très efficace utilisant la représentation par quaternion a été introduite par Faugeras et Hebert [Faugeras 86] et utilisée plus tard par Besl et McKay [Besl 92]. La représentation par quaternion fait que la minimisation de la fonction de coût est équivalente à la maximisation de la forme quadratique du quaternion-unité.

Méthode des quaternions [Faugeras 86]

Soit un ensemble de points de mesure $P = \{\vec{p}_i\}$ et un ensemble de points de modèle $X = \{\vec{x}_i\}$ appairés.

Soit les centres de gravité $\vec{\mu}_p = \frac{1}{N} \sum_{i=1}^N \vec{p}_i$ et $\vec{\mu}_x = \frac{1}{N} \sum_{i=1}^N \vec{x}_i$

La matrice de cross-covariance Σ_{px} entre les deux ensembles de points est donnée par :

$$\Sigma_{px} = \frac{1}{N} \sum_{i=1}^N [(\vec{p}_i - \vec{\mu}_p)(\vec{x}_i - \vec{\mu}_x)^t]$$

La matrice anti-symétrique $A_{ij} = (\Sigma_{px} - \Sigma_{px}^t)_{ij}$ est utilisée pour former le vecteur $\Delta = \begin{bmatrix} A_{23} \\ A_{31} \\ A_{12} \end{bmatrix}$

qui intervient dans la matrice 4 x 4 symétrique :

$$Q(\Sigma_{px}) = \begin{bmatrix} tr(\Sigma_{px}) & \Delta^t \\ \Delta & \Sigma_{px} + \Sigma_{px}^t - tr(\Sigma_{px})I_3 \end{bmatrix}$$

où I_3 est la matrice identité 3 x 3.

Le quaternion vecteur propre unitaire correspondant à la valeur propre maximum de la matrice $Q(\Sigma_{px})$ fournit la rotation optimale.

Le vecteur translation est donné par : $T = \vec{\mu}_x - R \cdot \vec{\mu}_p$.

Cette méthode ne produit une solution que si au moins trois paires de points sont données.

Conclusion

Nous avons abordé dans ce chapitre le problème de la mise en correspondance de données D avec un modèle M . Nous avons présenté un rapide état de l'art sur la mise en correspondance 2D, puis avons approfondi le problème de la mise en correspondance 3D qui est l'objet de notre travail. Nous avons dissocié ce problème en différents points pour lesquels nous avons établi un état de l'art complet et récent : le type de représentation de la transformation rigide recherchée, les primitives de D et de M à mettre en correspondance, le calcul de distance associé aux primitives sélectionnées pour D et M , le critère à optimiser et la méthode d'optimisation choisie, le choix de la transformation initiale, ainsi que des solutions pour accélérer le calcul de distance. Nous avons ensuite proposé notre propre méthode de mise en correspondance 3D. Celle-ci présente les avantages suivants : elle est automatique, robuste, et fonctionne pour la mise en correspondance de données et modèles de types quelconques et dans des positions initiales quelconques. Seule la distance entre le modèle choisi et les données est à définir.

Dans les deux chapitres suivants, nous nous proposons de présenter le modèle et les données que nous avons utilisés, et évaluer notre méthode sur des exemples afin de tester sa sensibilité aux variations des paramètres intrinsèques ainsi que sa robustesse.

Chapitre III

Données 3D et Modèles CAO

La méthode de mise en correspondance, que nous avons développée et présentée lors du chapitre précédent, va être utilisée avec des données et modèles réels. Les points données 3D, obtenus soit par numérisation d'un objet à l'aide d'un capteur de profondeur 3D, soit par une machine à mesurer 3D, vont être mis en correspondance avec leur modèle CAO dans une représentation triangulée ou dans une représentation exacte. Pour les données 3D, nous parlerons des données 3D que nous avons le plus utilisées c'est à dire celles fournies par un capteur télémétrique.

Toutes les données 3D et les modèles CAO des objets utilisés dans cette thèse sont présentés en Annexe C. Pour cette étude, nous disposons de deux types d'objets :

- Objets de forme simple : Objets comprenant des surfaces planes ou cylindriques, servant principalement pour tester la méthode de mise en correspondance,
- Objets de forme complexe : Objets d'origine industrielle comprenant des surfaces gauches et peu de surfaces planes.

I. Données 3D

Les données 3D qui vont servir pour la mise en correspondance sont obtenues en deux temps. Tout d'abord, plusieurs images de l'objet étudié sont réalisées à l'aide d'un capteur de profondeur 3D à partir de différents points de vue. Dans un deuxième temps, ces informations sont fusionnées pour constituer un ensemble de points 3D décrivant complètement l'objet. Nous concluons par un paragraphe sur les erreurs de mesure du capteur télémétrique utilisé. Pour quelques applications de la méthode de mise en correspondance, nous avons utilisé des données 3D provenant d'une machine à mesurer.

I.1. Système d'acquisition des images 3D

Le système d'acquisition 3D utilisé [Rioux 88] consiste en un système à triangulation laser (figure III.1). Les performances de ce système sont optimisées par l'utilisation d'une technique de synchronisation utilisant un galvanomètre à miroir, telle que la projection du faisceau laser et sa détection s'effectuent de façon synchrone. Une particularité du système est d'utiliser un miroir recouvert sur ses deux faces d'une couche réfléchissante, la première étant utilisée pour la projection du laser sur la scène, tandis que la deuxième assure la synchronisation de la détection. Bien qu'une lumière incohérente puisse être utilisée pour la projection, il est préférable de se servir d'une source laser afin de bénéficier de la très grande profondeur de champ procurée par la cohérence spatiale.

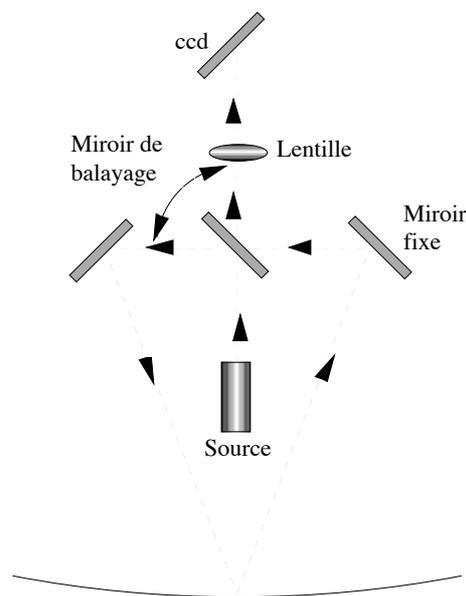


Figure III.1 : Système de mesure 3D utilisant la triangulation et l'autosynchronisation.

Le profil des objets dans le champs de vision est mesuré, à partir de la surface de référence définie par les axes de projection et de détection. Chaque mesure fournit les coordonnées x,y,z de l'intersection du faisceau projeté avec la surface de l'objet ainsi que l'intensité de la lumière diffusée (image d'intensité). Ces informations sont obtenues grâce à l'utilisation d'un détecteur de position (une barrette CCD linéaire) et d'une lentille correctrice. La position du faisceau diffusé sur la barrette CCD fournit la coordonnée z (donnée d'élévation ou de distance), tandis que la position du miroir de balayage ainsi que la position de la caméra (le long de l'axe y) donnent respectivement les coordonnées x et y . L'information d'intensité est obtenue par l'amplitude du signal détecté sur la barrette CCD, laquelle est proportionnelle à la réflectivité de la surface de l'objet. La barrette CCD est inclinée par rapport à l'axe optique afin de compenser la défocalisation qui se produit le long de l'axe z .

Nous avons utilisé dans cette étude un capteur commercialisé par la société Hymarc Ltd (Hyscan-45c) disposant de la technologie développée au Conseil National de Recherche du Canada (décrite ci-dessus), d'une résolution de $\pm 25 \mu\text{m}$ pour une profondeur de champ de 100 mm, d'une largeur de balayage de 70 mm et d'une vitesse d'acquisition de 10 000 points/s. Le fichier d'informations fourni par le système d'acquisition contient l'ensemble de points numérisés non-ordonnés, ainsi que leur intensité respective, et d'éventuelles informations de couleur.

A partir des informations géométriques et des intensités d'une image 3D, la scène peut être reconstruite et visualiser en 3D à l'aide d'un éclairage artificiel (figure III.2).

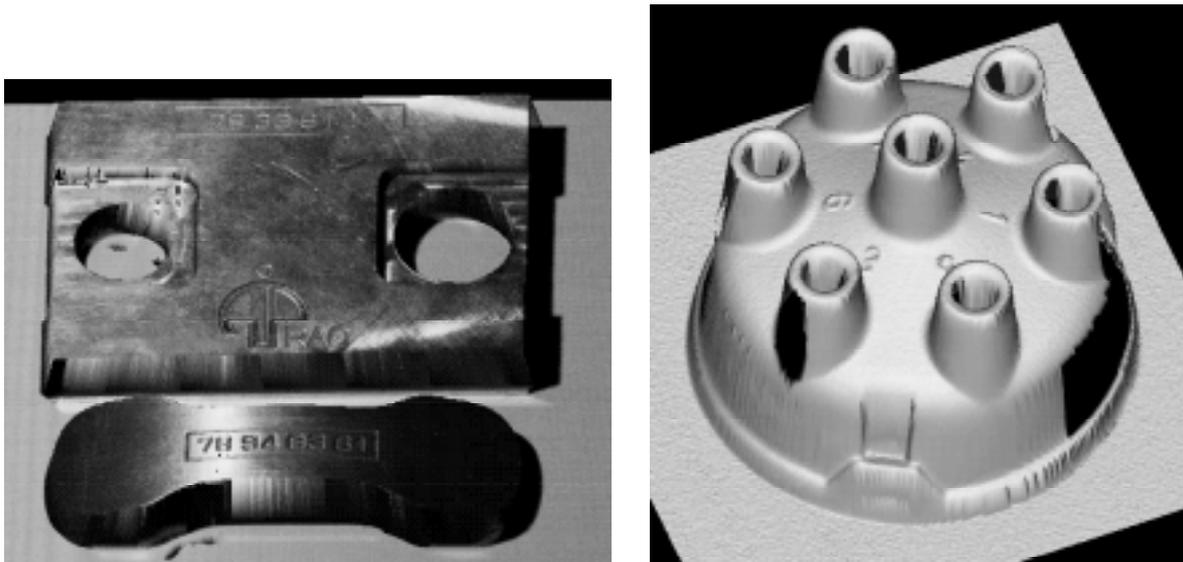


Figure III.2 : Images reconstruites à partir des informations géométriques et d'intensité d'objets numérisés.

Des informations géométriques de l'image 3D de la figure III.2 (à gauche), on peut séparer celles concernant les deux objets d'intérêt (figure III.3) du fond de l'image.

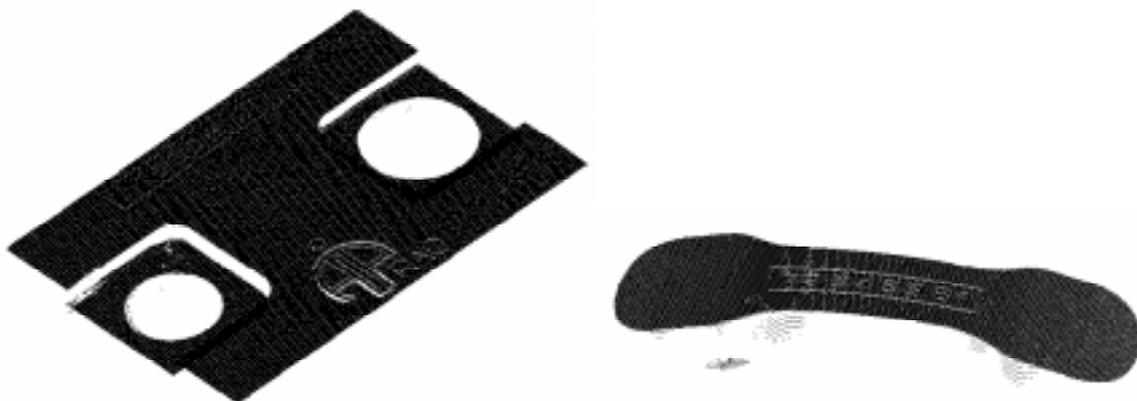


Figure III.3 : Informations géométriques extraites pour chaque objet de l'image 3D de la figure III.2 gauche.

I.2. Mise en correspondance des images 3D

En fonction du type d'application visée, une seule image 3D est rarement suffisante, aussi la fusion de plusieurs images 3D s'avère nécessaire pour décrire de façon complète les surfaces apparentes de l'objet étudié. Cette opération de fusion s'effectue par mise en correspondance des images 3D.

Mathématiquement le problème est le suivant : Disposant de N images 3D $\{I_1, \dots, I_N\}$, l'image de référence étant I_1 , il faut trouver les transformations $T_i, i = 2, \dots, N$ à appliquer sur les I_i , pour minimiser les distances entre I_i et I_1 .

Beaucoup de travaux actuels portent sur la fusion de données provenant de différentes images, sans connaissance a priori sur leur positionnement relatif, et ceci principalement pour la construction de modèles 3D [Bergevin 95] [Bittar 93] [Blais 95] [Chen 92] [Dorai 94] [Godin 94] [Godin 95] [Higuchi 94] [Jacq 95] [Masuda 95],... Ces techniques fonctionnent plutôt bien lorsque les formes décrites sont assez complexes, mais dans le cas d'objets simples ou industriels disposant de surfaces planes, cylindriques ou circulaires, elles sont parfois mises en défaut par la présence de nombreuses symétries (glissement plan sur plan, rotation cylindre/cylindre, rotation sphère/sphère, ...).

Dans notre cas, le capteur est fixé sur une machine à mesurer; sa position et son orientation sont ainsi connues avec précision. Des images de la scène sont réalisées, à partir de différentes positions du capteur, chacune étant exprimée dans le référentiel du capteur, qui lui est parfaitement connu par rapport au référentiel de la machine à mesurer. Il est ainsi facile de fusionner les informations provenant de ces images, afin d'obtenir un ensemble de points 3D (données 3D) décrivant presque complètement l'objet (point de fixation ou surface d'appui mis à part) (figure III.4). Cette technique simple, appelée mise en correspondance mécanique, fonctionne très bien quelque soit la complexité de l'objet étudié, mais nécessite un dispositif de fixation du capteur extrêmement précis et donc onéreux.

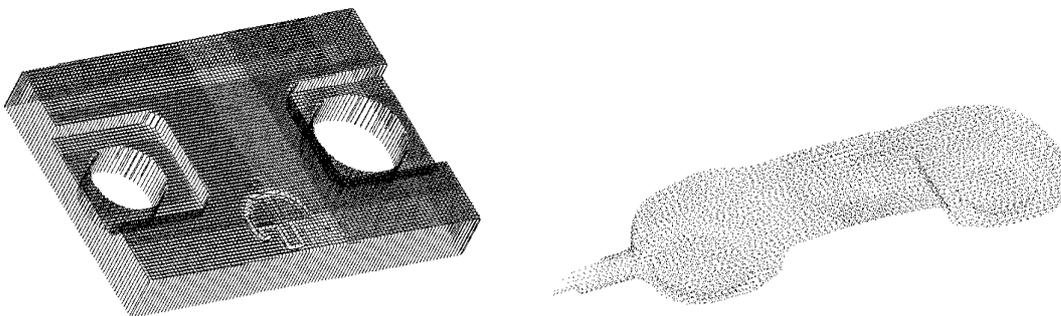


Figure III.4 : Ensembles de points 3D obtenus après fusion de plusieurs images 3D, pour les objets *Support* et *Téléphone*.

L'objectif de ce travail n'étant pas la mise en correspondance d'images 3D, celle-ci a été effectuée de façon mécanique par la société Hymarc Ltd lors de la numérisation des objets.

I.3. Bruit de mesure sur les points 3D

Boullanger a présenté dans [Boullanger 94] un modèle géométrique du télémètre utilisé :

Une surface peut être définie mathématiquement ainsi :

$$\bar{\eta}(r, s) = \left\{ (x, y, z) \in \mathfrak{R}^3 \mid \left(x = h_x(r, s), y = h_y(r, s), z = h_z(r, s) \right), \mathbf{V}(r, s) \in \mathfrak{R}^2 \right\}$$

ou h_n est une fonction qui établie une correspondance entre les coordonnées paramétriques (r, s) et le système de coordonnées réelles (x, y, z) situé dans un repère centré objet T_0 .

Les capteurs géométriques produisent des images télémétriques correspondant à des estimées de la forme géométrique : $\bar{r}(u, v) = (x\mathcal{Q}(u, v), y\mathcal{Q}(u, v), z\mathcal{Q}(u, v))$, corrompue par un bruit géométrique \bar{e} .

$$\bar{r}(u, v) = O \left[R \left(\bar{\eta}(a(u, v), b(u, v)) - \bar{t} \right) \right] + \bar{e}$$

où :

- O : Opérateur d'occlusion qui masque la partie de l'objet non-vue
- R : Matrice de rotation
- \bar{t} : Vecteur translation
- $a(u, v)$ et $b(u, v)$: Re-paramétrisation de la surface sous-jacente de sorte à ce que la paramétrisation (r, s) de la surface observée par le capteur, corresponde à la paramétrisation (u, v) produite par le capteur.

Le bruit géométrique \bar{e} peut être causé par différents processus statistiques qui dépendent du type de capteur utilisé et du type de surface mesurée. Le bruit géométrique est défini comme étant tout processus non prévisible par un modèle mathématique continu. A partir de cette définition, le bruit dans les signaux géométriques peut être décomposé en différentes contributions : $\bar{e} = \bar{e}_n + \bar{e}_q + \bar{e}_s + \bar{e}_r + \bar{e}_0$

où :

- \bar{e}_n : Bruit gaussien provenant du capteur,
- \bar{e}_q : Bruit de quantification,
- \bar{e}_s : Bruit systématique introduit par les imperfections du calibrage du capteur,
- \bar{e}_r : Bruit ne suivant pas une distribution normale (rugosité de surface).

La plus grande partie du bruit géométrique peut être analysée par ce modèle simple. En revanche, une certaine partie des données très corrompues par le bruit ne correspondent pas à une hypothèse de continuité. En particulier, dans certaines conditions les réflexions spéculaires ou multiples, les angles apparents abrupts, les discontinuités de profondeur, peuvent produire de mauvaises mesures, non-modélisables par un modèle probabiliste simple. Ces mesures sont appelées points aberrants et le bruit \vec{e}_0 correspondant est appelé bruit de points aberrants.

Pour le type de capteur utilisé (Hymarc 45c), l'erreur globale \vec{e} pour chaque point est d'environ $\pm 25 \mu m$.

II. Modèles CAO

Les modélisateurs géométriques des logiciels de CAO utilisent généralement deux types de représentation. La première forme est une représentation solide (Constructive Solid Geometry) où le modèle est construit par assemblage et/ou opérations logiques sur des formes paramétrées simples (cylindres, parallélépipèdes, ...). La seconde est une représentation surfacique (Boundary REPresentation) où seules les surfaces frontières de l'objet sont modélisées. C'est ce dernier mode de représentation que nous avons retenu (voir chapitre I). L'algorithme de mise en correspondance présenté pouvant utiliser n'importe quel type de modes de représentation, nous en avons implémenté deux, en fonction de la précision de représentation et de la rapidité de calcul souhaitées. Dans le premier, chaque surface du solide est décrite par un ensemble de triangles (triangulation STL) planaires, dont la densité de tessellation varie en fonction de l'erreur d'approximation. Dans le second, la géométrie des objets est décrite de façon exacte par des surfaces NURBS découpées. Ces deux modes de représentation géométrique ont l'avantage d'être des standards dans l'industrie et sont représentés par le format STL (format utilisé en prototypage rapide) pour le mode par triangulation et par le format IGES (Initial Graphics Exchange Specification - format d'échange des modélisateurs CAO) pour les surfaces NURBS.

II.1. Modèle triangulé STL

Le format STL est un format dédié à la stéréolithographie ou prototypage rapide [Dolenc 94]. La définition CAO de l'objet étudié est exportée à partir du logiciel de CAO au format IGES, puis converti au format STL par un logiciel externe. Les surfaces exactes du modèle IGES sont alors triangulées en respectant des règles propres à la modélisation STL; le nombre de triangles ainsi créés dépend de l'erreur maximale surface/triangle tolérée. Le modèle dont on dispose alors est un ensemble de triangles définis par leurs trois sommets et leurs normales orientées vers l'extérieur de l'objet. Ce modèle peut être ensuite découpé en tranches pour être interprété par les machines de prototypage rapide.

II.1.1. Contraintes de la modélisation STL

Dans un modèle STL correct (Figure III.5), chaque triangle a exactement un voisin le long de chaque arête, et les triangles ne sont autorisés à *s'intersecter* que le long d'une arête ou en un sommet. Grâce à ces conditions, il est possible de distinguer l'intérieur ou l'extérieur de l'objet modélisé.

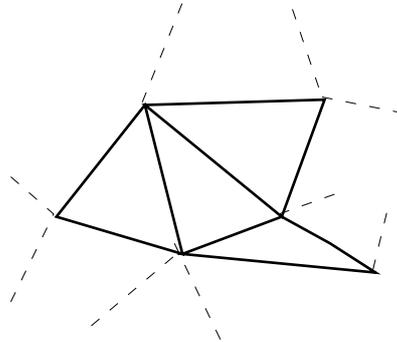


Figure III.5 : Triangulation STL correcte.

Le modèle STL est incorrect lorsqu'il contient des trous dus à l'absence de certains triangles : les facettes peuvent *s'intersecter* incorrectement, la même arête peut être partagée par plus de deux facettes ou des facettes peuvent se superposer (cas topologique non-manifold). Les logiciels qui produisent des fichiers STL n'informent pas toujours que le modèle généré n'est pas correct (figure III.6), les défauts ne sont alors détectés que lors du prototypage.

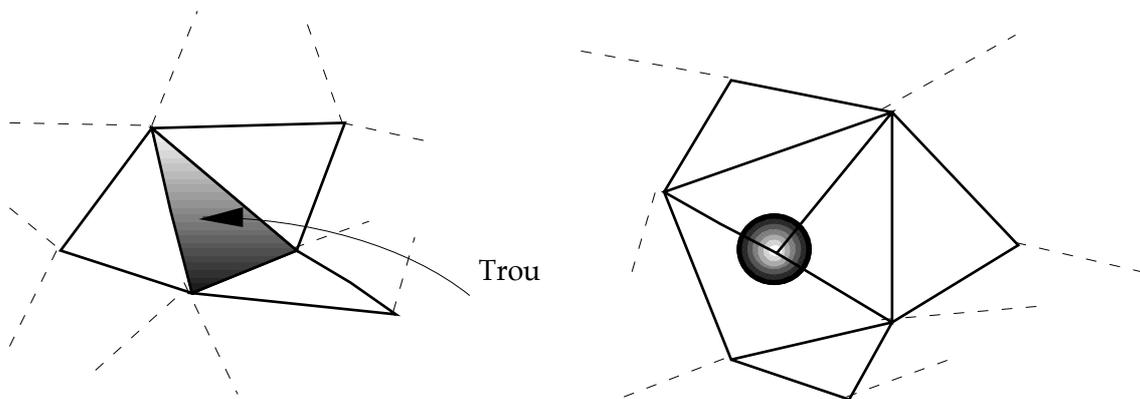


Figure III.6 : Triangulations incorrectes.

II.1.2. Exemples de triangulation STL

A partir de l'objet *Cylindre* défini en CAO, nous effectuons plusieurs conversions en modèle STL, avec différentes valeurs d'approximation. Nous obtenons des modèles du type de celui de la figure III.7 avec plus ou moins de triangles.

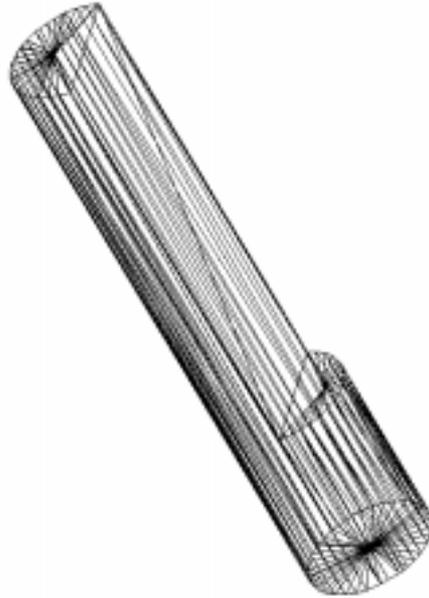


Figure III.7 : Exemple de triangulation STL.

Ci dessous, on peut constater l'évolution du nombre de triangles STL en fonction de l'erreur maximale d'approximation souhaitée :

Erreur d'approximation (mm)	Nombre de triangles
0,0005	8506
0,001	6554
0,005	2688
0,01	3144
0,05	1784
0,1	646
0,5	326
1	338
2	314

Tableau.III.1 : Nombre de triangles STL en fonction de l'erreur d'interpolation souhaitée.

D'un point de vue global, le nombre de triangles STL augmentent lorsque l'erreur d'approximation diminue, surtout dans les zones de faible courbure de surface. Compte tenu des règles de modélisation STL à respecter, pour des erreur d'approximation proches, cette règle n'est pas toujours vérifiée (tableau III.1), ceci s'explique par une réorganisation totale des triangles.

II.1.3. Calcul de distance point/modèle STL

Pour accélérer le calcul de distance minimal entre un point M et un triangle, nous avons mis au point une technique de représentation de l'espace à base de voxels contenant tous les triangles.

Calcul de la distance d'un point $M(x_M, y_M, z_M)$ à un triangle

Une facette triangulaire (figure III.8) est définie par ses trois sommets $P_1(x_1, y_1, z_1)$, $P_2(x_2, y_2, z_2)$ et $P_3(x_3, y_3, z_3)$.

Soit $P(x, y, z)$, la projection orthogonale de M sur la facette :

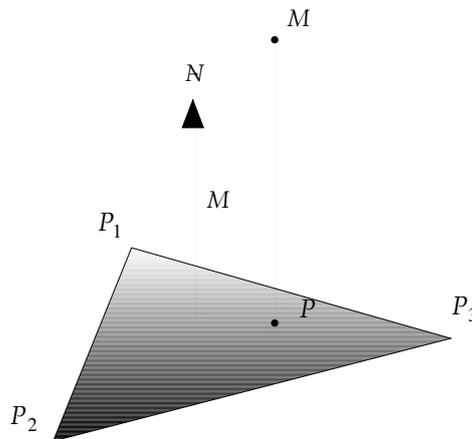


Figure III.8 : Facette triangulaire avec ses sommets et sa normale orientée.

Trois cas peuvent alors se présenter :

- Si P se situe à l'intérieur du triangle, l'équation suivante est vérifiée :

$$\vec{P_1P} = u \cdot \vec{P_1P_2} + v \cdot \vec{P_1P_3} \quad \text{avec} \quad \begin{cases} u \geq 0 \\ v \geq 0 \\ u + v \leq 1 \end{cases} \quad \text{où} \quad u = \frac{\begin{vmatrix} \vec{N} & \vec{MP_1} & \vec{P_3P_1} \end{vmatrix}}{\begin{vmatrix} \vec{N} & \vec{P_2P_1} & \vec{P_3P_1} \end{vmatrix}} \quad \text{et} \quad v = \frac{\begin{vmatrix} \vec{N} & \vec{P_2P_1} & \vec{MP_1} \end{vmatrix}}{\begin{vmatrix} \vec{N} & \vec{P_2P_1} & \vec{P_3P_1} \end{vmatrix}}$$

alors le point le plus proche de M est le point P défini par :

$$\vec{OP} = \vec{OM} + k \cdot \vec{N}, \quad \text{avec} \quad k = \frac{\begin{vmatrix} \vec{MP_1} & \vec{P_2P_1} & \vec{P_3P_1} \end{vmatrix}}{\begin{vmatrix} \vec{N} & \vec{P_2P_1} & \vec{P_3P_1} \end{vmatrix}}.$$

- P n'est pas dans le triangle (figure III.9) :

Soient P_A, P_B, P_C , les projections orthogonales de M sur les 3 cotés de la facette :

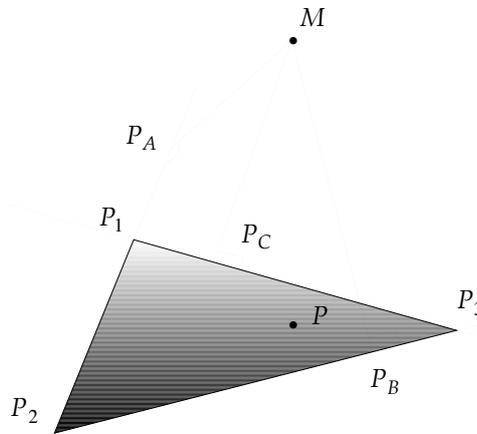


Figure III.9 : Projection du point M sur les 3 cotés du triangle.

Soit k_1, k_2, k_3 , définies par les relations suivantes :

$$\begin{aligned} \vec{P_1P_A} &= k_1 \cdot \vec{P_1P_2} \\ \vec{P_2P_B} &= k_2 \cdot \vec{P_2P_3} \\ \vec{P_3P_C} &= k_3 \cdot \vec{P_3P_1} \end{aligned} \text{ d'où : } k_1 = \frac{\vec{P_1M} \cdot \vec{P_1P_2}}{\|\vec{P_1P_2}\|^2}, k_2 = \frac{\vec{P_2M} \cdot \vec{P_2P_3}}{\|\vec{P_2P_3}\|^2}, k_3 = \frac{\vec{P_3M} \cdot \vec{P_3P_1}}{\|\vec{P_3P_1}\|^2}$$

Le point le plus proche de M est parmi P_A, P_B et P_C le point dont la distance à M est la plus petite, et qui vérifie $0 \leq k_i \leq 1$.

- S'il n'existe aucun point projeté pour lequel $0 \leq k_i \leq 1$, alors le point de la facette le plus proche de M est l'un de ses sommets.

Accélération du calcul de distance point/ensemble de triangles.

La méthode de calcul de distance précédemment présentée peut être accélérée en choisissant une structure de données plus appropriée présentée au chapitre 2, (paragraphe III.4). Dans notre cas, nous avons divisé l'espace des triangles en un nombre paramétrable de voxels élémentaires auxquels sont attribués deux types d'informations : Les coordonnées du centre de gravité du voxel, ainsi que la liste des triangles l'intersectant lui-même ou ses voxels voisins. En utilisant cette représentation, le calcul de distance entre un point M et le modèle triangulé consiste d'abord à trouver le voxel le plus proche de M, puis à appliquer le calcul de distance point/triangles aux seuls triangles pointés par ce voxel.

Pour déterminer si un triangle, défini par ses deux paramètres u et v et ses sommets $P_1(x_1, y_1, z_1)$, $P_2(x_2, y_2, z_2)$ et $P_3(x_3, y_3, z_3)$, intersecte un parallélépipède de centre de gravité $G(X_G, Y_G, Z_G)$ de dimensions (L_x, L_y, L_z) , il suffit de trouver un point M vérifiant :

- La condition d'appartenance au triangle :

$$\vec{OM} = \vec{OP}_1 + u \cdot \vec{P}_1P_2 + v \cdot \vec{P}_1P_3 \quad \text{avec } 0 \leq u \leq 1, 0 \leq v \leq 1 \quad \text{et} \quad u + v \leq 1$$

- et la condition d'appartenance au parallélépipède :

$$\begin{cases} x_G - \frac{L_x}{2} < x_1 + u \cdot (x_2 - x_1) + v \cdot (x_3 - x_1) < x_G + \frac{L_x}{2} \\ y_G - \frac{L_y}{2} < y_1 + u \cdot (y_2 - y_1) + v \cdot (y_3 - y_1) < y_G + \frac{L_y}{2} \\ z_G - \frac{L_z}{2} < z_1 + u \cdot (z_2 - z_1) + v \cdot (z_3 - z_1) < z_G + \frac{L_z}{2} \end{cases}$$

Pour qu'un point appartenant à une facette soit détecté, il est nécessaire que le balayage de la facette en u et v s'effectue avec un incrément plus petit que la plus petite dimension d'un parallélépipède élémentaire.

II.2. Modèle surface NURBS découpée

Les surfaces NURBS (Non Uniform Rational B-Spline) sont souvent utilisées pour leur grande capacité descriptive et l'exactitude de leur représentation. En effet, une sphère, un plan, un ellipsoïde peuvent être représentés par une seule surface NURBS. Quand aux objets complexes, ils peuvent être modélisés par un assemblage de plusieurs surfaces NURBS découpées, qui délimitent ainsi un volume fermé exact.

II.2.1. Définition

Une surface NURBS est définie de façon paramétrique [Farin 92b] et [Faux 81] :

$$\vec{s}(u, v) = \frac{\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} \vec{d}_{ij} N_i^k(u) N_j^l(v)}{\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} N_i^k(u) N_j^l(v)}$$

où :

- u et v sont les paramètres de la surface,
- n_i : le nombre de points de contrôle dans la direction du paramètre u ,

- n_j : le nombre de points de contrôle dans la direction du paramètre v ,
- w_{ij} : le poids associé au point de contrôle \vec{d}_{ij} ,
- \vec{d}_{ij} : le point de contrôle,
- N_i^k : les fonctions de la base B-Spline définies par la formule récurrente suivante :

$$N_i^k(u) = \frac{u - u_{i-1}}{u_{i+k-1} - u_{i-1}} N_i^{k-1}(u) + \frac{u_{i+k} - u}{u_{i+k} - u_i} N_{i+1}^{k-1}(u) \quad N_i^0(u) = \begin{cases} 1 & \text{si } u_{i-1} \leq u \leq u_i \\ 0 & \text{sinon} \end{cases}$$

- u_i, v_i : les noeuds appartenant aux vecteurs de noeuds de la surface NURBS,
 $u_i \in [u_0, u_1]$ et $v_i \in [v_0, v_1]$.

De façon analogue, on peut définir une courbe NURBS qui sera utilisée ultérieurement comme courbe de découpe complexe d'une surface NURBS.

$$\vec{s}(u) = \frac{\sum_{i=0}^{n_i} w_i \vec{d}_i N_i^k(u)}{\sum_{i=0}^{n_i} w_i N_i^k(u)}$$

où :

- u : est le paramètre de la courbe s ,
- n_i : le nombre de points de contrôle dans la direction du paramètre u ,
- w_i : le poids associé au point de contrôle \vec{d}_i ,
- \vec{d}_i : le point de contrôle,
- u_i : le noeud appartenant au vecteur de noeuds de la courbe NURBS $u_i \in [u_0, u_1]$.

Afin de mieux décrire les surfaces de l'objet modélisé, surtout au niveau des raccordements de leurs surfaces ou des perçages, sans augmenter le degré des surfaces ou le nombre de leurs paramètres, les surfaces NURBS sont découpées par des courbes de découpe intérieures ou extérieures plus ou moins complexes (figure III.10).

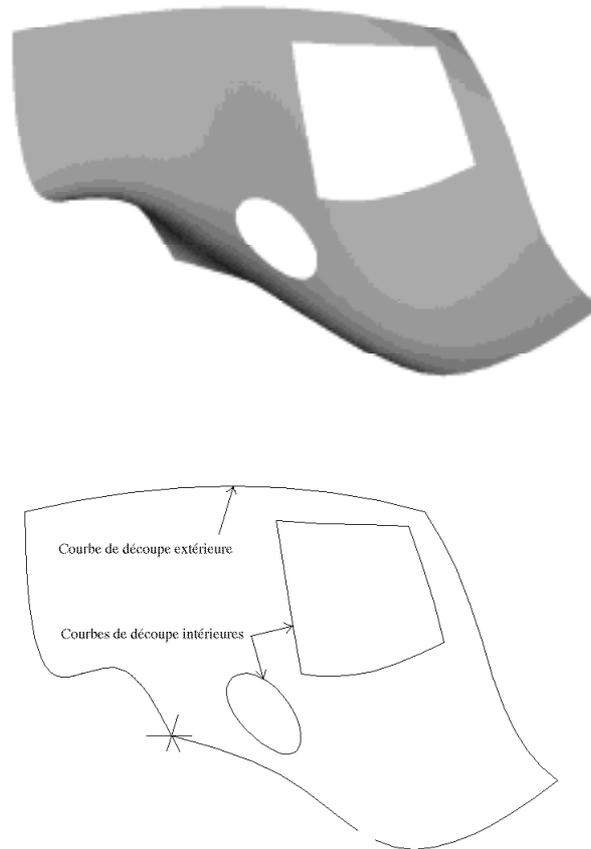


Figure III.10. : Surface NURBS découpée (haut), courbes de découpe (bas)

II.2.2. Calcul de distance point/surface NURBS découpée

Nous avons développé une méthode permettant la mesure de distance d'un point \vec{r} à une surface NURBS découpée. Celle-ci consiste dans un premier temps à trouver le point de la surface non découpée le plus proche de \vec{r} et à situer ce point par rapport aux courbes de découpe, pour savoir s'il appartient ou non à la surface découpée. Le cas échéant, le point de la surface découpée le plus proche de \vec{r} est un point situé sur une des courbes de découpe.

Calcul de la distance point/surface NURBS non-découpée

Afin de déterminer la distance d'un point à une surface NURBS non-découpée, nous avons utilisé une méthode de calcul de distance entre un point et une surface quadrique, développée par P. Boulanger dans [Boulanger 94] et que nous avons appliquée aux surfaces NURBS.

Déterminer la distance d'un point \vec{r} à une surface revient à trouver la paramétrisation (u_0, v_0) de la surface telle que la distance de $\vec{s}(u_0, v_0)$ à \vec{r} soit minimale dans la direction normale au plan tangent à cette surface.

Le critère à minimiser est alors le suivant : $\min_{u_0, v_0} \|\vec{r} - \vec{s}(u_0, v_0)\|^2$

La surface étant initialement découpée en N carreaux, on cherche le centre du carreau (de paramètres u et v) le plus proche de \vec{r} , afin d'initialiser les paramètres du point recherché $u_0 = u$ et $v_0 = v$ pour appliquer l'algorithme d'optimisation non-linéaire. Ce partitionnement initial est nécessaire afin d'éviter les minimums locaux du critère de distance apparaissant pour des surfaces dont le degré est supérieur à deux dans au moins une des deux directions de paramétrisation.

En limitant le développement de Taylor au premier ordre de la surface $\vec{s}(u, v)$, on obtient :

$$\vec{s}(u_0, v_0) = \vec{s}(u, v) + \frac{\partial \vec{s}}{\partial u}(u_0 - u) + \frac{\partial \vec{s}}{\partial v}(v_0 - v)$$

avec :

$$\frac{\partial}{\partial u} \vec{s}(u, v) = \frac{\left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} \vec{d}_{ij} \frac{\partial}{\partial u} N_i^k(u) N_j^l(v) \right) \left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} N_i^k(u) N_j^l(v) \right) - \left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} \frac{\partial}{\partial u} N_i^k(u) N_j^l(v) \right) \left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} \vec{d}_{ij} N_i^k(u) N_j^l(v) \right)}{\left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} N_i^k(u) N_j^l(v) \right)^2}$$

$$\frac{\partial}{\partial v} \vec{s}(u, v) = \frac{\left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} \vec{d}_{ij} N_i^k(u) \frac{\partial}{\partial v} N_j^l(v) \right) \left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} N_i^k(u) N_j^l(v) \right) - \left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} N_i^k(u) \frac{\partial}{\partial v} N_j^l(v) \right) \left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} \vec{d}_{ij} N_i^k(u) N_j^l(v) \right)}{\left(\sum_{i=0}^{n_i} \sum_{j=0}^{n_j} w_{ij} N_i^k(u) N_j^l(v) \right)^2}$$

$$\frac{\partial}{\partial u} N_i^k(u) = \frac{k}{u_{i+k-1} - u_{i-1}} N_i^{k-1}(u) - \frac{k}{u_{i+k} - u_i} N_{i+1}^{k-1}(u)$$

$$\frac{\partial}{\partial v} N_j^l(v) = \frac{l}{u_{j+l-1} - u_{j-1}} N_j^{l-1}(v) - \frac{l}{u_{j+l} - u_j} N_{j+1}^{l-1}(v)$$

Le critère à minimiser devient :

$$\min_{u_0, v_0} \left\| \vec{r} - \vec{s}(u, v) - \frac{\partial \vec{s}}{\partial u}(u_0 - u) - \frac{\partial \vec{s}}{\partial v}(v_0 - v) \right\|^2$$

Sous forme matricielle, ce critère peut s'exprimer ainsi : $\min_{u_0, v_0} \|J\vec{w} - \vec{d}\|^2$

où, J la matrice Jacobienne de $\vec{s}(u, v)$, est égale à : $J = \begin{bmatrix} \frac{\partial s_x}{\partial u} & \frac{\partial s_x}{\partial v} \\ \frac{\partial s_y}{\partial u} & \frac{\partial s_y}{\partial v} \\ \frac{\partial s_z}{\partial u} & \frac{\partial s_z}{\partial v} \end{bmatrix}$ et $\vec{w} = \begin{bmatrix} u_0 - u \\ v_0 - v \end{bmatrix}$

Le vecteur $\vec{d}(u, v)$ est le vecteur d'erreur pour la paramétrisation originale (u, v) :

$$\vec{d}(u, v) = \vec{r} - \vec{s}(u, v)$$

La solution de l'équation de minimisation correspond à la condition d'Euler, c'est-à-dire :

$$\frac{\partial}{\partial \vec{w}} \|J\vec{w} - \vec{d}\|^2 = 0 = 2J^T J\vec{w} - 2J^T \vec{d}$$

D'où : $J^T J\vec{w} = J^T \vec{d}$

où $J^T \vec{d}$ correspond à la projection du vecteur d'erreur dans le plan tangent. La nouvelle position paramétrique est égale à : $\vec{w} = (J^T J)^{-1} J^T \vec{d}$

La procédure est itérative; le critère d'arrêt correspond à la condition d'orthogonalité du vecteur d'erreur et du plan tangent à la surface en ce point, mesurée pour les deux conditions suivantes :

$$\left| \frac{\pi}{2} - \arccos \frac{\vec{d} \cdot \frac{\partial \vec{s}}{\partial u}}{|\vec{d}| \cdot \left| \frac{\partial \vec{s}}{\partial u} \right|} \right| \leq \varepsilon \quad , \quad \left| \frac{\pi}{2} - \arccos \frac{\vec{d} \cdot \frac{\partial \vec{s}}{\partial v}}{|\vec{d}| \cdot \left| \frac{\partial \vec{s}}{\partial v} \right|} \right| \leq \varepsilon$$

Le paramètre ε correspond à la précision avec laquelle on veut que cette condition d'orthogonalité soit respectée. Par expérience, comme compromis entre vitesse d'exécution et précision, on fixe $\varepsilon = 0,1$ degré.

L'algorithme peut être résumé comme suit :

a. Calcul de la matrice Jacobienne J,

b. $\vec{d}(u_0, v_0) = \vec{r} - \vec{s}(u_0, v_0)$,

c. $\vec{w} = (J^T J)^{-1} J^T \vec{d}$,

d. $(u_0, v_0) = (u_0, v_0) + \vec{w}$,

e. $\gamma = \arccos \frac{\vec{d} \cdot \frac{\partial \vec{s}}{\partial u}}{|\vec{d}| \cdot \left| \frac{\partial \vec{s}}{\partial u} \right|}$, $\xi = \arccos \frac{\vec{d} \cdot \frac{\partial \vec{s}}{\partial v}}{|\vec{d}| \cdot \left| \frac{\partial \vec{s}}{\partial v} \right|}$,

f. Si $\left| \frac{\pi}{2} - \arccos \frac{\vec{d} \cdot \frac{\partial \vec{s}}{\partial u}}{|\vec{d}| \cdot \left| \frac{\partial \vec{s}}{\partial u} \right|} \right| \leq \varepsilon$ et $\left| \frac{\pi}{2} - \arccos \frac{\vec{d} \cdot \frac{\partial \vec{s}}{\partial v}}{|\vec{d}| \cdot \left| \frac{\partial \vec{s}}{\partial v} \right|} \right| \leq \varepsilon$, alors on arrête, sinon on retourne à

l'étape a.

Détermination de la distance entre un point $M(x,y,z)$ et une surface NURBS découpée

Prenons le cas de la surface découpée décrite figure III.11, celle-ci présente une courbe de découpe extérieure et deux courbes de découpe intérieures.

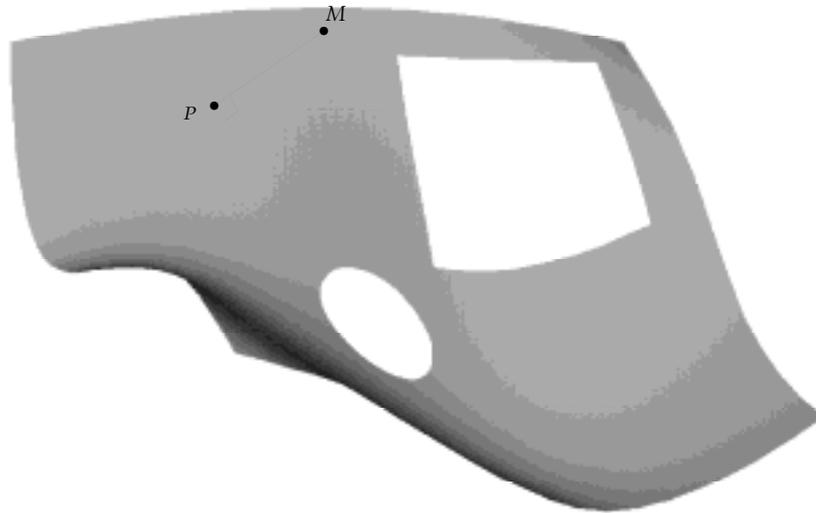


Figure III.11 : Projection du point M en P sur la surface NURBS.

Il est nécessaire dans un premier temps de déterminer le point P appartenant à la surface NURBS non-découpée, pour lequel la distance PM est minimum. Pour cela, on utilise l'algorithme de calcul de distance point/surface présenté ci-dessus.

Afin de déterminer si le point P trouvé appartient à la surface NURBS découpée, on cherche le point C_1 de la courbe de découpe extérieure et le point C_2 d'une des courbes de découpe intérieures pour lesquels les distances C_1P et C_2P sont minimales. On se place ensuite dans l'espace 2D des paramètres u et v . Dans ce plan (figure III.12), le sens de description des courbes NURBS est différent, s'il s'agit de courbes de découpe intérieures ou extérieures (sens des aiguilles d'une montre ou sens inverse). Soit \vec{n}_1 le vecteur normal à la courbe de découpe extérieure en C_1 et \vec{n}_2 le vecteur normal en C_2 à la courbe de découpe intérieure la plus proche.

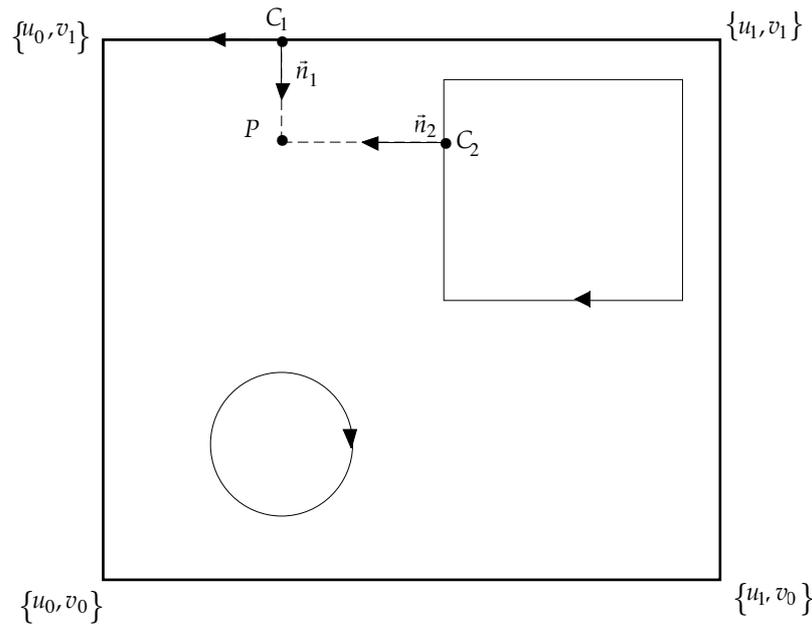


Figure III.12 : Surface et courbes de découpe dans l'espace des paramètres 2D.

Le point P appartient à la surface NURBS découpée si
$$\begin{cases} \vec{n}_1 \cdot \vec{C}_1 P \geq 0 \\ \vec{n}_2 \cdot \vec{C}_2 P \geq 0 \end{cases},$$

sinon le point P recherché est le point des courbes de découpe le plus proche de M . Dans ce cas, on adapte l'algorithme de calcul de distance point/surface NURBS au calcul de distance point/courbe NURBS.

II.3. Compromis modèle STL/modèle NURBS

De par la complexité des entités à manipuler en mode de représentation NURBS, les temps de calcul sont beaucoup plus longs qu'en mode de représentation STL. Par ailleurs, le mode de représentation STL n'est pas exact et les erreurs de représentation interviennent plus ou moins dans l'évaluation des distances point/Modèle pendant la mise en correspondance.

II.4. Complémentarité modèle STL/modèle NURBS

Les deux modèles peuvent être utilisés simultanément, afin de hiérarchiser la représentation géométrique, dans l'objectif de rendre plus efficace le calcul de distance point/modèle géométrique.

Nous avons utilisé cette simplification pour les applications d'inspection (chapitre VI).

Il est ainsi possible de précalculer :

- une correspondance triangle STL/surface NURBS découpée, car les sommets de chaque triangle STL appartiennent à la même surface NURBS découpée.
- une correspondance voxel/ensemble de triangles (paragraphe II.1.3).

Ainsi, Lorsqu'on recherche le point d'une surface le plus proche d'un point 3D P (figure III.13) :

- On recherche le voxel V le plus proche de P
- Ce voxel pointe sur une série de triangles (précalculée) parmi lesquels on recherche celui qui est le plus proche de P
- Les sommets de ce triangle appartiennent à une surface NURBS découpées, le point recherché est situé dans une région proche des trois sommets du triangle.

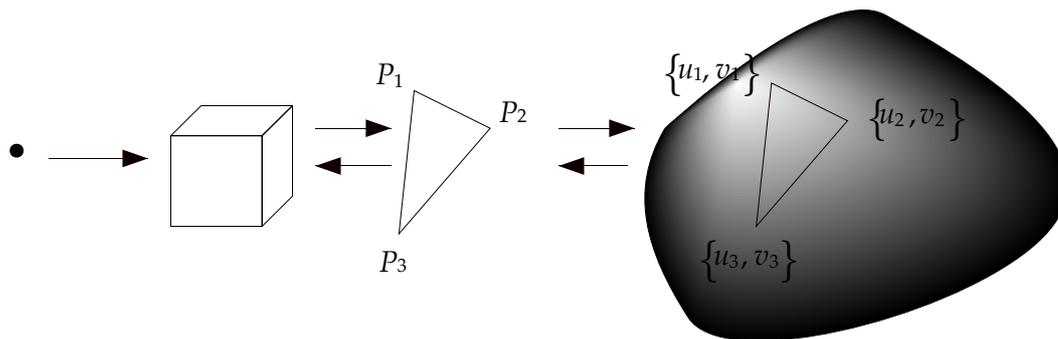


Figure III.13 : Représentation hiérarchique point/surface NURBS découpée.

Conclusion

Dans ce chapitre, nous avons présenté, les matériels et logiciels nécessaires à l'obtention des modèles et données 3D, ainsi que les formats utilisés. Notre apport a été dans la définition du calcul des calculs de distance point/ensemble de triangles et point/ensemble de surfaces NURBS découpées, dont la connaissance est nécessaire pour la mise en correspondance. Par ailleurs, nous avons défini une hiérarchisation de la représentation du modèle géométrique surfacique, qui nous a permis de trouver une méthode d'accélération du calcul de distance point/modèle.

Chapitre IV

Evaluation de la méthode de Mise en Correspondance

Introduction

La validation d'un algorithme nécessite de tester son efficacité et sa robustesse par sa mise en oeuvre sur différents types de données. Celle-ci inclut une étude du comportement de l'algorithme en fonction de ses paramètres internes, en présence de bruit, ainsi qu'une estimation des erreurs de calcul.

L'évaluation que nous proposons a été réalisée en utilisant deux modèles CAO : *Cylindre* et *Support* à mettre en correspondance avec des ensembles de points les décrivant de façon exacte. Afin de se placer dans les meilleures conditions pour étudier la méthode, nous n'utiliserons que des points "synthétiques", c'est à dire générés artificiellement sur les surfaces de l'objet (pas de bruit occasionné par l'acquisition des données ou de points aberrants causés par l'imperfection de l'objet), et des modèles STL présentant une faible erreur de modélisation. A l'état initial, les points 3D et le modèle CAO sont confondus, nous appliquons une transformation initiale T_0 connue sur ces points afin de les séparer du modèle, la transformation à retrouver est alors la transformation inverse $[T_0]^{-1}$.

Les critères d'évaluation de la méthode sont les suivants : Un critère de distance représentant la distance moyenne de chaque point au modèle CAO après mise en correspondance et un critère de temps de calcul de l'opération de mise en correspondance. Les calculs ont été effectués sur une station de travail Silicon Graphics Indy munie d'un processeur RISC Mips R5000SC 64 bits (150 Mhz).

Nous nous proposons d'évaluer les points suivants :

- Justesse et fidélité de la méthode,
- Sources des erreurs de calcul,
- Comportement de la méthode en fonction de ses paramètres internes,
- Robustesse au bruit.

I. Justesse et fidélité

D'un point de vue théorique, on peut juger un algorithme sur les propriétés suivantes :

- sa précision : aptitude à donner des résultats proches des vrais résultats,
- sa justesse : aptitude à donner un résultat égal au résultat idéal,
- sa fidélité : aptitude à aboutir aux mêmes résultats à partir de données identiques, dans les mêmes conditions.

Si on considère les résultats fournis par l'algorithme répartis comme des impacts de tirs sur une cible, dont le résultat idéal figure au centre de la cible, les propriétés de l'algorithme peuvent s'illustrer ainsi (figure IV.1) :

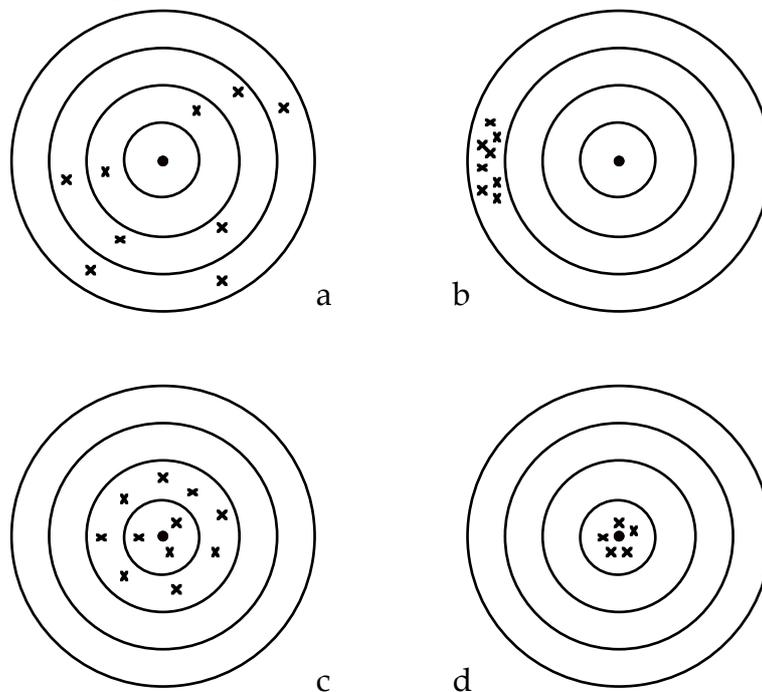


Figure IV.1 : (a) Tir imprécis, erreur de justesse et de fidélité, (b) Tir fidèle mais pas juste, (c) Tir juste mais pas fidèle, (d) Tir précis juste et fidèle

Afin de tester le comportement de notre algorithme vis à vis de ces deux propriétés, nous avons effectué les deux essais suivants : Le premier consiste à répéter la même mise en correspondance un grand nombre de fois afin de tester la répétitivité de la méthode, le second consiste à comparer les résultats de mise en correspondance de deux formes à partir de positions relatives initiales différentes.

I.1. Premier essai

Objet : Cylindre

Nous avons répété 30 fois l'opération de mise en correspondance suivante :

Données 3D	: Points idéaux de "Cylindre"
Modèle STL	: Erreur 0.0005 mm
N_t	: 5
N_s	: 30
Translation initiale	: {10,10,10}
Rotation initiale	: 15° autour de {1,1,1}

Nous relevons la distribution d'erreurs de position suivante (figure IV.2) :

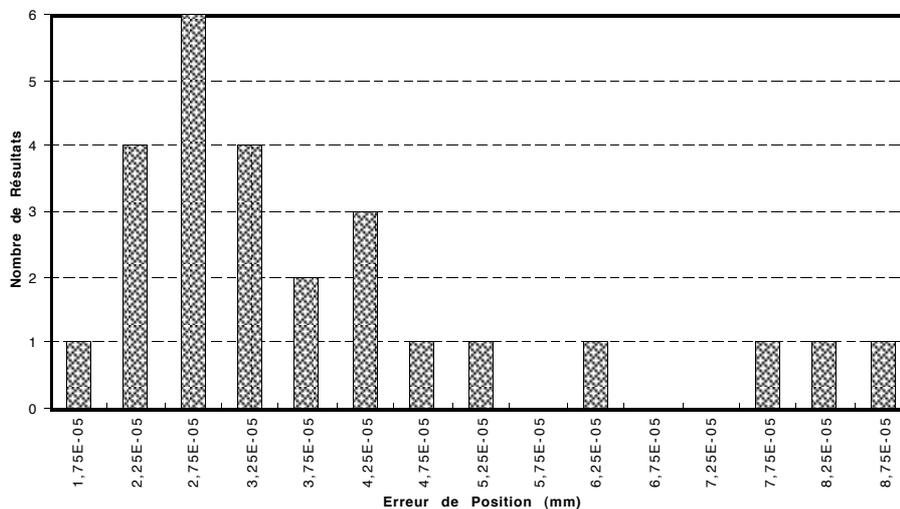


Figure IV.2 : Distribution des erreurs de position (*Cylindre*)

D'où nous déduisons une erreur de position moyenne de 0,035 μm avec un écart-type de 0,019 μm . Le temps de calcul moyen est de 26:41 mn avec un écart type de 33 s.

Objet Support

{	Données 3D	: Points idéaux de "Support"
	Modèle STL	: Erreur 0.005 mm
	N_t	: 5
	N_s	: 30
	Translation initiale	: {10,10,10}
	Rotation initiale	: 15° autour de {1,1,1}

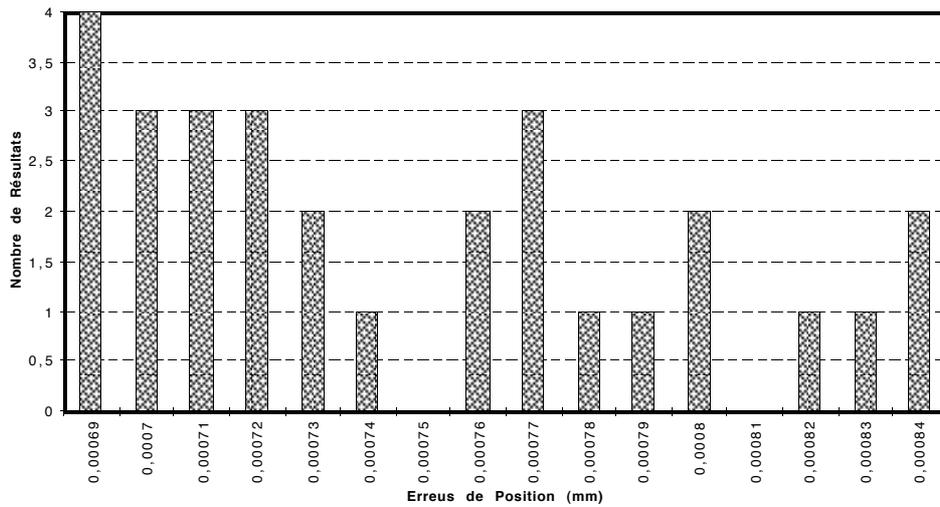


Figure IV.3 : Distribution des erreurs de position (*Support*)

D'où nous déduisons une erreur de position moyenne de 0,74 μm avec un écart-type de 0,048 μm (figure IV.3). Le temps de calcul moyen est de 13:56 mn avec un écart-type de 1:21 mn.

I.2. Deuxième essai

En plus de la justesse et de la fidélité de la méthode, nous vérifions maintenant le comportement de la méthode en fonction du positionnement initial relatif des deux formes à mettre en correspondance.

Nous avons répété la mise en correspondance suivante :

{	Données 3D	: Points idéaux de "Support"
	Modèle STL	: Erreur 0.005 mm
	N_t	: 3
	N_s	: 30
	Translation initiale	: variable
	Rotation initiale	: variable

en utilisant des transformations initiales différentes à chaque fois, combinant :

- un angle de rotation de 30° ou 135° ,
- un axe de rotation parmi les quatre axes suivants : $(1,1,1)$ $(1,1,-1)$ $(1,-1,1)$ et $(-1,1,1)$,
- une translation parmi les sept translations suivantes : $(0,0,100)$, $(0,100,0)$, $(100,0,0)$, $(0,100,100)$, $(100,0,100)$, $(100,100,0)$ et $(100,100,100)$.

Soit au total 56 transformations initiales.

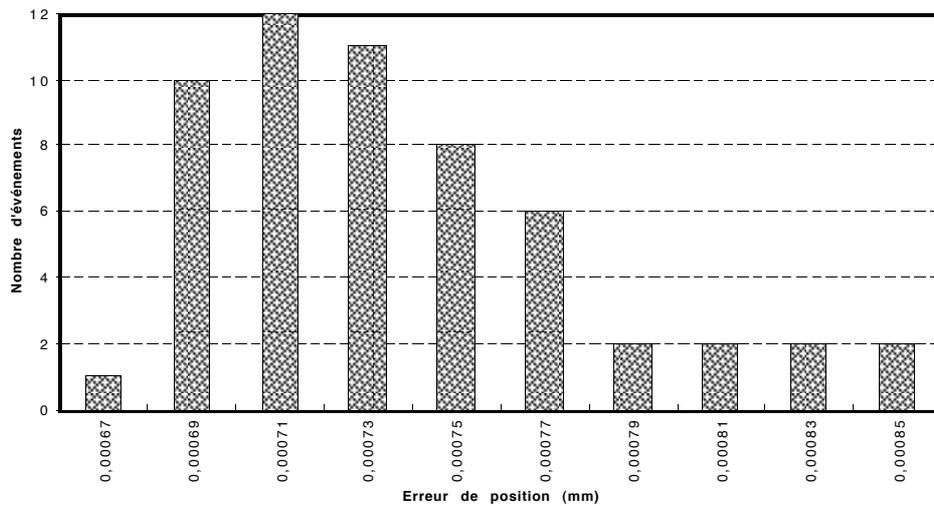


Figure IV.4: Distribution de l'erreur de position (Support).

Nous obtenons (figure IV.4) une erreur moyenne de $0,73 \mu\text{m}$ (écart-type de $0,04 \mu\text{m}$) et un temps de calcul moyen de 1 h 26 mn 53 s (écart-type de 11 mn 41 s).

La difficulté dans le deuxième essai de mise en correspondance réside dans l'évitement des minimums locaux du critère de distance à minimiser, occasionnées par la présence d'axes ou de plans de symétrie dans la description des pièces utilisées (figure IV.5). Ceci explique également le temps de calcul plus élevé.

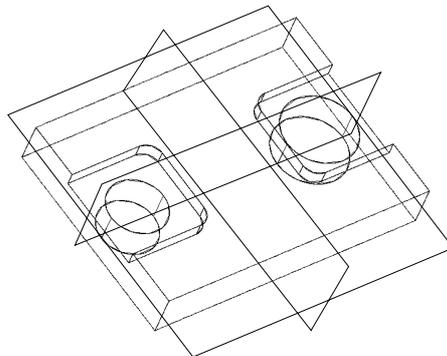


Figure IV.5: Plans de symétrie ou de pseudo-symétrie de l'objet *Support*.

Au vue des résultats obtenus au cours de ces deux essais du point de vue de la répétitivité, nous pouvons affirmer que la méthode est fidèle et juste et que l'erreur de calcul introduite par la méthode de mise en correspondance est minime par rapport aux applications envisagées. Par ailleurs, la méthode est capable d'éviter de conduire à des mauvaises solutions au problème de mise en correspondance occasionnées par l'existence de plans (ou pseudo-plans) de symétrie, sur l'objet étudié.

II. Sources des erreurs de calcul

Pour une bonne utilisation de la méthode de mise en correspondance, il est nécessaire de connaître l'influence des différentes erreurs de calcul sur le résultat de la mise en correspondance.

L'erreur de calcul globale est une contribution de ces deux types d'erreurs :

- Erreur de la méthode de mise en correspondance seule,
- Erreur de modélisation STL.

Nous allons essayer d'évaluer individuellement l'influence de chacune de ces sources d'erreurs.

II.1. Erreur de la méthode de mise en correspondance seule

Afin de tester l'erreur de la méthode seule, il faut que l'erreur de modélisation STL soit négligeable (erreur d'approximation minime, modèle STL comportant un grand nombre de triangles) et que points et modèle CAO se superposent exactement (erreur idéale nulle). A partir de cela, nous allons étudier l'erreur de mise en correspondance de la méthode en fonction du paramètre de convergence de l'algorithme ICP (chapitre II.IV.3), c'est à dire la valeur $\varepsilon = \tau \cdot \sigma$.

II.1.1. Objet Support

Données 3D	: Points idéaux de "Support"
Modèle STL	: Erreur 0.005 mm
N_t	: 10
N_s	: 100
Translation initiale	: $\{10, 10, 10\}$
Rotation initiale	: 10° autour de $\{1, 1, 1\}$

Les valeurs de $\varepsilon = \tau \cdot \sigma$ sont prises dans l'intervalle suivant : $\varepsilon \in [10^{-10}, 5 \cdot 10^{-2}]$

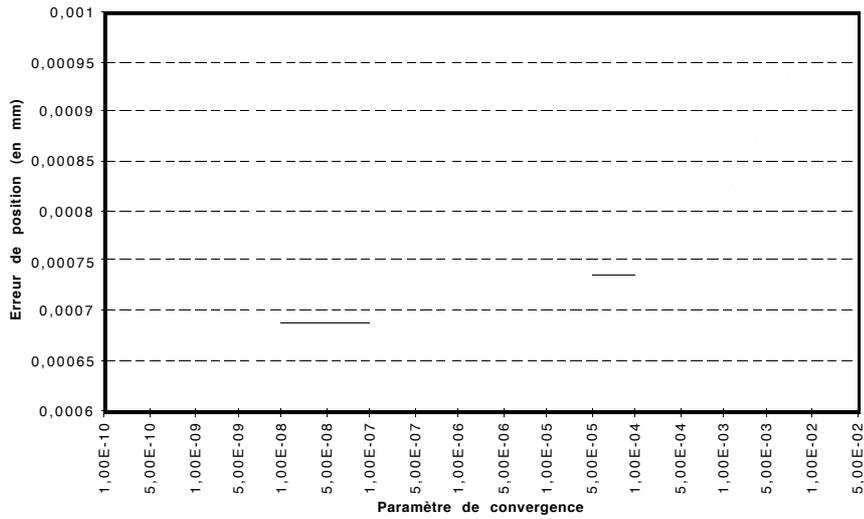


Figure IV.6 : Erreur de position en fonction du paramètre ϵ .

II.1.2. Objet Cylindre

{	Données 3D	: Points idéaux de "Cylindre"
	Modèle STL	: Erreur 0.0005 mm
	N_t	: 10
	N_s	: 100
	Translation initiale	: {10, 10, 10}
	Rotation initiale	: 15° autour de {1, 1, 1}

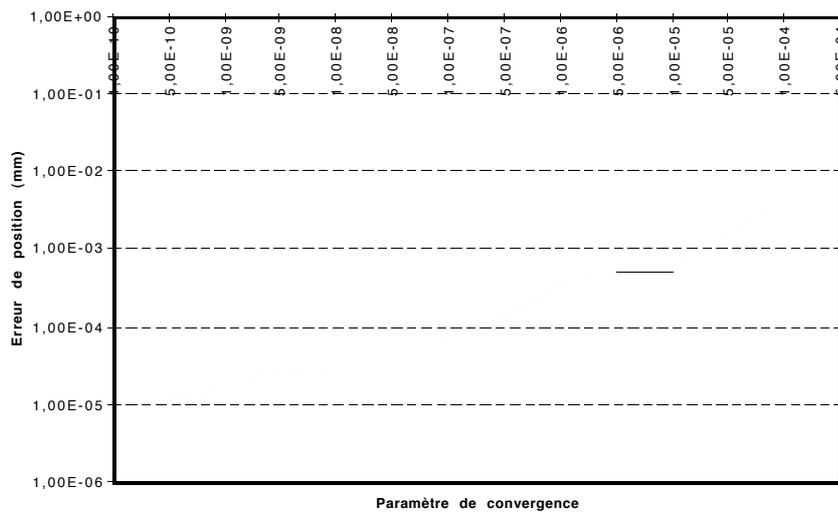


Figure IV.7 : Erreur de position en fonction du paramètre ϵ .

En choisissant $\varepsilon = 10^{-8}$ (figures IV.6, IV.7), on peut affirmer que la contribution de la méthode à l'erreur de position finale est inférieure à $0,7 \mu\text{m}$ pour le *Support* et à $0,025 \mu\text{m}$ pour le *Cylindre*, ce qui est vraiment minime pour le type d'applications envisagées.

II.2. Erreur de modélisation STL

Nous avons vu lors de la présentation de la modélisation STL, que le paramètre d'approximation d'un ensemble de surfaces à l'aide de triangles est l'erreur maximale triangle/surface admissible.

Plus l'erreur d'approximation est faible, plus il y a de triangles dans le modèle STL et plus l'erreur moyenne points/modèle est faible (fig. IV.8).

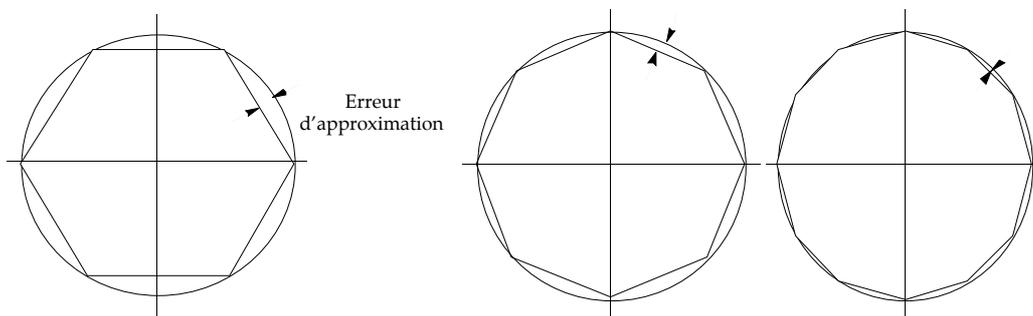


Figure IV.8 : Illustration de l'évolution du nombre de triangles en fonction de l'erreur d'interpolation souhaitée (dans le plan de coupe d'un cylindre).

Pour plusieurs des modèles STL étudiés, nous calculons l'erreur moyenne points/modèle STL, en utilisant des points 3D "idéaux" générés synthétiquement sur les surfaces exactes de l'objet.

II.2.1. Objet "Cylindre"

Nous nous contentons dans l'essai suivant, de calculer sans mise en correspondance l'erreur points 3D/modèle STL, pour différents modèles en utilisant les points idéaux.

Données 3D	: Points idéaux de "Cylindre"
Modèle STL	: Erreur d'approximation variable
N_t	: 0
N_s	: 30
Translation initiale	: $\{0,0,0\}$
Rotation initiale	: 0°

Nous obtenons les erreurs moyennes suivantes :

Erreur d'approximation (mm)	Nombre de triangles	Erreur de position (μm)
0,0005	8506	0,0076
0,001	6554	0,015
0,005	2688	0,066
0,01	3144	0,1
0,05	1784	0,7
0,1	646	1,2
0,5	326	5,7
1	338	10,6
2	314	16,1

Tableau IV.1 : Erreur de position en fonction du nombre de triangles et de l'erreur d'approximation.

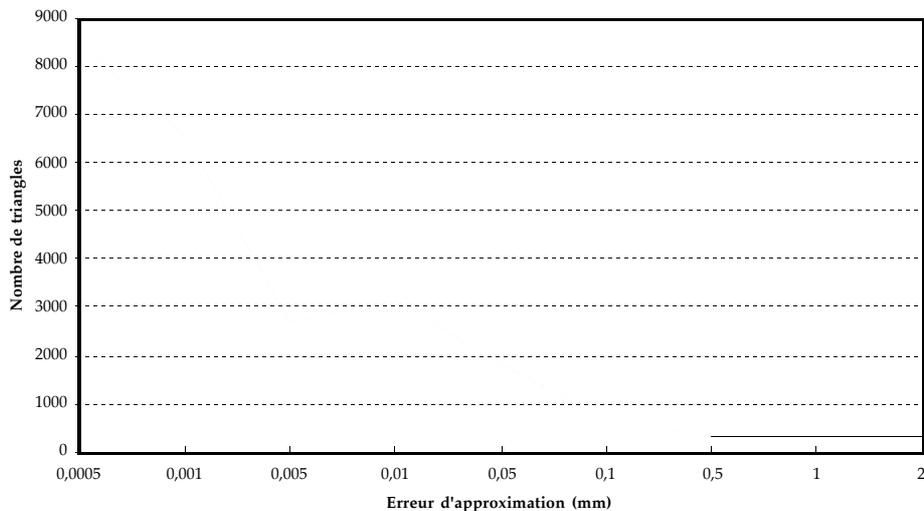


Figure IV.9 : Evolution du nombre de triangles en fonction de l'erreur d'approximation STL (Cylindre).

D'une manière générale, le nombre de triangles STL augmente lorsque l'erreur d'approximation diminue. Ceci n'est cependant pas toujours vérifié localement ; on remarque, par exemple figure IV.9, qu'il y a plus de triangles générés pour une approximation de 0,01 mm que pour une approximation de 0,005 mm. Ceci est dû à la réorganisation des triangles due au respect des règles de triangulation exposées au chapitre III paragraphe II.1.1.

On constate (figure IV.10), l'augmentation de l'erreur de position en fonction de l'erreur d'approximation STL.

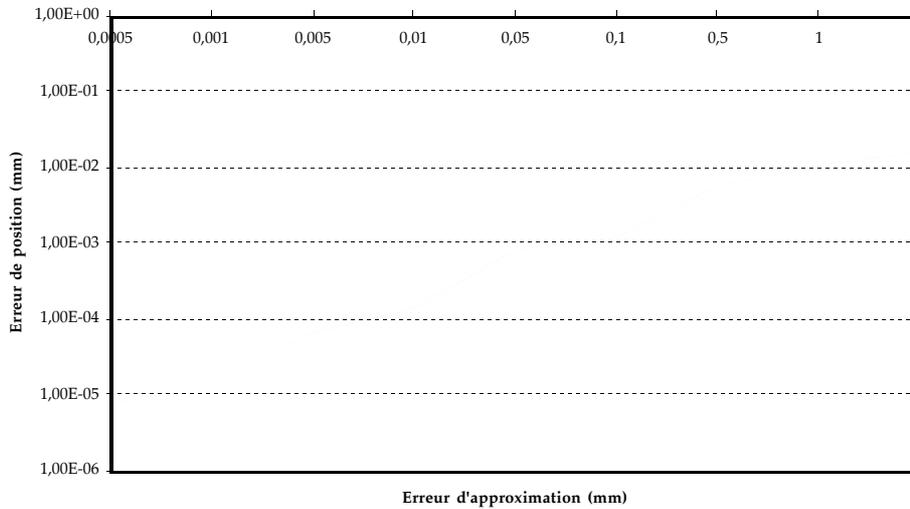


Figure IV.10 : Evolution de l'erreur de position en fonction de l'erreur d'approximation STL

Le choix du paramètre d'approximation et donc du nombre de triangles manipulés dans le modèle STL, représente un compromis entre exactitude de représentation et gain de temps de calcul.

En nous plaçant dans un cas de mise en correspondance classique, nous étudions l'influence de l'erreur d'interpolation sur le temps de calcul (figure IV.11) et vérifions que le temps de calcul diminue lorsque l'erreur d'interpolation diminue.

Données 3D	: Points idéaux de "Cylindre"
Modèle STL	: Erreur d'approximation variable
N_t	: 10
N_s	: 30
Translation initiale	: $\{10, 10, 10\}$
Rotation initiale	: 15° autour de $\{1, 1, 1\}$

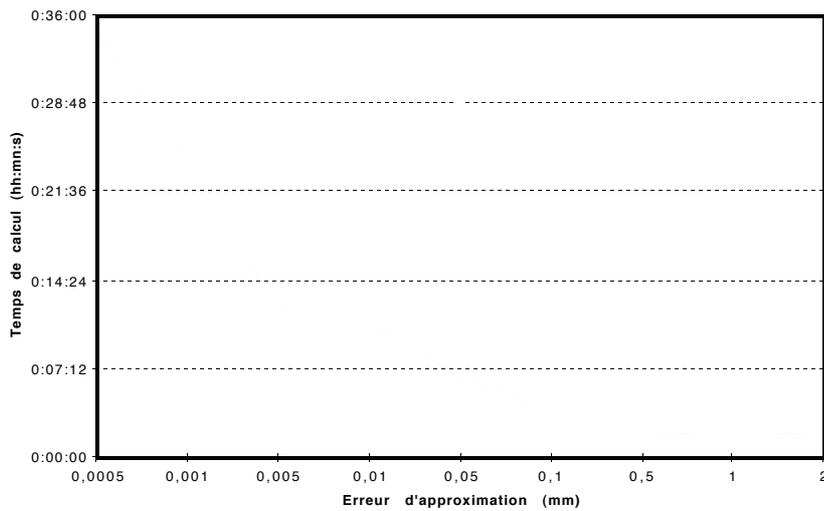


Figure IV.11 : Evolution du temps de calcul en fonction de l'erreur d'approximation STL (Cylindre).

II.2.2. Objet "Support"

Comme pour l'objet "Cylindre", nous évaluons l'erreur de modélisation pour l'objet Support.

Données 3D	: Points idéaux de "Support"
Modèle STL	: Erreur d'approximation variable
N_t	: 10
N_s	: 30
Translation initiale	: $\{0,0,0\}$
Rotation initiale	: 0°

Sans mise en correspondance, les points 3D et le modèle STL étant initialement confondus, nous évaluons l'influence de l'erreur de modélisation (tableaux IV.2, IV.3).

Erreur d'approximation (mm)	Nombre de triangles	Erreur de position (μm)
0,005	1636	0,3
0,01	1290	0,5
0,1	536	4,5

Tableau IV.2 : Nombre de triangles et erreur de position en fonction de l'erreur d'approximation

Données 3D	: Points idéaux de "Support"
Modèle STL	: Erreur d'approximation variable
N_t	: 10
N_s	: 200
Translation initiale	: $\{10, 10, 10\}$
Rotation initiale	: 5° autour de $\{1, 1, 1\}$

Erreur d'approximation (mm)	Temps de calcul (h:mn:s)
0,005	0:11:45
0,01	0:10:63
0,1	0:07:22

Tableau IV.3 : Temps de calcul en fonction de l'erreur d'interpolation (*Support*).

Nous avons évalué précédemment les erreurs de la méthode de mise en correspondance, nous devons donc choisir une erreur de modélisation bien inférieure. Volontairement, nous choisirons le modèle comportant le plus de triangles afin d'étudier le comportement de la méthode dans les meilleures conditions (erreurs de modélisation moindres), mais ceci au détriment des temps de calcul.

III. Comportement de la méthode en fonction de ses paramètres internes

Tel que nous l'avons vu lors de la présentation de la méthode de mise en correspondance, ses paramètres internes sont les suivants :

- N_s : Nombre de points extraits aléatoirement de l'ensemble de points initiaux et utilisés pour la mise en correspondance,
- N_t : Nombre d'essais réalisés dans l'algorithme ICP,
- Nombre de parallélépipèdes utilisés pour l'accélération de la méthode.

Certains de ces paramètres interviennent directement dans la précision du résultat fourni par la mise en correspondance (N_s , N_t), d'autres tels que le nombre de parallélépipèdes divisant l'espace STL n'interviennent que pour la diminution du temps de calcul.

Nous testons pour chacun de ces paramètres son influence sur la qualité de la mise en correspondance et son incidence sur le temps de calcul.

III.1. Paramètres N_s et N_t

L'efficacité de l'algorithme de mise en correspondance dépend en partie de la phase échantillonnage aléatoire, qui essaie de trouver une bonne estimation de la transformation rigide à partir de sous-ensembles de points extraits aléatoirement, en souhaitant lors des tirages sélectionner un minimum de points aberrants. Les deux paramètres qui interviennent alors sont : N_s le nombre de points extraits et N_t le nombre de sous-ensembles de points utilisés.

III.1.1. Etude théorique

Masuda présente [Masuda 95] comment évaluer la probabilité de sélectionner un "bon" échantillon : si on appelle ε le pourcentage de points aberrants, tels que ceux liés au bruit, aux occlusions, aux points qui apparaissent ou disparaissent de la scène, la probabilité de sélectionner un point non-aberrant est : $1 - \varepsilon$; la probabilité d'extraire un sous-ensemble de N_s points tous non-aberrants est : $(1 - \varepsilon)^{N_s}$.

La probabilité qu'un processus aléatoire sélectionne un sous-ensemble de points tous non-aberrants lors de N_t essais est : $p(\varepsilon, N_s, N_t) = 1 - (1 - (1 - \varepsilon)^{N_s})^{N_t}$. Cette probabilité augmente lorsque ε décroît vers zéro, lorsque N_s décroît ou N_t augmente, ce qui est vérifié dans les figures ci-dessous (figures IV.12a, IV.12b).

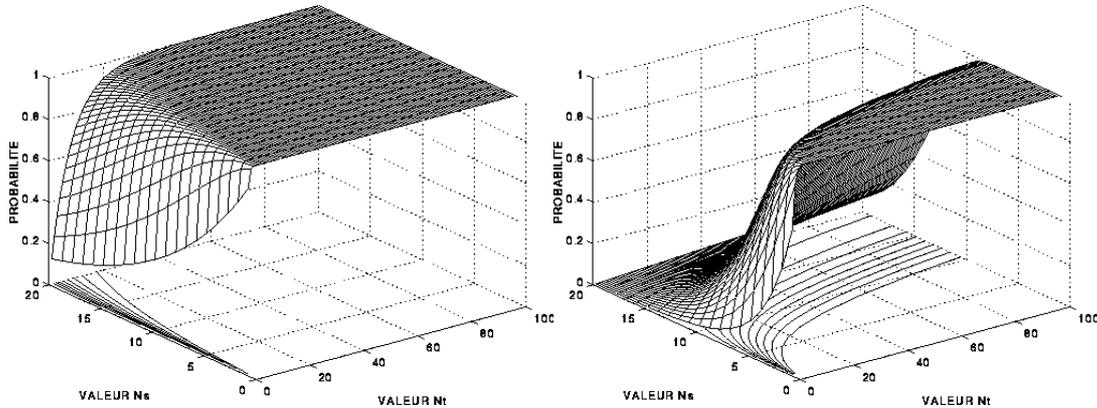


Figure IV.12a : Evolution de p en fonction de N_s et N_t pour $\varepsilon = 0,1$ (à gauche), et $\varepsilon = 0,3$ (à droite) (courbes théoriques).

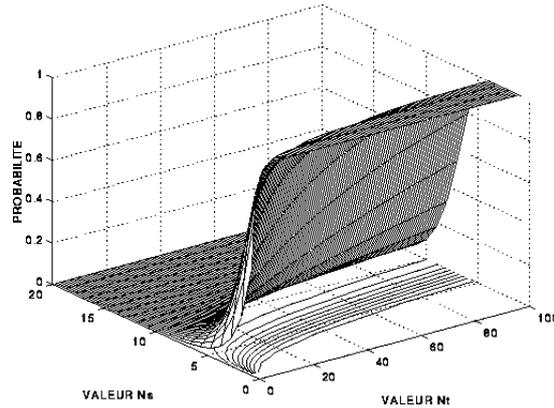


Figure IV.12b :Evolution de p en fonction de N_s et N_t pour $\varepsilon=0.5$ (Courbe théorique).

La probabilité p devient élevée lorsque N_s est petit, mais une faible valeur de N_s fait que l'estimation ICP devient moins fiable, car un échantillon de quelques points peut facilement être mis en correspondance avec de nombreuses parties du modèle CAO. Ceci implique qu'une petite valeur de N_s requiert habituellement de nombreux essais pour obtenir une bonne estimation. La probabilité que l'estimation ICP produise un résultat correct en utilisant N_s points ne peut être estimée, car elle dépend à la fois de la forme de l'objet et des propriétés de bruit dans l'acquisition de points 3D. Par ailleurs, si N_s est grand, N_t doit l'être également, car il devient difficile d'extraire aléatoirement un échantillon de points contenant uniquement des points non-aberrants.

Le tracé de $p(\varepsilon, N_s, N_t)$ (figure IV.12) confirme que lorsque ε augmente, il faut diminuer le nombre de points N_s et augmenter le nombre d'essais N_t . On déduit également de l'attitude logarithmique des courbes de niveaux $N_s=f(N_t)$, qu'il est intéressant pour diminuer les temps de calcul de diminuer le nombre de points N_s , ce qui diminue également fortement le nombre d'essais nécessaires N_t .

III.1.2. Vérification expérimentale

Objet Support

Afin de tester l'influence des valeurs de N_s et de N_t , nous avons effectué les mises en correspondance suivantes :

{	Données 3D	: Points idéaux de "Support"
	Modèle STL	: Erreur 0.005 mm
	N_t	: 5, 10, 15, 20, 30 ou 50
	N_s	: 5, 10, 30, 50, 100, 200, 500 ou 1000
	Translation initiale	: {10, 10, 10}
	Rotation initiale	: 5° autour de {1, 1, 1}

et tracé l'évolution des erreurs de position et des temps de calcul en fonction de N_s et N_t (figures IV.13, IV.14).

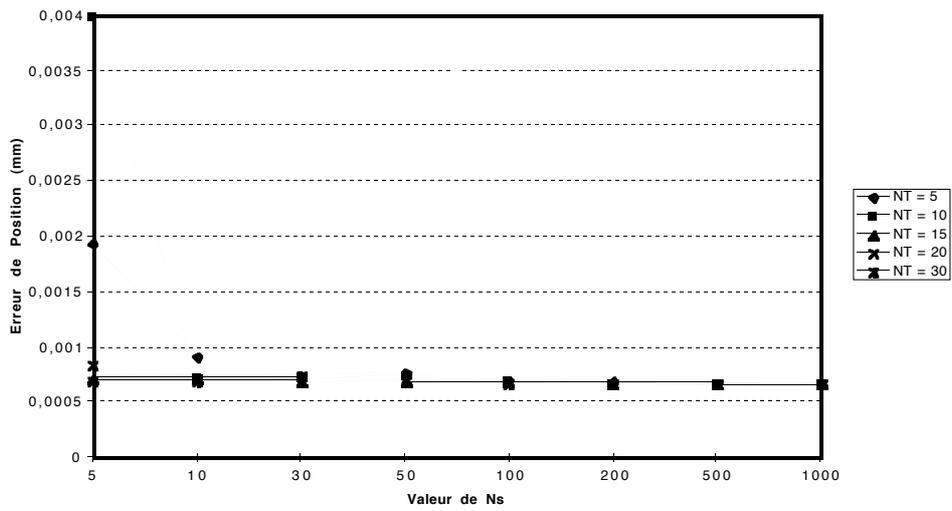


Figure IV.13 : Erreur de position (*Support*) en fonction de N_s et N_t .

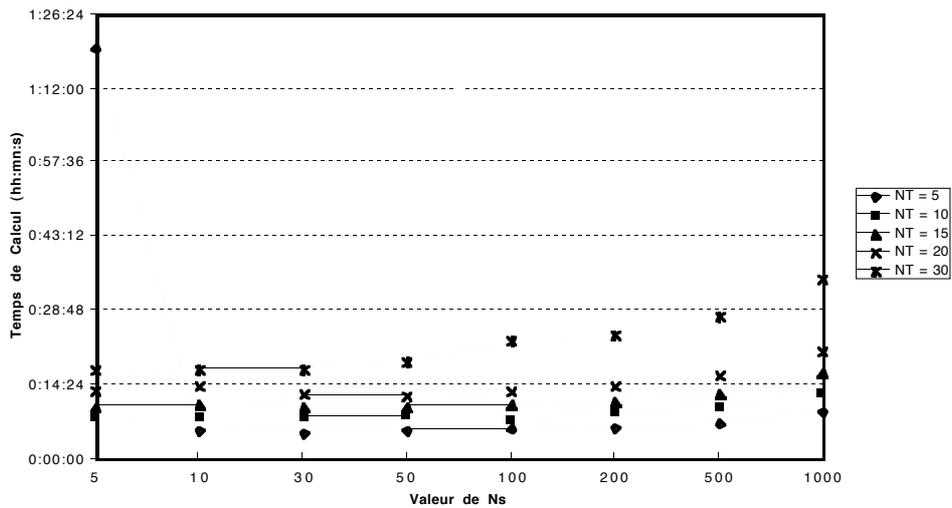


Figure IV.14 : Temps de calcul (*Support*) en fonction de N_s et N_t .

Cette expérience montre que pour $N_s=10$ et $N_t=10$, on obtient une erreur de position de $0,7 \mu\text{m}$, qui est l'erreur de la méthode pour cet objet.

Objet *Cylindre*

- Données 3D : Points idéaux de "Cylindre"
- Modèle STL : Erreur 0.0005 mm
- N_t : 5, 10, 15, 20, 30 ou 50
- N_s : 5, 10, 30, 50, 100, 200, 500 ou 1000
- Translation initiale : {10, 10, 10}
- Rotation initiale : 15° autour de {1, 1, 1}

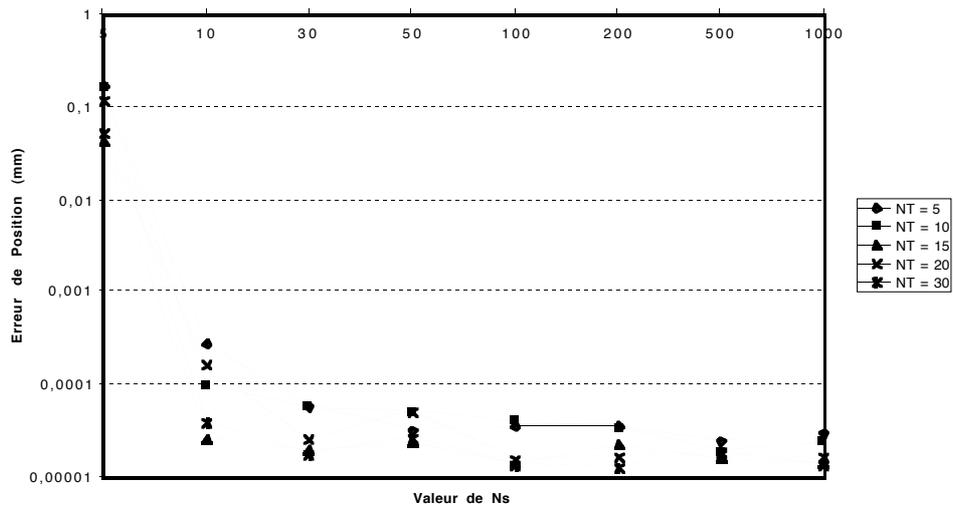


Figure IV.15 : Erreur de position (*Cylindre*) en fonction de N_s et N_t .

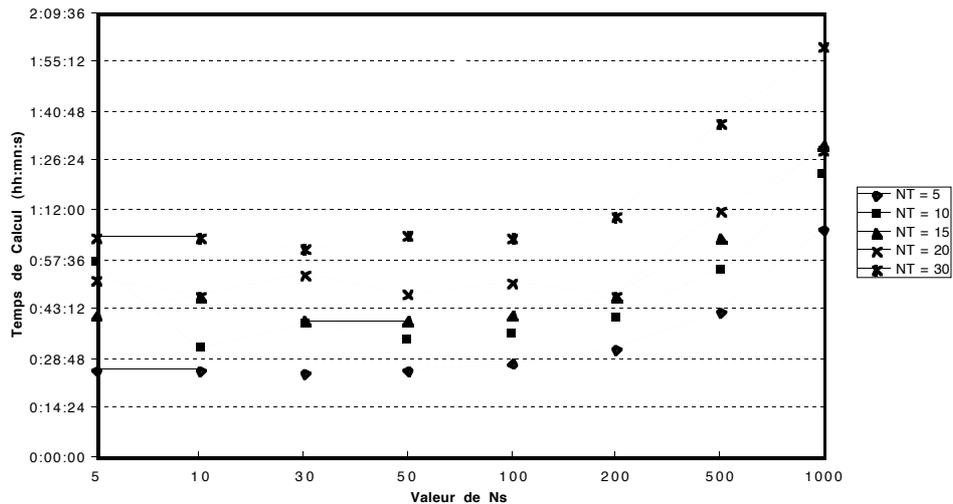


Figure IV.16 : Temps de calcul (*Cylindre*) en fonction de N_s et N_t .

Pour $N_s=30$ et $N_t=10$, on obtient une erreur de position inférieure à 0,1 μm , pour l'objet *Cylindre* (figures IV.15, IV.16).

D'une manière générale, nous constatons que l'erreur de position diminue lorsque N_t et N_s augmentent, ce qui confirme ce qui a été avancé d'un point de vue théorique au paragraphe III.1.1. Le choix de N_t égal à 10 et N_s supérieur à 30 s'avère suffisant, pour espérer obtenir une erreur de positionnement correcte. Par ailleurs, le temps de calcul augmente lorsque N_t et N_s augmentent, ce qui se comprend aisément puisque le nombre d'opérations à réaliser augmente proportionnellement à N_t et N_s . Prendre des valeurs de N_s et N_t plus élevées n'apporterait donc rien à la qualité du résultat et ne ferait qu'augmenter les temps de calcul.

III.2. Diminution du temps de calcul

Il a été présenté au chapitre II plusieurs méthodes récentes permettant de diminuer notablement le temps de mise en correspondance par une meilleure organisation du modèle géométrique. Une de ces méthodes a été implémentée, nous avons testé l'amélioration que celle-ci a apportée.

L'utilisation des voxels introduit un temps de calcul supplémentaire dû au partitionnement de l'espace en voxels (t_{part}), mais réduit le temps de la mise en correspondance (t_{MenC}).

Le temps de calcul global vaut maintenant :

$$t_{total} = t_{part} + t_{MenC}$$

Nous avons évalué le temps de partitionnement et le temps de mise en correspondance en fonction du nombre de parallélépipèdes, afin d'estimer pour deux objets le nombre de parallélépipèdes optimal. Nous avons représenté ces temps sur un graphique, en faisant apparaître t_{part} et t_{MenC} en gris et t_{total} en noir (figures IV.17, IV.18).

Objet *Cylindre*

Données 3D	: Points idéaux de "Cylindre"
Modèle STL	: Erreur 0.0005 mm
N_t	: 5
N_s	: 30
Translation initiale	: {10,10,10}
Rotation initiale	: 15° autour de {1,1,1}

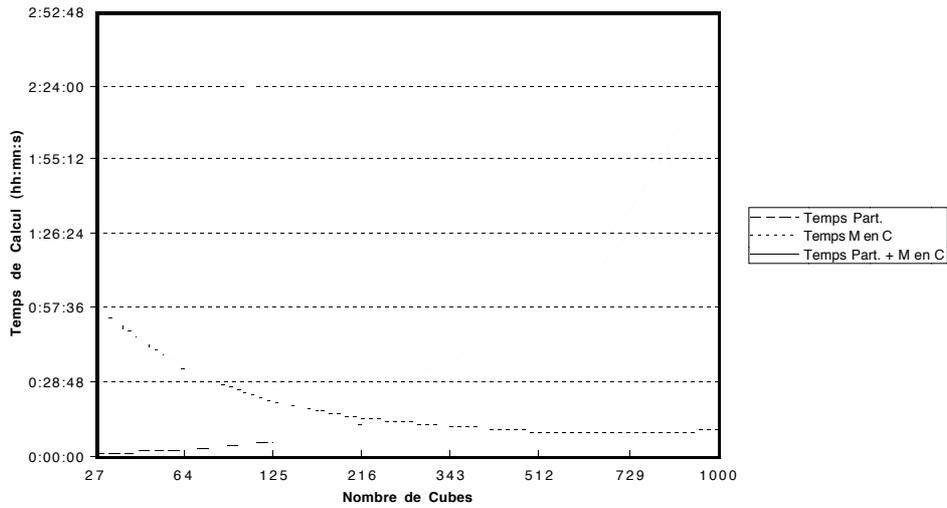


Figure IV.17 : Temps de calcul global en fonction du nombre de parallélépipèdes (*Cylindre*).

Support

}	Données 3D	: Points idéaux de "Support"
	Modèle STL	: Erreur 0.005 mm
	N_t	: 5
	N_s	: 30
	Translation initiale	: $\{10, 10, 10\}$
Rotation initiale	: 5° autour de $\{1, 1, 1\}$	

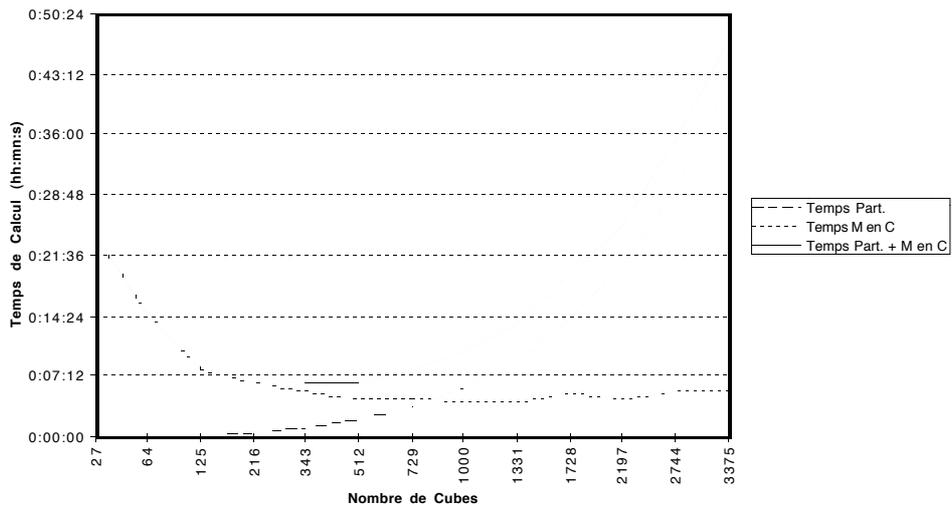


Figure IV.18 : Temps de calcul global en fonction du nombre de parallélépipèdes (*Support*).

Nous constatons que lorsque le nombre de parallélépipèdes devient plus important, nous obtenons une augmentation du temps de partitionnement due à une plus grande quantité de calculs à effectuer, et une diminution du temps de mise en correspondance due à la seule prise en compte des triangles les plus proches.

On constate un temps de calcul optimum pour 216 parallélépipèdes, soit 6 parallélépipèdes dans chaque direction, pour chacun des objets, divisant ainsi par trois le temps de calcul.

IV. Robustesse au bruit

Afin de tester la robustesse de notre algorithme de mise en correspondance, nous avons ajouté sur un pourcentage variable de points, un bruit normal dont l'amplitude maximale varie entre 0,0001 mm et 10 mm sur l'ensemble des points idéaux.

Données 3D	: Points idéaux de "Support"
Modèle STL	: Erreur 0.005 mm
N_t	: 10
N_s	: 30
Translation initiale	: $\{10, 10, 10\}$
Rotation initiale	: 10° autour de $\{1, 1, 1\}$

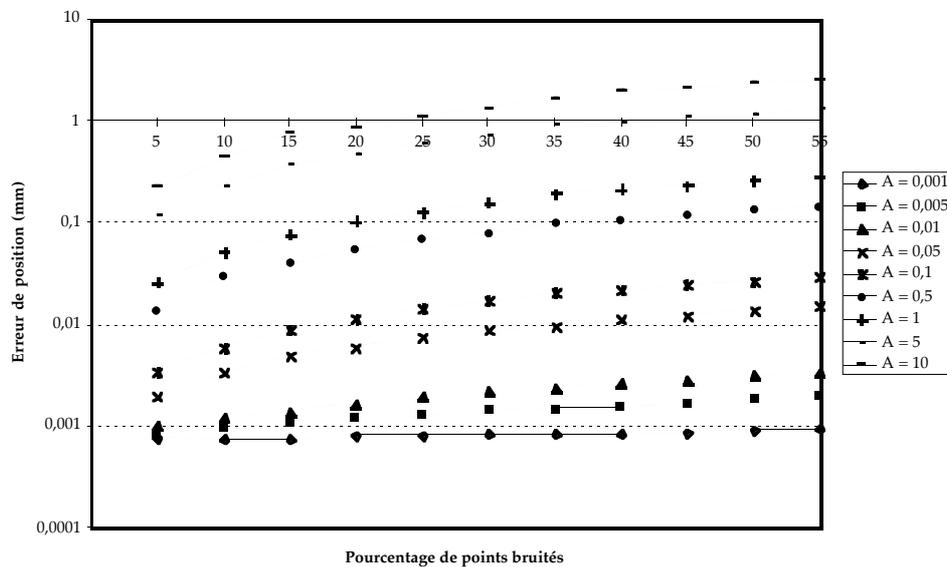


Figure IV.19: Erreur de position en fonction du pourcentage de points bruités et de l'amplitude du bruit ajouté (*Support*).

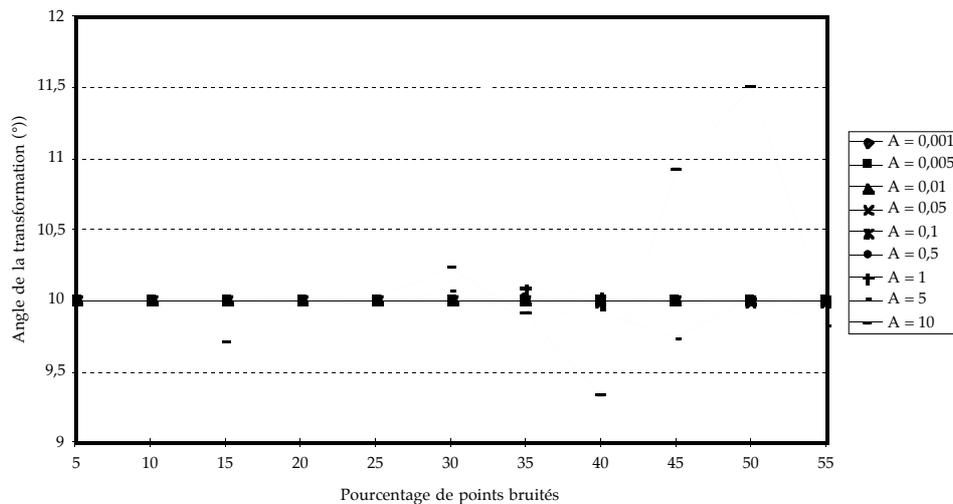


Figure IV.20 : Angle de la transformation en fonction du pourcentage de points bruités et de l'amplitude du bruit ajouté (Support).

Nous vérifions (figures IV.19, IV.20) que malgré le bruitage des points la méthode demeure robuste à moins de 50% de points parasites.

Conclusions

Les tests effectués sur la méthode ont mené aux conclusions suivantes :

- La méthode de mise en correspondance présentée est répétitive vis-à-vis des erreurs de positionnement avec une faible dispersion et ce quelque soit la position initiale relative des deux formes à mettre en correspondance. L'erreur de la méthode lorsque le nombre de triangles du modèle est élevé a été estimée à 0,035 μm en moyenne par point, donc négligeable par rapport aux applications traitées.
- Il est nécessaire d'utiliser un modèle STL avec un nombre suffisant de triangles afin que l'erreur de modélisation introduite reste négligeable par rapport aux erreurs de calcul tolérables pour nos applications. C'est l'erreur de modélisation qui va fixer les performances de la méthode de mise en correspondance.
- La méthode d'accélération de distance que nous proposons, c'est à dire la hiérarchisation de l'espace des triangles en voxels parallélépipédiques, s'avère efficace pour 216 parallélépipèdes c'est à dire 6 dans chacune des directions de modélisation. Nous avons juste prouvé l'efficacité de cette hiérarchisation, il est possible d'optimiser

le nombre de parallélépipèdes en fonction de la complexité de l'objet dans chacune des directions de modélisation.

- Nous avons établi que 10 itérations de l'algorithme ICP ($N_i=10$) sont suffisantes et que le nombre de points (N_s) extraits aléatoirement dépend de la complexité de l'objet et doit être compris entre 10 et 50.
- En bruitant un pourcentage connu de points avec un bruit d'amplitude connue, nous vérifions la robustesse de la méthode pour un pourcentage de points parasites inférieur à 50%.

Chapitre V

Inspection Automatique

Parmi les applications possibles de la méthode de mise en correspondance que nous proposons, figure l'inspection de pièces industrielles. En effet, disposant du modèle CAO de la pièce à inspecter, c'est-à-dire de son modèle idéal, ainsi que d'un ensemble de nombreux points 3D décrivant la pièce usinée, il nous est possible par mise en correspondance des deux formes de vérifier la conformité de la pièce usinée, selon un standard de tolérance.

Nous classons les différentes pièces qui peuvent être inspectées en deux catégories, en fonction de la complexité de leurs surfaces :

- Objets de forme simple : Objets comprenant des surfaces planes ou cylindriques (surfaces de degré maximum égal à 2), pour lesquels il existe une définition non ambiguë des cotes et des tolérances.
- Objets de forme complexe : Objets d'origine industrielle comprenant des surfaces gauches et peu de surfaces planes. Ceux-ci sont difficilement mesurables par les techniques traditionnelles (réglet, pied à coulisse, comparateur, ...), de par le manque de références et de méthodes de mesures.

La validité de l'opération d'inspection passe par une connaissance précise des erreurs induites par la mise en correspondance d'une part et de l'incertitude de mesure du capteur utilisé d'autre part, erreurs qui doivent impérativement être négligeables par rapport aux dimensions contrôlées lors de l'inspection.

Nous nous proposons donc dans un premier temps de présenter un tour d'horizon des méthodologies existantes pour représenter les cotes et tolérances des pièces complexes ou non-complexes, puis nous proposerons dans chacun des cas une méthode d'inspection utilisant les données 3D (capteur télémétrique ou machine à mesurer) et les modèles CAO dont nous disposons.

I. Cotes et tolérances

D'une manière générale, on peut définir l'inspection d'une pièce industrielle comme étant la vérification des dimensions et tolérances imposées lors de sa conception. Les dimensions déterminent les propriétés géométriques d'une pièce, en définissant les tailles, les formes, les caractéristiques, les positions, les orientations et distances entre caractéristiques nominales de la pièce. Comme l'imprécision inévitable des procédés de fabrication fait qu'une pièce ne peut jamais être réalisée rigoureusement conforme aux dimensions fixées par son concepteur, il est toléré que chaque dimension réalisée soit comprise entre deux dimensions limites, compatibles avec un fonctionnement correct de la pièce. La différence entre ces dimensions définit la tolérance. La définition des tolérances d'une pièce industrielle est aussi importante que sa cotation dans le sens où il n'assure pas seulement sa qualité de réalisation mais également son coût. Ainsi, une pièce est jugée correcte lorsque chacune de ses dimensions est comprise dans les tolérances admissibles.

Nous abordons maintenant les différentes façons de faire figurer la cotation et la définition des tolérances selon qu'une pièce est de forme complexe ou non.

I.1. Pièces de forme non-complexe

De façon standard, il existe deux types de tolérance : la tolérance "traditionnelle" $D \pm_{t_2}^{t_1}$, ou d , dimension réelle, doit vérifier $D - t_2 < d < D + t_1$, et les tolérances géométriques.

Les tolérances géométriques limitent les écarts admissibles de forme, d'orientation, de position ou de battement d'un élément. Certaines de ces tolérances sont intrinsèques, c'est-à-dire qu'elles ne dépendent que de la caractéristique qui est tolérancée (tolérance de forme,...). D'autres sont extrinsèques, elles sont relatives à d'autres caractéristiques et demandent la spécification de références.

Parmi ces tolérances géométriques, on peut distinguer :

- les tolérances de forme,
- les tolérances de position,
- les tolérances d'orientation,
- les tolérances de battement.

Parmi celles-ci, nous ne détaillerons que celles qui nous intéressent, la totalité de ces tolérances étant détaillées dans [Chevalier 95].

1.1.1. Tolérances de forme

Ces tolérances contrôlent la déviation de la forme d'une caractéristique à partir de sa forme nominale.

Les tolérances de forme sont :

- la rectitude,
- la planéité (figure V.1),

Une partie quelconque de la surface sur doit être comprise entre deux plans parallèles distants de h .

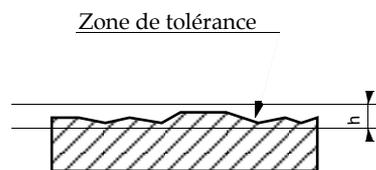


Figure V.1 : Tolérance de planéité.

- la circularité (figure V.2),

Le profil de chaque section droite doit être compris entre deux circonférences concentriques dont les rayons différent de h

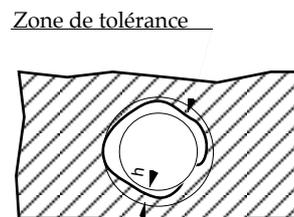


Figure V.2 : Tolérance de circularité.

- la cylindricité (figure V.3),

La surface doit être comprise entre deux cylindres coaxiaux dont les rayons différent de h .

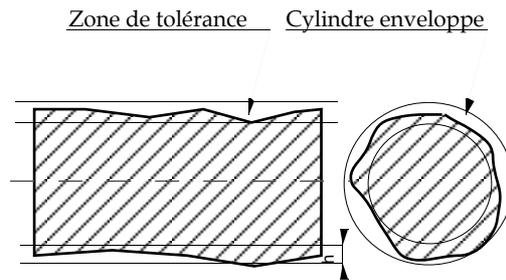


Figure V.3 : Tolérance de cylindricité.

- Forme d'une surface quelconque (figure V.4)

La surface tolérancée doit être comprise entre les deux surfaces qui enveloppent l'ensemble des sphères de tolérance centrées sur la surface nominale

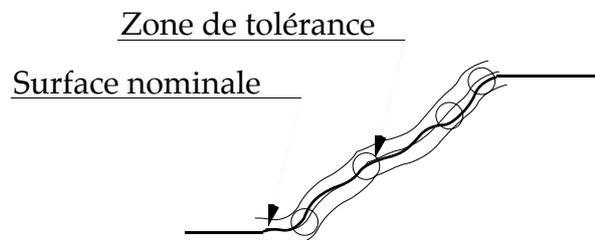


Figure V.4 : Tolérance de forme d'une surface quelconque.

1.1.2. Tolérances d'orientation

Ces tolérances contrôlent l'orientation d'une caractéristique par rapport à une autre.

- Perpendicularité
- Inclinaison
- Parallélisme (figure V.5)

La surface contrôlée doit être comprise entre deux plans parallèles à la surface choisie comme référence.

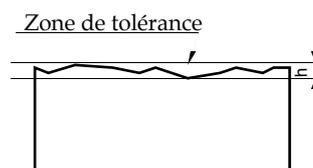


Figure V.5 : Tolérance de parallélisme.

1.1.3. Tolérances de position

Ces tolérances contrôlent la position relative de deux caractéristiques.

- Localisation,
- Coaxialité - Concentricité,
- Symétrie.

1.1.4. Tolérances de battement

Ces tolérances contrôlent la déviation d'une caractéristique cylindrique, lorsque celle-ci tourne autour de son axe de rotation.

- Battement simple axial,
- Battement simple radial,
- Battement total axial,
- Battement total radial,
- Battement total dans une direction donnée.

I.2. Pièces de forme complexe

Cardew-Hall dans [Cardew-Hall 93] présente une méthode pour représenter les tolérances et dimensions à partir d'un modèle surfacique, ayant la particularité d'être applicable à n'importe quelle forme, quelque soit sa complexité.

Cette méthode est basée sur trois formats de tolérance : une tolérance de position, une tolérance de forme et une tolérance de référence.

1.2.1. Matrice de dimension

Soit F_g , le repère de référence de l'objet modélisé. Chaque face i a son repère de référence F_i . Chacun de ces repères F_i a 6 degrés de liberté et peut être positionné par une matrice de transformation 4×4 T_i : $F_i = F_g \Sigma T_i$, T_i est obtenue par concaténation des matrices de rotation autour des axes X,Y,Z de F_i et d'une translation : $T_i = T_{i\alpha} T_{i\beta} T_{i\gamma} T_{i\delta}$ (figure V.6).

$$T_{i\delta} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \Delta x & \Delta y & \Delta z & 1 \end{pmatrix} \quad T_{i\alpha} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$T_{i\beta} = \begin{pmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad T_{i\gamma} = \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 & 0 \\ \sin \gamma & \cos \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

où α : Rotation autour de l'axe X Δx : Translation selon x
 β : Rotation autour de l'axe Y Δy : Translation selon y
 γ : Rotation autour de l'axe Z Δz : Translation selon z

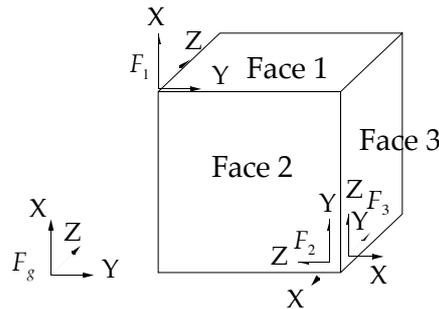


Figure V.6 : Positionnement des repères de référence.

1.2.2. Matrice de tolérance

La matrice de dimension contraint la position de chaque face par rapport au repère de référence. En relaxant ces contraintes suivant les six degrés de liberté et en les autorisant à varier dans certaines limites, on définit les tolérances sur la position de chaque repère de surface. Ces limites peuvent être représentées par une série de matrices de transformation 4×4 , qui sont les matrices de tolérance de positionnement. L'application de ces matrices sur chaque repère de surface donne les limites de mouvement de chaque surface.

Pour chaque degré de liberté, deux matrices sont définies ; soit douze matrices au total :

$$+T_{\delta x'} + T_{\delta y'} + T_{\delta z'} + T_{\delta \alpha'} + T_{\delta \beta'} + T_{\delta \gamma'} \quad \text{pour les tolérances maximales}$$

$$-T_{\delta x'} - T_{\delta y'} - T_{\delta z'} - T_{\delta \alpha'} - T_{\delta \beta'} - T_{\delta \gamma'} \quad \text{pour les tolérances minimales}$$

Chaque surface est définie relativement à son repère de surface. Soit P , un point de la surface, défini par les paramètres u et v . $P(u, v) = S(u, v)$ où $S(u, v)$ est l'équation de la surface.

La transformation de la surface nominale par une matrice de tolérance définit une zone de tolérance, dans laquelle la surface doit être comprise.

Chaque point sur la surface de tolérance créée par la matrice de tolérance est obtenu par :

$$P_{\max}(u, v) = S(u, v)\Sigma(+T_{\delta_i}) \text{ et } P_{\min}(u, v) = S(u, v)\Sigma(-T_{\delta_i})$$

où T_{δ_i} est la matrice de tolérance pour le i -ième degré de liberté, $i = x, y, z, \alpha, \beta, \gamma$. Ceci aboutit à douze surfaces limites qui décrivent le volume de tolérance suivant :

$$V_{tol} = \iint (P_{\max}(u, v) - P_{\min}(u, v)) du dv$$

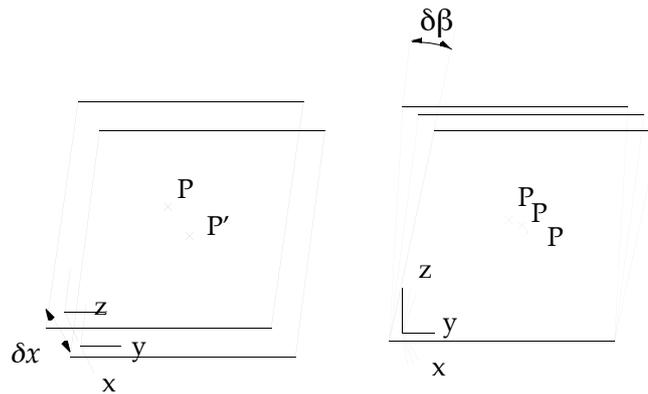


Figure V.7 : Effet d'une matrice de tolérance linéaire (gauche) et d'une matrice de tolérance rotationnelle (droite).

La figure V.7 illustre l'effet de l'application d'une transformation représentée par une matrice de tolérance linéaire ou par une matrice de tolérance rotationnelle sur un repère de référence.

1.2.3. Tolérance de forme des repères de référence

Jusqu'à présent, nous avons uniquement considéré des contraintes de position sur les repères de référence, mais il est possible que ces repères de référence présentent une déviation autorisée, indiquant une torsion ou un pliage de la surface.

- Pliage de la surface

La déviation d'un axe le long de sa longueur peut être limitée par un cylindre-limite de diamètre δ_{rx} (figure V.8). Cette déviation est représentée sur chacun des axes par le vecteur de tolérance de forme des repères de référence :

$$t_{\delta_r} = (\delta_{rx}, \delta_{ry}, \delta_{rz})$$

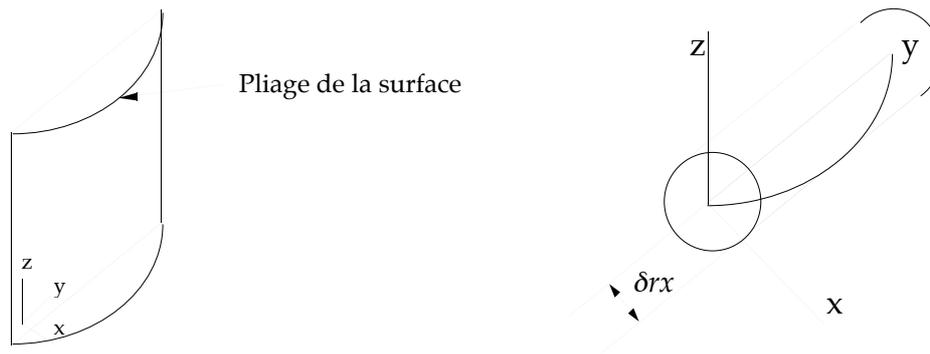


Figure V.8 : Tolérance de forme (pliage).

- Torsion de la surface

La limite de torsion d'une section relative à une autre est donnée en torsion par unité de longueur ; elle est représentée par le vecteur de tolérance de torsion des repères de référence (figure V.9) :

$$t_{\delta\theta} = (\delta\theta_x / l_x, \delta\theta_y / l_y, \delta\theta_z / l_z)$$

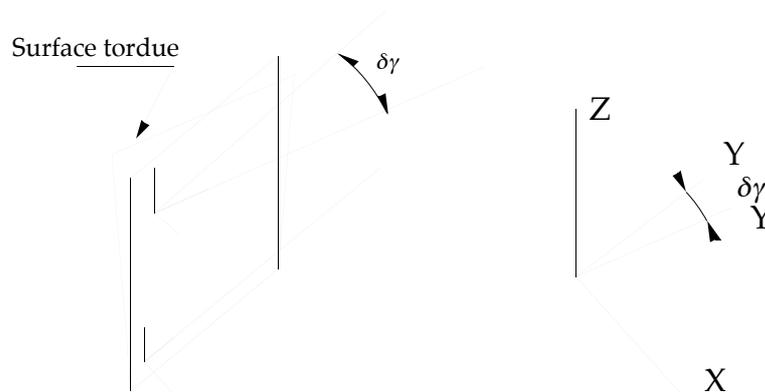


Figure V.9 : Tolérance de forme (torsion).

1.2.4. Tolérance de forme générale

Une tolérance de forme générale peut se formuler comme étant une fonction qui fait correspondre un point 3D d'une surface géométrique à un couple de paramètre (u, v) .

Les limites de position de la surface pour les tolérances de forme sont données par :

$$\vec{t}_{f_{\max}}(u, v) = \vec{S}(u, v) + \vec{f}_{to}(u, v) \quad \vec{t}_{f_{\min}}(u, v) = \vec{S}(u, v) - \vec{f}_{ti}(u, v)$$

- Les fonctions $\vec{f}_{ti}(u, v)$ et $\vec{f}_{to}(u, v)$ sont les fonctions de tolérance de forme intérieure et extérieure relatives à la surface en un point (u, v) .
- La fonction de tolérance $\vec{f}_{to}(u, v)$ est un vecteur normal à la surface au point (u, v) , où :

$$\vec{f}_{to}(u, v) = \vec{n}(u, v) \cdot R(u, v)$$

$\vec{n}(u, v)$: Vecteur-unitaire normal à la surface en (u, v) .

$R(u, v)$: Fonction scalaire représentant le décalage de tolérance en (u, v) .

I.3. Intégration des tolérances en CAO

Les années 60 ont vu l'avènement des systèmes de Conception Assistée par Ordinateur, qui facilitent énormément les opérations de création, de stockage, de modification et de transfert de dessins industriels pour la fabrication.

Lorsqu'un objet est conçu à l'aide d'un système de CAO, l'ensemble des dimensions de la pièce forment les spécifications d'une pièce idéale. Mais les variations du processus de fabrication font qu'il est impossible de réaliser une pièce suivant ses spécifications exactes. Ainsi, la conception d'une pièce est enrichie avec des informations définissant les limites acceptables sur les variations de taille ou de forme. Historiquement, les spécifications de tolérances sont faites de façon informelle, ce qui rend difficile leur interprétation automatique en particulier dans un processus d'inspection automatique. L'étude de spécification de tolérances formelles a reçu une attention particulière dans la littérature relative à la CAO et à la fabrication. La plupart des techniques courantes d'analyse de tolérances reposent sur le fait que le concepteur doit formuler le problème et résoudre les ambiguïtés résultant des groupes de caractéristiques ou des groupes de tolérances.

Jusqu'à peu, les logiciels de CAO utilisaient une représentation par surfaces pour représenter les objets. Plus récemment, sont apparus les systèmes de modélisation solide 3D, remplaçant progressivement la modélisation surfacique pour définir la géométrie nominale des objets. Ceux-ci permettent de disposer des données nécessaires permettant de calculer les propriétés de l'objet conçu, de pouvoir l'analyser à l'aide d'éléments finis, pour calculer ses interférences statiques et dynamiques, puis de créer et de vérifier les programmes utilisés pour sa fabrication. Par contre, cette modélisation solide ne permet pas de représenter de façon souhaitable les tolérances pour l'automatisation de la fabrication, de l'assemblage, de l'inspection ou de l'analyse de tolérance.

De nombreux chercheurs (dont [Fleming 88] [Gossart 88] [Juster 92] [Requicha 86] [Turner 87]) se sont penchés sur les problèmes de création, d'analyse des tolérances en CAO en représentation solide ou surfacique, car il n'existe pas de définition robuste pour la

définition des tolérances. Un état de l'art sur les développements dans le domaine de la modélisation et de la représentation des tolérances peut se trouver dans [Juster 92].

1.3.1. Espace vectoriel de variables modèles

Juster [Juster 92] donne une représentation des dimensions d'un objet consistant à le paramétrer, et obtient alors un espace de variables modèles. Par exemple, dans cette représentation, un cylindre est représenté par sa hauteur et son diamètre. L'inconvénient de cette représentation, est qu'elle ne limite pas les variations des paramètres, qui peuvent aboutir à la définition d'un objet valide et incomplet.

Turner [Turner 87] présente les résultats de ses recherches utilisant la modélisation solide, proposant des solutions au problème de définition de tolérances de façon automatique, en autorisant à chaque paramètre de varier dans un espace vectoriel de variables modèles. Cet espace vectoriel doit posséder les propriétés suivantes : indépendance linéaire et commutativité vis-à-vis de l'addition. Deux méthodes d'analyse numérique de tolérances sont particulièrement développées : Une méthode de programmation linéaire pour l'analyse de tolérances basée sur le "pire des cas" et l'analyse statistique des tolérances. Une terminologie a été introduite, utilisant des variables modèles pour représenter les variations du modèle géométrique, des variables de tolérance et des variables de conception pour les variations contraintes par le concepteur.

Gossart [Gossart 88] quant à lui a introduit une représentation combinant à la fois la représentation solide et la représentation par les frontières dans un graphe appelé graphe d'objet, où les dimensions sont représentées par un opérateur de positionnement relatif. Chaque changement de dimension aboutit à un changement de géométrie et de topologie. L'avantage de cette représentation est sa robustesse, qui élimine tous les problèmes de stabilité et de convergence liés aux méthodes numériques. Son inconvénient est qu'elle ne s'applique qu'aux objets polyédriques, dont les arêtes sont verticales ou horizontales.

L'inconvénient commun à ces méthodes est qu'elles ne peuvent pas modéliser les tolérances de forme.

1.3.2. Modèle Offset

La plupart des définitions données pour le dimensionnement et la définition de tolérances sont en fait des cas particuliers. Par exemple, la planéité d'une plaque est définie séparément de la cylindricité d'un trou, bien qu'en réalité elles contraignent la même propriété géométrique, limitant la déviation autorisée à partir de la forme idéale. Pour pallier l'inconvénient de ces multiples définitions, Requicha [Requicha 86] a introduit une nouvelle théorie pour les tolérances géométriques, basée sur le décalage des surfaces frontières d'un modèle solide nominal, nous appellerons cette opération "offset" de surfaces. Ces offsets

également appelés zones de tolérance sont utilisés pour contraindre les variations autorisées en taille, forme, orientation ou position.

Une spécification de tolérance consiste alors en :

- une représentation non-ambiguë d'un solide nominal S ,
- une représentation pour la décomposition de ∂S (frontières de S) en sous-ensembles F_i appelés caractéristiques de surface nominale,
- une liste de caractéristiques géométriques A_{ij} concernant les caractéristiques surfaciques nominales de S .

Ainsi la procédure d'inspection théorique d'une pièce physique, pour tester si celle-ci est en conformité est : Un objet P est défini comme respectant les spécifications de tolérance si et seulement si il existe une décomposition de P (∂P) en sous-ensembles G_i tels que les points suivants soient vérifiés :

- $\cup G_i = \partial P$
- Il y a une correspondance bijective entre G_i et les caractéristiques de surface nominale F_i de S
- Chaque G_i vérifie les caractéristiques géométriques A_{ij} de S .

Les caractéristiques géométriques A_{ij} sont définies comme des offsets. A chaque caractéristique est attribué un certain nombre de valeurs de tolérance pour définir les variations autorisées en taille, forme, position ou orientation. Requicha [requicha 86] établit comment une caractéristique G peut être comprise dans une tolérance de T : il doit être possible de décaler (offset) la caractéristique nominale correspondante F d'une distance positive a_p , et d'une distance négative a_n , tel que $a_p + |a_n| = T$. Ensuite, il doit être possible de transformer la zone de tolérance obtenue par une transformation rigide de sorte que G tout entier soit à l'intérieur de la zone de tolérance transformée.

II. Inspection Visuelle Automatique

Par définition, la tâche d'inspection consiste à déterminer si un produit dévie d'une liste de spécifications données. Ainsi, l'inspection implique la mesure de certaines caractéristiques telles que l'intégrité des assemblages, la finition de surface, les dimensions géométriques et les tolérances.

II.1. Définition de l'inspection visuelle automatique

Parmi les différents types d'inspections rencontrées dans le domaine industriel, nous appellerons inspection visuelle celle qui permet de contrôler sans manipuler les pièces, ce qui présente l'avantage de ne pas les détériorer et de ne pas nécessiter le ralentissement ou l'arrêt de la chaîne de fabrication.

L'inspection non-visuelle telle que l'utilisation de machine à mesurer 3D est très lente et demande une programmation individuelle détaillée pour chaque objet traité, d'où un manque certain de flexibilité. Le principe d'inspection à l'aide d'une machine à mesurer est le suivant : Par connaissance du positionnement exact de la pièce, la localisation de certains points ou surfaces peut être mesurée ; ce qui permet de connaître la position théorique d'autres caractéristiques (informations données par le programmeur), et en mesurant leur position réelle avec précision, on peut juger si un objet est correct ou non. L'avantage majeur de l'inspection utilisant une machine à mesurer est de procurer des informations 3D de grande précision (μm), jusqu'à présent inégalée par les méthodes sans contact.

L'inspection visuelle est l'une des phases du processus manufacturier à être la moins automatisée. Ceci est dû au fait que celle-ci demande un minimum de capacité de raisonnement, puisqu'il s'agit de rejeter ou accepter une pièce (inspection qualitative) ou de classer un type de défaut constaté (inspection quantitative). L'inspection visuelle automatisée comble en partie les défauts d'un évaluateur humain, dont la constance (aptitude à résister à la lassitude due à la répétitivité) et l'efficacité (efficacité maximum de 80% dans des conditions bien définies) ne sont pas optimales. Par ailleurs, certains défauts ne sont pas détectables par l'oeil humain et certains environnements sont hostiles à l'intervention humaine. L'inspection visuelle automatique coûte globalement moins cher, est plus rapide, procure une inspection de bonne qualité sans contact, donc sans endommagement de la production.

La faisabilité de l'inspection automatique visuelle réside dans une réalisation en temps réel, c'est-à-dire que cette opération ne doit pas constituer une limitation de temps dans le déroulement du processus manufacturier global. Cette faisabilité est souvent liée à des conditions d'inspection favorables telles qu'un bon prépositionnement de la pièce à inspecter ou de bonnes conditions d'éclairage.

Les techniques d'inspection automatique visuelles peuvent se classer en deux catégories :

- Mise en correspondance de parties d'une image modèle (image de synthèse ou image stockée) avec l'image fournie par le capteur. Applications : Inspection de circuits imprimés, inspection de pare-chocs, ...

- Extraction des caractéristiques de l'objet scanné et leur comparaison à une description ou à une liste de règles qui décrivent le modèle idéal. Si toutes les règles sont satisfaites, alors l'objet est considéré sans défaut.

Les Caractéristiques utilisées sont : Aire de surfaces, périmètres, ratio de périmètres sur aire, aire de trous, rayon minimum, maximum, L'avantage de cette méthode très utilisée est qu'il n'est pas nécessaire de maintenir une base de données. Par contre, elle n'est pas facilement adaptable à des changements de conception, et l'extraction de règles n'est pas toujours commode.

Le schéma classique de l'inspection automatique est le suivant (figure V.10) :

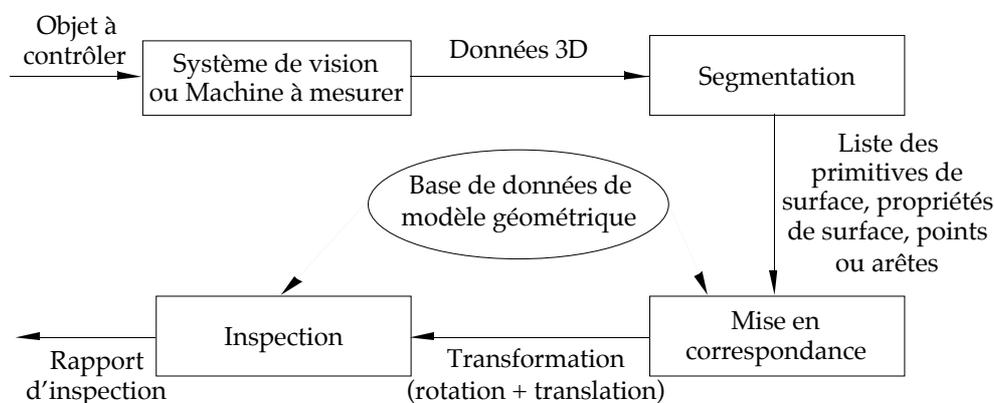


Figure V.10 : Schéma classique d'inspection automatique visuelle.

L'utilisation accrue des modèles CAO, le développement de nouveaux capteurs sans contact, l'augmentation des puissances de calcul à des coûts intéressants, ont favorisé le développement de nombreuses techniques d'inspection automatique.

II.2. Types d'images utilisées en inspection visuelle

Parmi les différents types d'informations utilisés en inspection visuelle, provenant de capteurs 2D ou 3D, nous allons présenter ceux qui sont les plus couramment rencontrés. Nous n'aborderons pas l'imagerie par rayonnement ionisant.

II.2.1. Images binaires

Historiquement, la plupart des systèmes commerciaux de vision utilisent des images binaires pour l'inspection automatique. Beaucoup de ces images sont de résolution limitée et permettent des vérifications relativement simples, telles que la présence d'une pièce. Les données binaires sont suffisantes pour la plupart des objets industriels, qui peuvent être simplement représentés par leur silhouette. Les objets plats, sans caractéristique surfacique

par exemple, se prêtent très bien à ce type d'inspection. Par ailleurs, il est économique d'utiliser des images binaires, car elles peuvent être acquises par des capteurs peu coûteux en conjonction avec des éclairages simples, tels que l'éclairage diascopique. En plus, l'utilisation d'images binaires réduit la quantité de données à traiter, ce qui permet d'aboutir à des coûts faibles et des vitesses de réalisation élevées.

II.2.2. Images à niveaux de gris

Beaucoup de systèmes utilisant des données à niveaux de gris ont été présentés dans la littérature, mais peu d'entre eux sont capables de réaliser une inspection visuelle dans un environnement industriel, souvent dans des conditions de poussières et d'éclairage défavorables. La plupart de ces systèmes d'inspection ont des capacités limitées et utilisent la soustraction d'images ou des méthodes d'histogrammes localisés pour la détection de défauts.

II.2.3. Images couleurs

La couleur est un facteur particulièrement important pour l'inspection de tissus ou de nourriture, où les changements d'ombrage ou de couleur indiquent un défaut. L'opérateur humain n'est pas efficace pour ce type d'inspection, car il n'a pas une bonne mémoire des couleurs, c'est pourquoi dans ce cas l'inspection automatique est plus appropriée.

II.2.4. Images de profondeur

Pour beaucoup d'applications, l'inspection utilisant des images binaires, à niveaux de gris ou de couleur n'est pas pratique. Par exemple, l'inspection d'aubes de turbine est généralement impossible en utilisant des informations d'intensité, parce que les informations de silhouette sont pauvres et que les informations d'intensité peuvent contenir des variations de lumière réfléchi trop importantes. L'inspection dimensionnelle de pièces qui présentent des variations de hauteur trop grande, des déformations ou beaucoup de reliefs demandent une mesure précise de profondeur. L'avantage majeur des capteurs de profondeur est donc de fournir directement des informations décrivant explicitement les surfaces. Par ailleurs, la mesure de profondeur n'est pas sensible à l'éclairage ambiant et les objets peuvent être facilement extraits du fond de la scène.

II.3. Utilisation de données CAO pour l'inspection

L'intérêt d'utiliser des modèles CAO pour l'inspection provient du fait que ceux-ci constituent une description exacte de l'objet, bien définie pour l'inspection prête à être utilisée. En effet, ceux-ci contiennent à la fois une description mathématique de la forme de l'objet, incluant une paramétrisation explicite de la forme des surfaces, et un codage explicite

des relations inter-surfaces. Un autre avantage des modèles CAO est leur grande flexibilité, qui permet par exemple d'ajouter de nouveaux objets dans le système d'inspection. Les modèles CAO sont également capables de modéliser les défauts les plus courants ; ils permettent ainsi au système d'inspection de reconnaître ceux-ci et de les localiser. On peut également rajouter des informations telles que le type de matériaux utilisé, sa couleur, la qualité souhaitée pour la finition des surfaces, constituant de la sorte très intéressante pour l'inspection.

Il existe cependant quelques barrières à l'utilisation des modèles CAO pour l'inspection visuelle automatique [Noble 95] :

- Le processus de formation des images est très complexe et habituellement non-modélisé par le modèle CAO. Par exemple, les images peuvent contenir des réflexions spéculaires, compliquant les tâches de reconnaissance ou d'inspection.
- Le calcul des surfaces "offsets" des surfaces complexes est plutôt délicat, car il requiert la solution de systèmes d'équations polynomiales de degrés élevés. Pour certaines surfaces, il n'existe d'ailleurs pas de méthode connue pour calculer leur offset exact.
- Bien que les informations de tolérance peuvent être incluses dans le modèle CAO, il est difficile de construire un modèle solide d'usage général, combinant les modèles mathématiques des dimensions et des tolérances.
- Bien souvent la pièce usinée est différente de la pièce originale. Ceci est dû aux ajustements effectués sur le processus de fabrication pendant le développement. Idéalement, le modèle CAO devrait être mis à jour pour rendre compte de ces modifications, mais généralement il n'y a pas de retour d'informations de la fabrication vers la conception.
- Les modèles CAO ne sont pas toujours disponibles, car même s'ils existent ils ne sont pas toujours mis à jour.
- Une pièce peut différer de son modèle CAO et être fonctionnellement acceptable. En pratique, ceci signifie que le rejet basé sur l'incapacité à vérifier les spécifications de conception peut amener à l'élimination de pièces tout à fait acceptables.
- Il n'est pas toujours possible mesurer certaines dimensions telles qu'elles sont spécifiées en CAO, par exemple lorsqu'un plan de référence n'existe pas physiquement, ou alors lorsque des dimensions ne sont pas physiquement accessibles par des outils classiques.

- La conception des objets diffère selon la convention de dimensionnement qui a été utilisée, ce qui rend difficile la généralisation des algorithmes pour l'analyse dimensionnelle.
- Les standards de dimensionnement et de définition de tolérances concernent seulement les incertitudes géométriques. Il n'est pas évident que ces standards soient adéquats pour la métrologie visuelle, où à la fois les incertitudes géométriques et les erreurs de mesure doivent être prises en compte.

II.4. Exemples d'inspection automatique visuelle

En annexe D, nous présentons quelques méthodes récentes d'inspection automatique visuelle de pièces comportant des surfaces complexes :

- Inspection de pièces moulées à partir de leurs modèles CAO et d'une image de profondeur,
- Inspection de pièces mécaniques à partir d'un modèle géométrique et d'images de profondeur,
- Inspection de pièces mécaniques définies en CAO avec une image 2D,
- Vérification de cylindricité avec une machine à mesurer,
- Inspection à partir d'un modèle CAO et d'une machine à mesurer.

Ces méthodes utilisent en référence un modèle géométrique de l'objet industriel, souvent défini en CAO, comparé avec des données 3D fournies soit par un capteur télémétrique, soit par une machine à mesurer ou une seule image 2D.

Ces différents exemples montrent que pour l'inspection de pièces industrielles, des données 3D provenant aussi bien d'une machine à mesurer que d'un capteur de profondeur peuvent être utilisées. Ces données peuvent être segmentées pour favoriser la mise en correspondance, mais cette segmentation impose une connaissance a priori de la nature des surfaces utilisées, généralement simples (plans, quadriques). Lorsque les points 3D et le modèle géométrique sont mis en correspondance, la déviation des points par rapport au modèle géométrique est calculée et comparée à la tolérance de forme, ce qui conduit à discriminer les points en tolérance de ceux qui ne le sont pas.

III. Méthode d'inspection d'objets de forme complexe

Nous distinguons deux niveaux d'inspection :

- L'inspection globale : celle-ci vérifie si chacune des surfaces est bien positionnée, bien orientée, et que sa forme est correcte.
- L'inspection locale : Chaque surface n'est plus considérée dans sa globalité mais comme une entité déformable pouvant comporter des variations locales.

Les méthodes d'inspection d'objets complexes existantes (voir Annexe D) se contentent, après mise en correspondance des données 3D avec le modèle géométrique de l'objet inspecté, de vérifier que pour chaque point la distance point/modèle géométrique est bien inférieure à la tolérance de forme souhaitée (tolérance de forme globale).

Notre méthode se distingue une inspection complète de l'objet qui a l'avantage d'être à la fois globale et locale, à partir de la mise en correspondance points 3D/modèle géométrique. La tâche d'inspection est d'abord préparée par une segmentation de l'ensemble de points 3D mis en correspondance, en plusieurs ensembles de points 3D, chacun d'eux étant associé à une même surface (surface la plus proche) ; ces N ensembles points/surface sont appelés E_i . Un point peut se trouver à égale distance de plusieurs surfaces (point proche d'un coin ou d'une arête) et appartenir à plusieurs de ces sous-ensembles. L'inspection de chaque surface va être menée à l'aide de ses points 3D proches. Lorsque toutes les surfaces ont été traitées, un rapport d'inspection constate la localisation des erreurs et leur caractérisation. Une décision peut alors être prise à partir de ce rapport (figure V.11).

La tâche d'inspection s'effectue point par point ; soit un point P proche d'une surface S , dès que le point ne respecte pas l'une des étapes de vérification, il est considéré comme défectueux.

Voici le déroulement de l'étape d'inspection :

- Vérification des tolérances globales
 - Tolérances de positionnement et d'orientation de chaque surface,
 - Tolérances de forme globale de chaque surface.
- Vérification des tolérances locales
 - Tolérance sur la normale,
 - Tolérance des courbures.

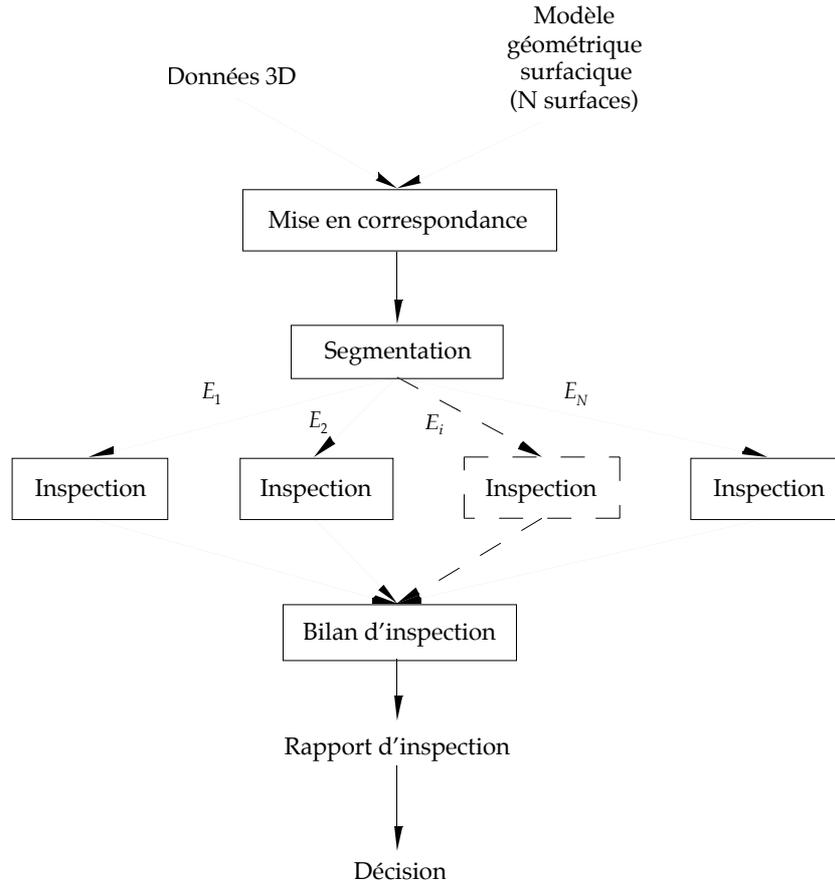


Figure V.11 : Stratégie d'inspection complète d'un objet.

III.1. Tolérances globales

La vérification des tolérances globales permet de juger si la surface dans sa globalité, vérifie les tolérances de positionnement, orientation, et forme.

III.1.1. Tolérances de positionnement et d'orientation de chaque surface

Pour contrôler ces tolérances, nous allons appliquer la représentation des tolérances de positionnement et orientation établie par Cardew-Hall [Cardew-Hall 93] et présentée paragraphe I.2.

Nous utilisons les douze matrices de tolérance sur le positionnement et l'orientation minimale et maximale des repères de référence de chaque surface, qui sont notées :

$$+T_{\delta x}, +T_{\delta y}, +T_{\delta z}, +T_{\delta \alpha}, +T_{\delta \beta}, +T_{\delta \gamma} \quad \text{pour les tolérances maximales}$$

$$-T_{\delta x}, -T_{\delta y}, -T_{\delta z}, -T_{\delta \alpha}, -T_{\delta \beta}, -T_{\delta \gamma} \quad \text{pour les tolérances minimales}$$

Le volume de tolérance défini par les douze surfaces de tolérance est :

$$V_{tol} = \iint (P_{\max}(u, v) - P_{\min}(u, v)) du dv$$

où

$$P_{\max}(u, v) = S(u, v) \Sigma(+T_{\delta_i}) \quad \text{et} \quad P_{\min}(u, v) = S(u, v) \Sigma(-T_{\delta_i})$$

Pour chaque point, il est nécessaire de vérifier, que celui-ci appartient bien au volume de tolérance. Ceci consiste pour chacune des six tolérances, à contrôler si le point se situe entre la surface de tolérance minimum et la surface de tolérance maximum.

Par exemple, soit S_N , la surface nominale, $S_{+\delta_x}$ et $S_{-\delta_x}$, les deux surfaces de tolérance, dans la direction des x (figure V.12). On doit avoir : $PP^- . PP^+ \leq 0$.

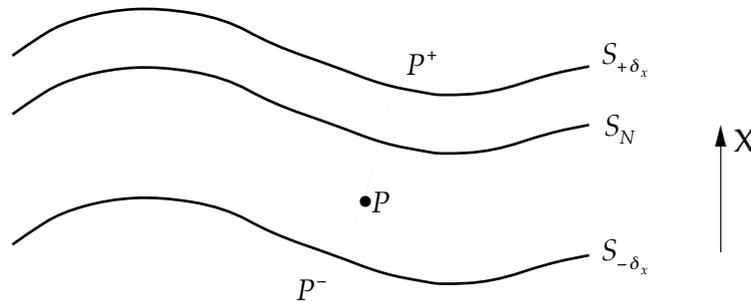


Figure V.12: Vérification des tolérances des repères de référence selon x .

III.1.2. Tolérances de forme globale de chaque surface

Une surface réelle $S_{réelle}$ vérifie la tolérance de forme d'une surface théorique $S_{théorique}$, si elle est entièrement comprise entre les deux surfaces de tolérances S_- et S_+ , qui sont définies de la façon suivante :

$$\begin{cases} S_+(u, v) = S_{théorique}(u, v) + \vec{n}(u, v) \cdot \varepsilon_+ \\ S_-(u, v) = S_{théorique}(u, v) - \vec{n}(u, v) \cdot \varepsilon_- \end{cases}$$

ε_+	Tolérance de forme maximum
ε_-	Tolérance de forme minimum
$\vec{n}(u, v)$	Vecteur unitaire normal à la surface théorique en u, v

Cette définition peut s'illustrer de la façon suivante (figure V.13) :

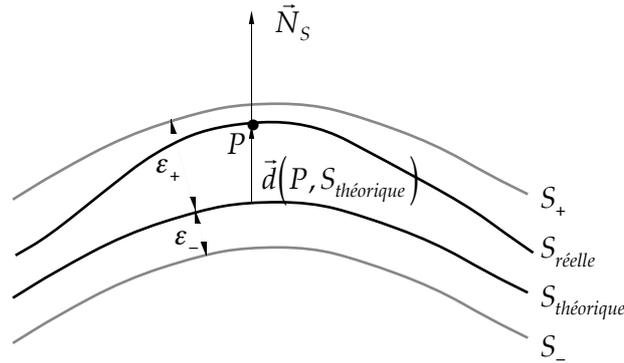


Figure V.13 : Tolérance de forme globale.

Un point P est considéré correct si une des deux conditions suivantes est vérifiée :

$$\|\vec{d}(P, S_{théorique})\| < \epsilon_+ \quad \text{si} \quad \vec{N}_S \cdot \vec{d}(P, S_{théorique}) > 0$$

$$\|\vec{d}(P, S_{théorique})\| < \epsilon_- \quad \text{si} \quad \vec{N}_S \cdot \vec{d}(P, S_{théorique}) < 0$$

où \vec{N}_S est la normale à la surface.

III.2. Tolérances de forme locales

La vérification des tolérances locales se fait par comparaison des propriétés locales (normale et courbures) du point P avec les propriétés locales du point de S le plus proche de P que nous appelons Q , de paramètres u et v sur la surface S . Le modèle géométrique de la surface S (définition littérale) nous permet de calculer exactement les propriétés locales en Q . Par contre, pour estimer les propriétés locales en P , il est nécessaire de réaliser une estimation de l'allure de la surface en ce point, obtenue en interpolant localement N_V points du voisinage de P à l'aide d'une surface polynomiale. Afin que les dérivées premières et secondes soient définies pour le calcul des courbures, une surface du deuxième ordre s'avère nécessaire et suffisante.

Nous noterons (figure V.14) :

\vec{N}_S : Normale à la surface S en Q ,

\vec{N}_P : Normale à la surface estimée S_p en P ,

K_S, H_S : Courbures gaussienne et moyenne de la surface S en Q ,

K_P, H_P : Courbures gaussienne et moyenne de la surface estimée S_p en P .

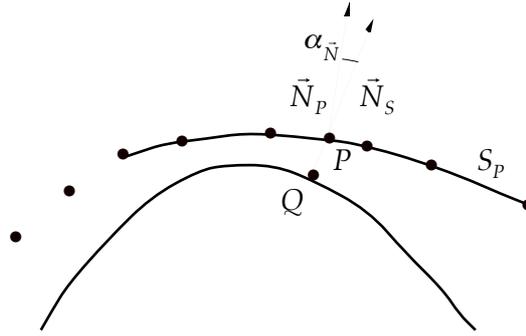


Figure V.14 : Surface nominale et surface interpolant les points 3D.

III.2.1. Calcul des paramètres de la surface

Nous allons calculer les propriétés géométriques de la surface en utilisant une méthode proposée par P. Boulanger [Boulanger 94b]. Cette méthode de calcul des paramètres géométriques est invariante par rapport à la paramétrisation de la surface et à l'orientation de celle-ci dans l'espace, ainsi qu'à la présence de discontinuités dans le voisinage de calcul. La méthode interpole les points du voisinage de P (\vec{r}_i) en utilisant un polynôme paramétrique du second ordre.

Ce polynôme s'écrit : $\vec{\eta}(u, v) = \sum_{i=0}^2 \sum_{j=0}^2 \vec{a}_{ij} u^i v^j$ où les \vec{a}_{ij} sont les coefficients de chaque composante de $\vec{\eta}(u, v)$ et sont égaux à zéro lorsque $i + j > 2$.

Trouver le polynôme caractéristique du second ordre, interpolant au mieux les points de du voisinage de P , consiste à trouver les paramètres \vec{a}_{ij} qui minimisent la différence entre le modèle et les données, c'est-à-dire : $\sum_{u,v} \|\vec{r}(u, v) - \vec{\eta}(u, v)\|^2$.

Lorsqu'on utilise la notation matricielle $\vec{\eta}(u, v) = M^t A$, on a :

$$\vec{\eta}(u, v) = \begin{bmatrix} 1 & u & v & u^2 & uv & v^2 \end{bmatrix} \begin{bmatrix} \vec{a}_{00} \\ \vec{a}_{10} \\ \vec{a}_{01} \\ \vec{a}_{20} \\ \vec{a}_{11} \\ \vec{a}_{02} \end{bmatrix}$$

où M correspond aux monômes de base et A au vecteur de coefficients.

La solution à ce problème est donnée par

$$\frac{\partial}{\partial \vec{a}_{ij}} \sum_{u,v} (\vec{r}(u, v) - M^t A)^2 = \vec{0}$$

d'où :

$$2 \sum_{u,v} (\vec{r}(u,v) - M^t A) u^i v^j = \vec{0} \quad \text{et} \quad A = \left(\sum_{u,v} (MM^t) \right)^{-1} \sum_{u,v} \vec{r}(u,v) M$$

La variable A peut se mettre sous la forme : $A = C^{-1}B$

$$\text{où } C = \begin{bmatrix} 1 & u & v & u^2 & uv & v^2 \\ u & u^2 & uv & u^3 & u^2v & uv^2 \\ v & uv & v^2 & u^2v & uv^2 & v^3 \\ u^2 & u^3 & u^2v & u^4 & u^3v & u^2v^2 \\ uv & u^2v & uv^2 & u^3v & u^2v^2 & uv^3 \\ v^2 & uv^2 & v^3 & u^2v^2 & uv^3 & v^4 \end{bmatrix}$$

Si on décompose l'équation vectorielle de A suivant les trois directions x , y et z , on a :

$$\begin{cases} A_x = \left(\sum_{u,v} (MM^t) \right)^{-1} \sum_{u,v} r_x(u,v) M \\ A_y = \left(\sum_{u,v} (MM^t) \right)^{-1} \sum_{u,v} r_y(u,v) M \\ A_z = \left(\sum_{u,v} (MM^t) \right)^{-1} \sum_{u,v} r_z(u,v) M \end{cases} \quad \text{ou} \quad \begin{cases} A_x = C^{-1}B_x \\ A_y = C^{-1}B_y \\ A_z = C^{-1}B_z \end{cases}$$

Nous déterminons A en utilisant les méthodes numériques "Décomposition en valeurs singulières" et "Résolution d'un système linéaire présentant plus d'équations que d'inconnues" exposées dans [Press 92].

III.2.2. Calcul des propriétés géométriques

Les courbures gaussiennes K et moyennes H sont deux invariants algébriques naturels, qui présentent des propriétés intéressantes :

- invariance par rapport au point de vue,
- invariance par rapport au type de paramétrisation,
- détermination unique de la forme d'une surface échantillonnée.

Soit une surface $\vec{r}(u, v)$, les courbures gaussienne et moyenne en un point de paramètres (u, v) peuvent être calculées par les équations suivantes [Boulanger 94a] :

$$K(u, v) = \frac{[\vec{r}_{uu}\vec{r}_u\vec{r}_v][\vec{r}_{vv}\vec{r}_u\vec{r}_v] - [\vec{r}_{uv}\vec{r}_u\vec{r}_v]^2}{|\vec{r}_u \times \vec{r}_v|^4}$$

$$H(u, v) = \frac{(\vec{r}_v \cdot \vec{r}_v)[\vec{r}_{uu}\vec{r}_u\vec{r}_v] + (\vec{r}_u \cdot \vec{r}_u)[\vec{r}_{vv}\vec{r}_u\vec{r}_v] - 2(\vec{r}_u \cdot \vec{r}_v)[\vec{r}_{uv}\vec{r}_u\vec{r}_v]}{2|\vec{r}_u \times \vec{r}_v|^3}$$

où $[\vec{a}, \vec{b}, \vec{c}] = \vec{a} \cdot (\vec{b} \times \vec{c})$

$$\text{et } \vec{r}_u = \frac{\partial \vec{r}}{\partial u}, \quad \vec{r}_v = \frac{\partial \vec{r}}{\partial v}, \quad \vec{r}_{uu} = \frac{\partial^2 \vec{r}}{\partial u^2}, \quad \vec{r}_{vv} = \frac{\partial^2 \vec{r}}{\partial v^2}, \quad \vec{r}_{uv} = \frac{\partial^2 \vec{r}}{\partial u \partial v}$$

Le vecteur normal $\vec{n}(u, v)$ est donné par $\vec{n}(u, v) = \frac{\vec{r}_u(u, v) \times \vec{r}_v(u, v)}{\|\vec{r}_u(u, v) \times \vec{r}_v(u, v)\|}$

Remarque : Dans le cas où $\vec{r}(u, v)$ est une surface du second ordre, les différentes dérivées valent :

$$\begin{cases} \vec{r}_u = \vec{a}_{10} + \vec{a}_{11} \cdot v_0 + 2\vec{a}_{20} \cdot u_0 \\ \vec{r}_v = \vec{a}_{01} + \vec{a}_{11} \cdot u_0 + 2\vec{a}_{02} \cdot v_0 \\ \vec{r}_{uv} = \vec{a}_{11} \\ \vec{r}_{uu} = 2\vec{a}_{20} \\ \vec{r}_{vv} = 2\vec{a}_{02} \end{cases}$$

où (u_0, v_0) sont les coordonnées paramétriques du centre du voisinage d'évaluation.

III.2.3. Vérification des tolérances

- Vérification de l'orientation locale de la normale aux surfaces :

Soit α_{tol} , l'angle maximum toléré entre les deux normales, alors on doit avoir :

$$-\alpha_{tol} \leq \text{Angle}(\vec{N}_S, \vec{N}_P) \leq +\alpha_{tol}$$

- Vérification des variations de courbures :

Soient H_{tol} et K_{tol} les tolérances de variations de courbures moyenne et gaussienne localement. On doit avoir :

$$\begin{cases} \|H_S - H_N\| \leq H_{tol} \\ \|K_S - K_N\| \leq K_{tol} \end{cases}$$

IV. Méthode d'inspection d'objets de formes non-complexes

Après segmentation de l'ensemble de points 3D, on dispose d'autant d'ensembles de points que de surfaces géométriques présentées par l'objet inspecté. Ceci va nous permettre de vérifier facilement les différentes tolérances de surface imposées à cet objet. Nous décrivons dans un premier temps de façon théorique comment vérifier chacune de ces tolérances, puis appliquons notre théorie à un exemple concret.

IV.1. Défauts d'usinage courants

Nous présentons une correspondance entre les termes utilisés pour les défauts d'usinage (figure V.15) et les tolérances géométriques [Turlan 88] :

- Ovalisation : Maximum de la différence sur diamètre de deux mesures perpendiculaires (défaut de circularité),
- Triangulation : Ecart maximal entre le diamètre du cercle inscrit et le diamètre du cercle circonscrit (défaut de circularité),
- Conicité : Différence de diamètres sur une longueur donnée L (défaut de parallélisme des génératrices entre elles),
- Tonneau : Différence entre le diamètre de la section centrale et les diamètre des sections extrêmes (défaut de rectitude des génératrices),
- Diabolo : Différence entre le diamètre des sections extrêmes et le diamètre de la section centrale (défaut de rectitude des génératrices),
- Fausse planéité: Ecart ente le creux le plus profond et le plan en appui sur les bosses le plus favorablement placé par rapport au plan moyen du pourtour (défaut de parallélisme),
- Faux parallélisme: Ecart de distance d'un axe, ou d'une face, avec un autre axe, ou une autre face, sur une longueur donnée L (mesurée perpendiculairement à la face ou à l'axe de référence) (défaut de parallélisme),
- Faux équerrage : Ecart avec l'angle droit de deux axes ou faces sur une longueur donnée L (défaut de perpendicularité),
- Désaxage : Ecart de deux axes (coaxialité),

- Faux rond : Variation de rayon sur 360° par rapport à un axe de référence (défaut de battement simple radial).

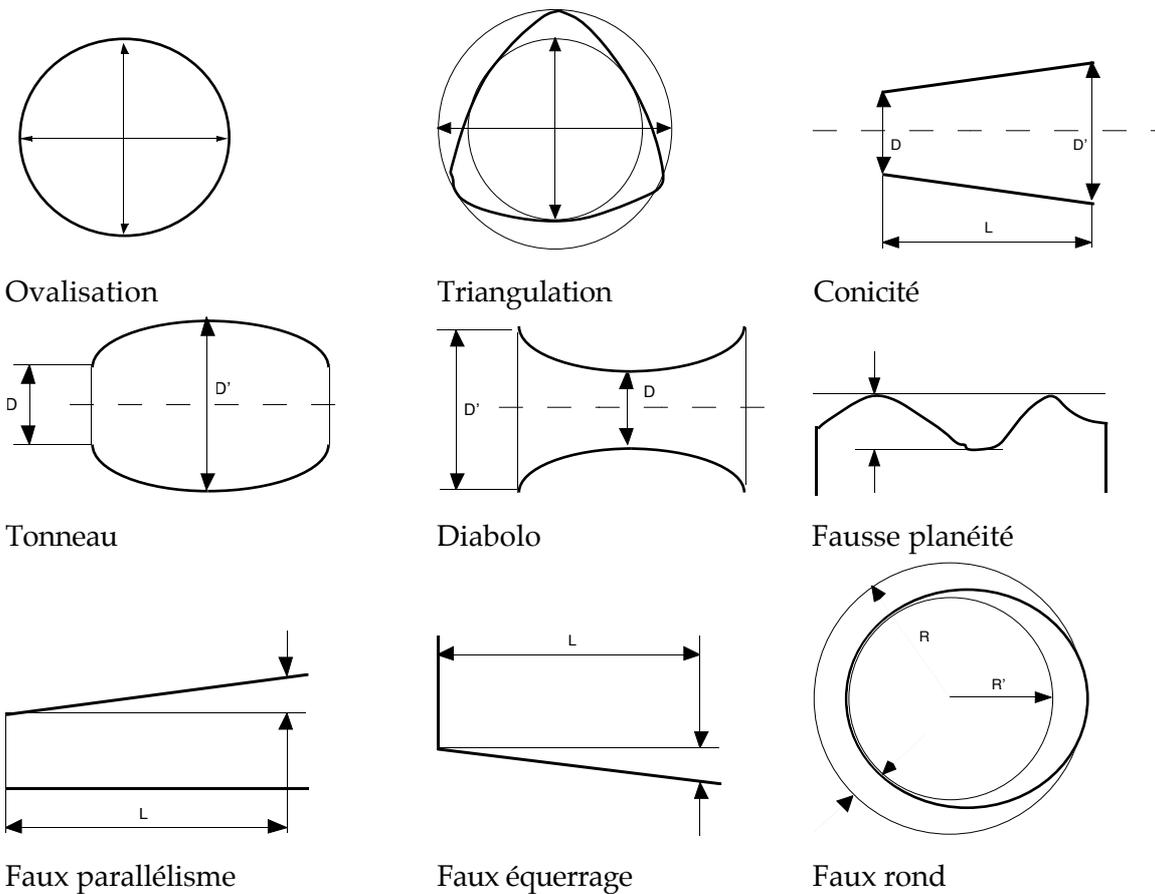


Figure V.15 : Défauts d'usinage.

IV.2. Raisonnement point par point

La pièce étudiée (*Support*) étant constituée de plans et de portions de cylindre (rayons de raccordement, trous), nous nous attachons à contrôler les tolérances de planéité et de circularité.

Cette pièce a été à la fois inspectée en métrologie et numérisée à l'aide du capteur télémétrique, ce qui permettra pour chacune des caractéristiques inspectées de comparer les deux techniques d'acquisition. En métrologie, les surfaces $z = 0$ et $y = 0$ sont choisies comme surfaces de référence.

La présentation du plan d'inspection et les résultats numériques obtenus figurent en Annexe E.

IV.2.1. Etude des défauts de la pièce étudiée

Pour chaque pas du plan d'inspection nous réalisons une comparaison des résultats fournis par la machine à mesurer et le capteur télémétrique, puis une détermination des défauts sur la pièce réelle. Les résultats qui apparaissent dans les tableaux sont exprimés en mm et correspondent à l'écart entre la cote calculée à l'aide des données laser ou métrologiques (machine à mesurer) et la cote réelle. Nous considérons une tolérance générale de 0,1 mm sur la cotation.

Plans $x=Constante$ et $y=Constante$

	Laser	Métrologie	Méto.-Laser
x=0	0,0026	0,0866	0,084
x=100	-0,0200	0,0736	0,0936
y=0	0,0200	0,0112	-0,0088
y=70	-0,0565	-0,1023	-0,0458
x=31 (lamage gauche)	-0,5089	-0,3688	0,1401
x=69 (lamage droit)	-0,4261	-0,3651	0,061
y=19 (lamage gauche)	-0,0537	-0,0948	-0,0411
y=51 (lamage gauche)	-0,1512	-0,1005	0,0507
y=19 (lamage droit)	0,0076	0,0898	0,0822
y=51 (lamage droit)	-0,0987	-0,0951	0,0036

Tableau V.1 : Plans $x=Constante$ et $y=Constante$

On constate une différence assez importante entre les mesures réalisées à l'aide du capteur télémétrique et celles effectuées avec la machine à mesurer (presque 0,1 mm sur certaines mesures). Ceci est dû à la méthode d'acquisition, et sera discuté paragraphe IV.2.2.

- Plans $x=0$, $x=100$ (Annexe E.1), $y=0$ et $y=70$ (Annexe E.2)

La longueur, la largeur de la pièce, comme la planéité des quatre plans extrêmes est correcte.

- Plans $x=31$ et $x=69$ (Annexe E.3)

Les planéités des deux lamages pour $x=31$ et $x=69$ sont correctes. On constate un mauvais positionnement de ces plans de plusieurs dixièmes de mm.

- Plans $y=19$ et $y=51$ (deux lamages) (Annexe E.4)

Les largeurs des lamages sont correctes, mais on constate un mauvais positionnement de ces plans d'un dixième de mm.

Plans $z=constante$

	Laser	Métrieologie	Métrie.-Laser
$z=20$ pour $y=15$	-0,0251	-0,0024	0,0227
$z=20$ pour $y=55$	-0,0006	-0,0046	-0,004
$z=20$ pour $x=40$	-0,0203	0,0152	0,0355
$z=20$ pour $x=60$	-0,0075	-0,0012	0,0063
$z=15,5$ pour $x=2,5$	0,1609	0,1539	-0,007
$z=15,5$ pour $x=27$	0,1853	0,1956	0,0103
$z=15,5$ pour $x=71$	0,2188	0,2587	0,0399
$z=15,5$ pour $x=99$	0,2979	0,2785	-0,0194
$z=15,5$ pour $y=21$ (lamage gauche)	0,1106	0,1242	0,0136
$z=15,5$ pour $y=49$ (lamage gauche)	0,2089	0,2258	0,0169
$z=15,5$ pour $y=21$ (lamage droit)	0,1831	0,2245	0,0414
$z=15,5$ pour $y=49$ (lamage droit)	0,2831	0,3154	0,0323

Tableau V.2 : Plans $z=Constante$.

Nous constatons un écart entre les mesures fournies par les deux procédés d'acquisition, bien plus faible que pour les mesures effectuées sur x et y (quelques centièmes de mm).

- Planéité de la face supérieure ($z=20$) (Annexes E.5, E.6)

La planéité de la surface supérieure de la pièce, ainsi que son épaisseur, est correcte.

- Planéité des lamages ($z=15,5$) (Annexes E.7, E.8)

On constate une inclinaison des deux lamages dans les deux directions.

Circularité des deux perçages (Annexe E.9)

	Cercle de gauche		Cercle de droite	
	Laser	Méto.	Laser	Méto.
Ovalisation	0,0747	0,0283	0,0390	0,0507
Direction axe principal	45°	51°	93°	129°
Triangulation	0,0855	0,0375	0,0469	0,0507
Faux rond	0,0961	0,0224	0,0613	0,0268

Tableau V.3 : Circularité des deux perçages.

Des données métrologiques, nous déduisons que le cylindre 1 a un diamètre moyen de 10,083 mm et un écart moyen de 0,005 mm, et que le cylindre 2 a une diamètre moyen de 12,498 mm et un écart moyen de 0,007 mm, ce qui est tout à fait tolérable. Les deux cylindres ont une forme légèrement ovoïde, qui s'explique par une probable rotation excentrée de l'outil.

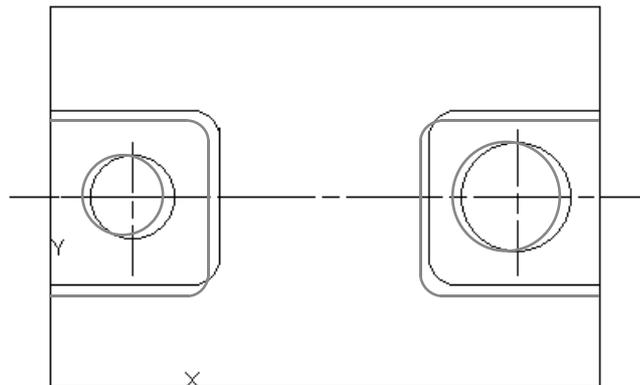


Figure V.16 : Vue de dessus.

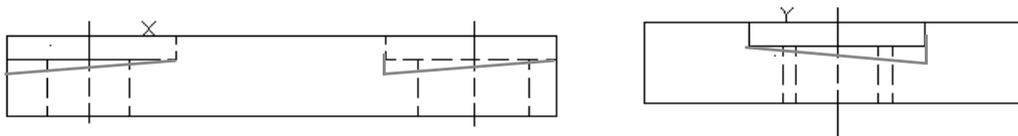


Figure V.17 : Vue de face et vue de gauche.

Les figures V.16 et V.17 montrent en gris les défauts constatés sur la pièce support.

IV.2.2. Comparaison acquisition par machine à mesurer et capteur laser

On constate une bien meilleure fiabilité des mesures fournies par la machine à mesurer. En effet, d'une part les points 3D fournis par le capteur télémétrique sont bien moins fiables, de par la technologie d'acquisition (précision de $\pm 25 \mu m$) que ceux fournis par la machine à mesurer (précision de $1 \mu m$). D'autre part, l'approche du palpeur de la machine à mesurer se fait toujours perpendiculairement à la surface inspectée, donc dans de meilleures conditions de précision (figure V.18). Pour le capteur télémétrique, l'incidence faisceau laser/surface inspectée n'est pas toujours optimale, lorsque l'incidence du faisceau est perpendiculaire à la surface, les points 3D obtenus sont proches de ceux fournis par la machine à mesurer, dans le cas contraire, les points obtenus sont décalés par rapport à ceux de la machine à mesurer. Par ailleurs, les données 3D laser résultent de la fusion de plusieurs images 3D, donc il peut y avoir des incertitudes lors de cette fusion (incertitudes sur les localisation du capteur).

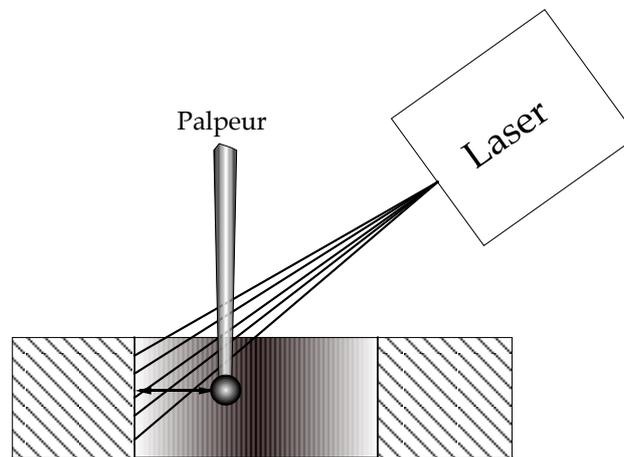


Figure V.18 : Méthodes de mesure laser et métrologique sur le diamètre d'un cylindre.

IV.3. Raisonnement par caractéristique

Ce type de raisonnement qui n'a pas été développé dans notre cas, il consiste à faire passer dans un sous-ensemble de points correspondant à la caractéristique étudiée une surface dont il faut trouver les paramètres. Par exemple, faire passer une surface-cylindre dans un ensemble de points de la numérisation d'un cylindre, afin de trouver son rayon, son positionnement, son orientation, et étudier ses tolérance de forme...

Conclusion

Nous avons présenté dans ce chapitre des outils pour l'inspection automatique visuelle de pièces présentant des formes complexes ou non-complexes. Nous avons effectué un état de l'art dans le domaine de la cotation et de la définition de tolérances de pièces de forme

complexes ou non-complexes, puis dans le domaine de l'inspection visuelle automatique. Ceci nous a amené à proposer notre propre stratégie d'inspection, qui à partir d'un ensemble de points 3D et d'un modèle CAO, réalise une inspection complète (globale et locale) des surfaces de l'objet étudié, alors que la plupart des méthodes courantes ne vérifient que la tolérance de forme globale. Nous avons également comparé les performances d'une machine à mesurer avec celles d'un capteur télémétrique, pour déceler les défauts d'une pièce réelle. Nous en déduisons que la machine à mesurer de par son procédé d'acquisition (approche perpendiculaire du palpeur) et sa précision (1 μm) fournit des résultats d'une grande fiabilité, qui sont uniquement limités par les dimensions du palpeur. Quant au capteur télémétrique, nous savons que par son principe de fonctionnement sa précision est limitée ; le constructeur garantit +/- 25 μm pour le type de capteur utilisé, ce que nous avons vérifié lors des mesures. De plus, certaines zones (intérieur d'alésage, surfaces verticales,...) difficilement accessibles au rayon laser, donnent ces mesures entachées d'erreurs. Nous avons pu montrer que le capteur télémétrique permet de déceler des défauts de l'ordre du dixième de mm, alors que la machine à mesurer permet le centième de mm.

Chapitre VI

Applications

Nous présentons dans ce chapitre différentes applications de notre méthode de mise en correspondance :

- La reconnaissance de formes,
- L'inspection d'objets industriels,
- La vérification de la présence de matière suffisante après ébauche et avant finition d'une pièce.

La méthode de mise en correspondance ainsi que ses applications ont été implémentées sur une station de travail Silicon Graphics Indy munie d'un processeur RISC Mips R5000SC 64 bits (150 Mhz) et programmées en C++ (Gnu g++).

I. Reconnaissance de formes

Il existe de nombreuses manières de résoudre les problèmes de reconnaissance de formes en 2D. L'une d'elles consiste à utiliser des modèles complexes associés à des classifieurs simples (plus proches voisins, distance euclidienne) ; à l'opposé, on peut utiliser des modèles simples, associés à des classifieurs plus complexes [Milgram 92].

Pour reconnaître des objets 3D, on utilise généralement des méthodes de reconnaissance 2D, en considérant toutes les vues potentielles de l'objet. L'inconvénient majeur de ces méthodes est de multiplier d'une façon significative la quantité de données à traiter.

Lorsqu'on dispose de données 3D, il est intéressant d'utiliser la modélisation CAO (surfaces, volumes, etc ...) pour décrire la géométrie des objets 3D à reconnaître. Notre méthode de mise en correspondance [Données 3D/Modèle CAO], automatique et robuste,

résoud les problèmes de mise en correspondance classiques créés par la grande quantité d'informations à traiter et par la complexité des modèles utilisés.

Pour conformer notre propos, nous appliquons notre méthode de mise en correspondance à deux ensembles d'objets en utilisant la méthode de modélisation STL. Le premier est constitué de quatre objets paramétriques polyédriques (*polyèdre 1, polyèdre 2, polyèdre 3, polyèdre 4*) tandis que le second est un ensemble de trois pièces plus complexes (*Support 1, Support 2, Support 3*) se distinguant par deux perçages de diamètres différents. Un objet de chaque famille a été numérisé (*polyèdre 4* et *Support 2*). Le critère de reconnaissance utilisé est la plus petite distance moyenne [données 3D/Modèles CAO NURBS découpées], celle-ci étant la somme de toutes les distances Points (transformés par la transformation rigide déterminée par l'algorithme de mise en correspondance)/Modèle CAO divisée par le nombre de points utilisés.

I.1. Polyèdres

La méthode de mise en correspondance, utilisant les quaternions pour éviter les minimums locaux, a été appliquée entre les données 3D issues de la numérisation et chacun des modèles CAO triangulés. Le tableau VI.1 donne la distance moyenne obtenue, pour chaque modèle CAO utilisé. La plus petite distance moyenne globale est obtenue pour le *polyèdre4*, qui était l'objet à reconnaître et qui procure une mise en correspondance parfaite. La non-nullité de cette distance s'explique par l'existence d'erreurs d'usinage sur l'objet numérisé, ainsi que par la présence de bruit sur les données capteur.

<i>Polyèdre 1</i>	<i>Polyèdre 2</i>	<i>Polyèdre 3</i>	<i>Polyèdre 4</i>
Distance moyenne (mm)	Distance moyenne (mm)	Distance moyenne (mm)	Distance moyenne (mm)
3,604	2,254	1,198	0,173

Tableau VI.1 : Distance moyenne obtenue pour chaque modèle CAO

I.2. Supports

Comme pour les objets polyédriques, la méthode de mise en correspondance utilisant les quaternions a été utilisée ; les résultats obtenus figurent dans le tableau VI.2.

<i>Support 1</i>	<i>Support 2</i>	<i>Support 3</i>
Distance Moyenne (mm)	Distance Moyenne (mm)	Distance Moyenne (mm)
0,147	0,083	0,117

Tableau VI.2 : Distance moyenne pour chacun des modèles CAO

Le support 2, qui présente la distance moyenne la plus faible, a été reconnu.

II. Inspection d'objets industriels

Différents objets conçus en CAO ont été usinés, puis numérisés ; les données correspondantes figurent en Annexe C. Une fois les points 3D mis en correspondance avec le modèle CAO, chaque distance [point/Modèle CAO NURBS découpées] est calculée. Après avoir proposé des outils (chapitre V) permettant de réaliser une inspection quantitative, nous proposons maintenant des outils pour une inspection graphique. Pour cela, nous faisons apparaître graphiquement, par différents procédés, la localisation et la quantification des erreurs points/modèle CAO relevées :

- Un premier procédé de visualisation consiste à attribuer à chaque triangle du modèle CAO une couleur dépendant de la distance moyenne triangle/points voisins.
- La seconde consiste à représenter chaque segment point 3D/point associé du modèle CAO avec une couleur dépendant de la distance entre les deux points ou une couleur dépendant d'une valeur distance seuil.
- La troisième consiste à associer à chaque surface une carte de texture plus ou moins fine dont la couleur locale dépend de la distance avec les points 3D voisins.

Dans ces trois cas (sauf pour le seuillage), il existe une correspondance entre la distance représentée et la couleur utilisée (Figure VI.1, page 147). Les couleurs s'échelonnent entre le bleu pour une erreur nulle et le rouge pour l'erreur maximum, le vert représentant l'erreur maximale divisée par deux.

II.1. Procédés de visualisation

Nous détaillons maintenant les trois procédés de visualisation que nous avons utilisés.

II.1.1. Coloration des triangles

A chaque triangle est associé une couleur dépendant de la distance moyenne de ce triangle avec ses points voisins. Les triangles pour lesquels il n'y a pas de points associés apparaissent en gris.

Cette représentation ne requiert pour la visualisation que le modèle STL de l'objet, le fichier de points transformés et le fichier de couleur des triangles généré par le programme de mise en correspondance.

Son inconvénient provient de la limitation de la représentation STL qui, à un plan de n'importe quelle dimension, va faire correspondre un faible nombre de triangles quelque soit

la valeur d'approximation STL. Ainsi, il est toujours difficile d'inspecter localement une surface plane décrite par une triangulation STL.

Nous proposons deux solutions pour pallier cet inconvénient :

- Subdiviser chaque triangle en un ensemble de triangles tel que chaque surface soit décrite de façon suffisamment précise.
- Utiliser directement un autre type de triangulation plus adapté (triangulation éléments finis par exemple).

II.1.2. Carte de texture

Dans le domaine du graphisme par ordinateur, à chaque surface à afficher est associée une carte de texture qui prend en compte la forme de la surface, les sources d'éclairage, et la texture de l'objet observé (figure VI.2), et qui épouse la forme de la surface géométrique à afficher.

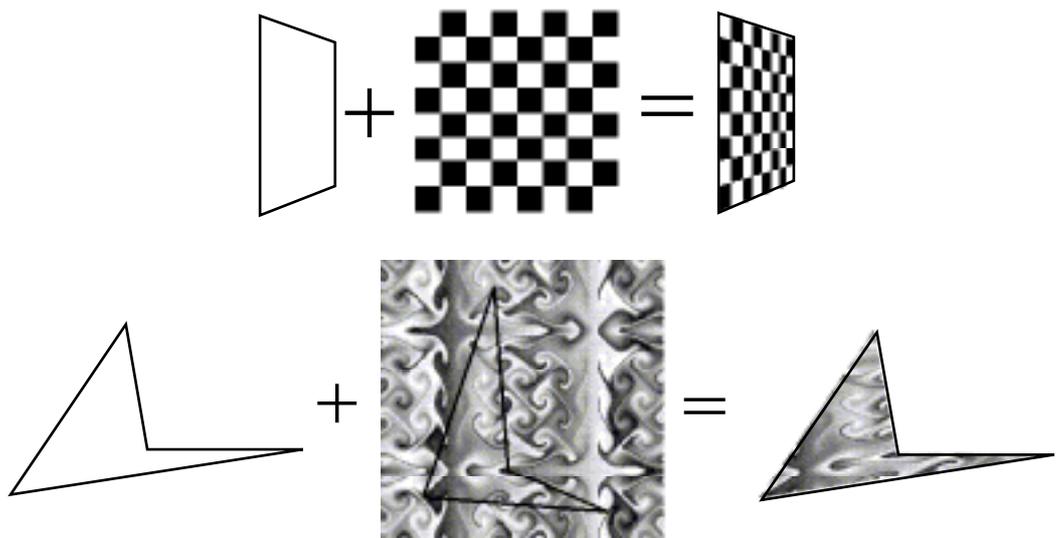
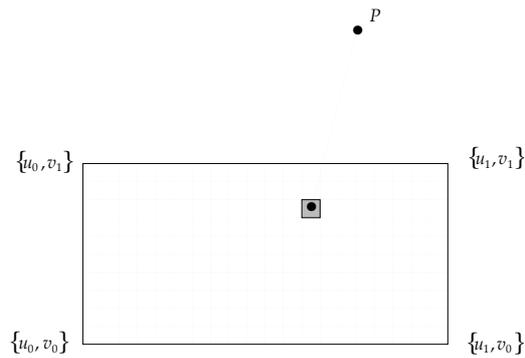


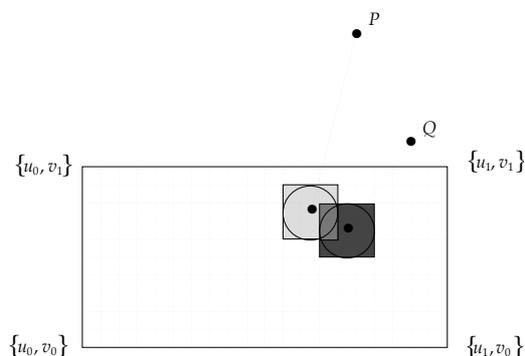
Figure VI.2 : Exemples de texture mapping, exemples extraits de "Open GL programming guide", Addison Wesley publishing Company, 860-0105-002.

Nous proposons de modifier cette carte (grille) de texture de deux façons différentes :

- *Texture Mapping 1* : Pour chaque carreau élémentaire de la carte de texture, nous attribuons une couleur de l'échelle figure VI.1 page 147, en fonction de la distance moyenne carreau/points voisins (voir figure ci dessous).



- *Texture Mapping 2* : Chaque point 3D se projette sur la carte de texture sous forme d'un disque dont la couleur dépend de la distance point/surface (figure ci-dessous).



Dans les deux cas, il s'agit de trouver un compromis sur la taille du carreau (disque) unité. Si celui-ci est trop petit, il sera difficilement visible et s'il est trop gros, la localisation des erreurs devient difficile. Pour le texture Mapping 2, il peut y avoir recouvrement des disques, et donc moyennage des erreurs, l'interprétation de la visualisation devient alors plus difficile.

II.1.3. Tracé des segments d'erreur

Chaque point 3D est relié à son point le plus proche sur le modèle géométrique par un segment de couleur, dont la couleur dépend de l'erreur locale (qui est également la longueur du segment).

Nous avons implémenté deux façons d'afficher les couleurs :

- **Seuillage** : Une valeur de distance-seuil d_{seuil} est fixée par l'opérateur de façon interactive ; l'affichage est modifié en temps réel. Pour les points dont la distance est comprise entre 0 et d_{seuil} , les segments apparaissent en bleu, les autres apparaissent en rouge. C'est une façon simple de sélectionner les points horstolérance de forme globale.
- De la même façon qu'au paragraphe précédent, deux valeurs extrêmes d_{min} et d_{max} sont fixées par l'opérateur. Pour les points dont la distance est comprise entre ces

deux bornes, les segments apparaissent avec une couleur dépendant de l'échelle présentée figure VI.1 page 147, s'étendant du bleu pour d_{\min} au rouge pour d_{\max} . Cette technique de visualisation permet d'adapter localement l'échelle de couleur et de mieux apprécier les erreurs.

Nous présentons maintenant des résultats d'inspection utilisant toutes les méthodes présentées dans ce mémoire.

II.2. Exemples

Les objets qui seront inspectés graphiquement ont été numérisés.

II.2.1. Polyèdre

{	Données 3D	:	Objet "Polyèdre" numérisé
	Modèle STL	:	12 triangles
	N_t	:	10
	N_s	:	30

De ces différentes visualisations (figures VI.3,VI.4,VI.5 page 147), nous déduisons qu'une des faces est mal positionnée (erreur de 0,93 mm), à cause d'une erreur d'usinage.

II.2.2. Plaque

{	Données 3D	:	Objet "Plaque" numérisé à l'aide d'un capteur télémétrique
	Modèle STL	:	994 triangles
	N_t	:	10
	N_s	:	50
[Temps de calcul : 38 mn 41 s			

La visualisation (figures VI.6 et VI.7 page 148) montre que cette plaque réalisée en prototypage rapide présente une courbure progressive anormale sur sa face supérieure allant jusqu'à une erreur de forme de 1,22 mm.

II.2.3. Support

Points 3D provenant du capteur télémétrique

{	Données 3D	:	Objet "Support" numérisé à l'aide d'un capteur télémétrique
	Modèle STL	:	1636 triangles
	N_t	:	5
	N_s	:	200
[Temps de calcul : 8 mn 22 s			

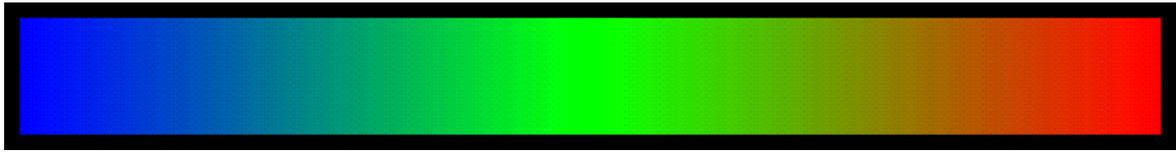


Figure VI.1 : Echelle de couleur (le bleu correspond à une erreur nulle et le rouge à l'erreur maximum).

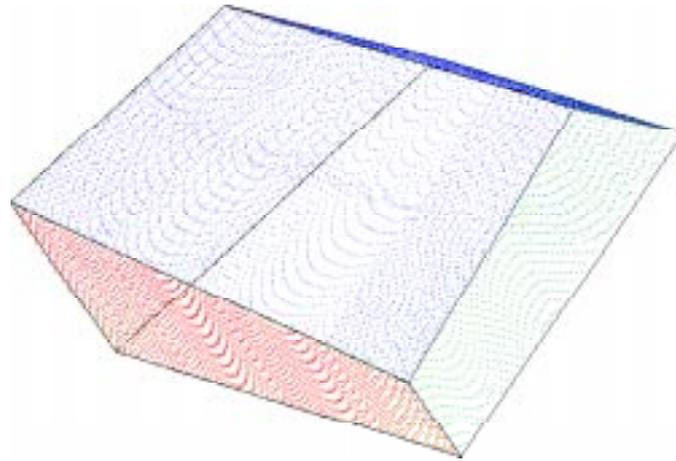


Figure VI.3 : Affichage des segments d'erreurs, objet *polyèdre* (erreur maximum = 0,93 mm).

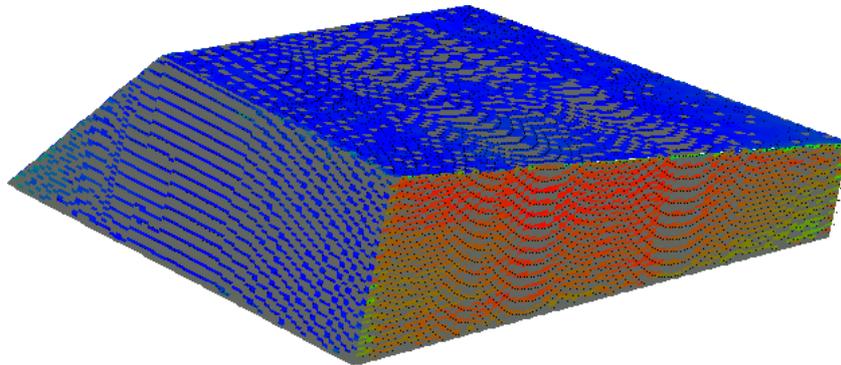


Figure VI.4 : Objet *polyèdre* : Texture mapping 1.

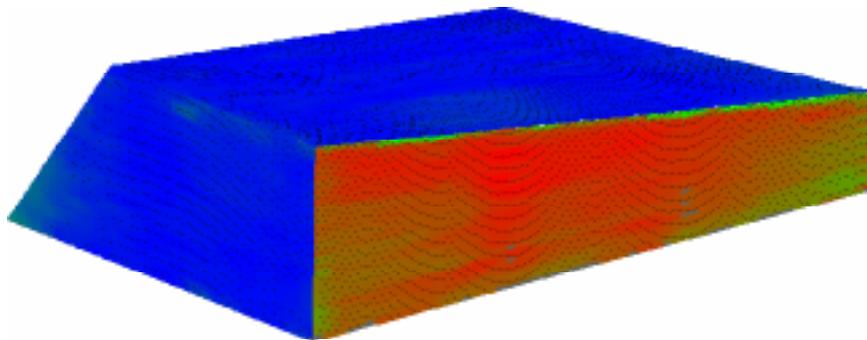


Figure VI.5 : Objet *polyèdre* : Texture Mapping 2.

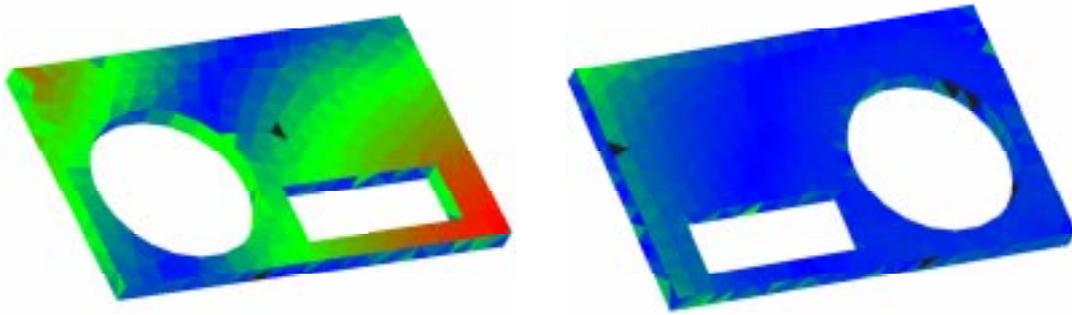


Figure VI.6 : Visualisation de l'objet *Plaque* utilisant les triangles colorés, vue de dessus à gauche, vue de dessous à droite, erreur maximale de 1,22 mm (rouge).

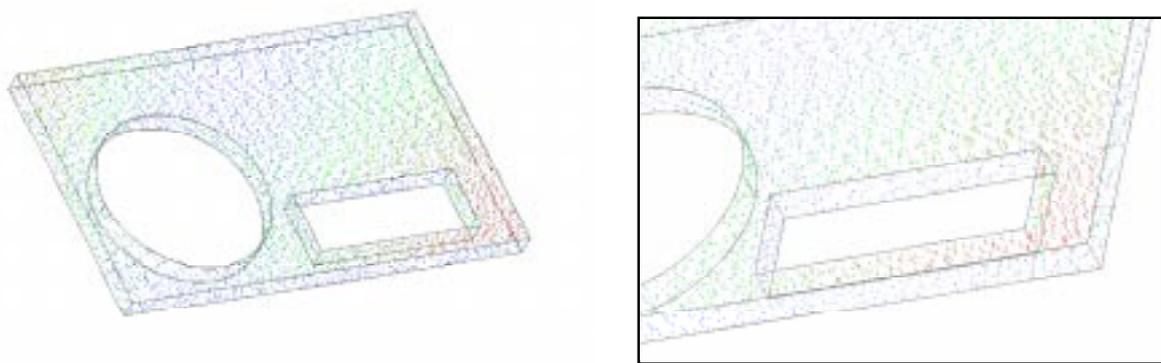


Figure VI.7 : Visualisation de l'objet *Plaque* utilisant les segments colorés, vue de dessus à gauche et zoom sur les erreurs les plus importantes (erreur maximale de 1,22 mm).

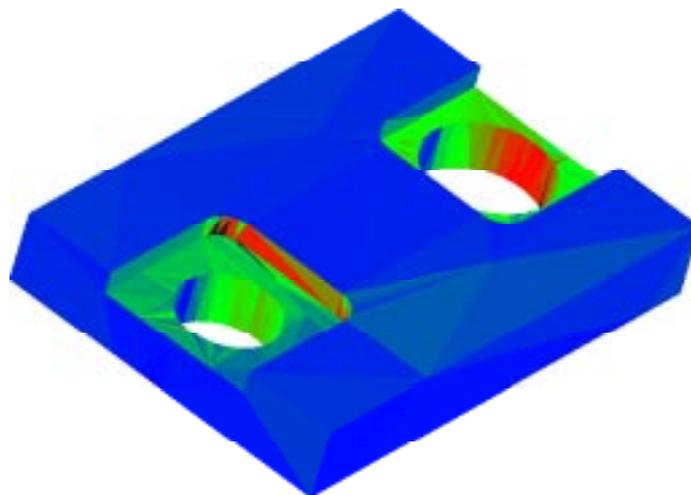


Figure VI.8 : Représentation avec triangles colorés de l'objet *Support*, erreur maximum/triangle : 0,54 mm.

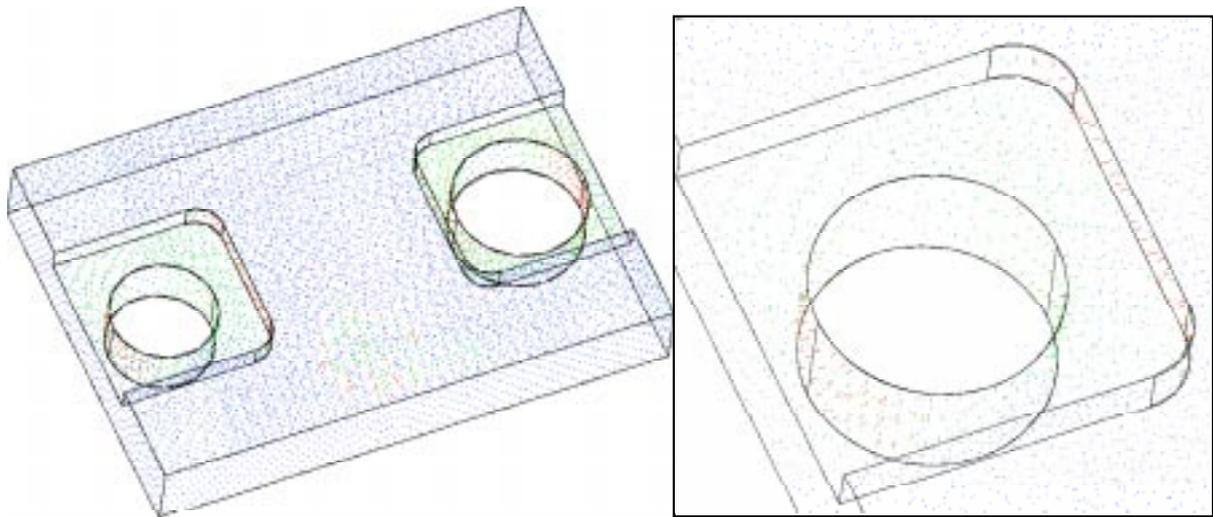


Figure VI.9: Représentation par segments colorés, vue de dessus à gauche et agrandissement du perçage gauche à droite, erreur maximale : 1,62 mm.

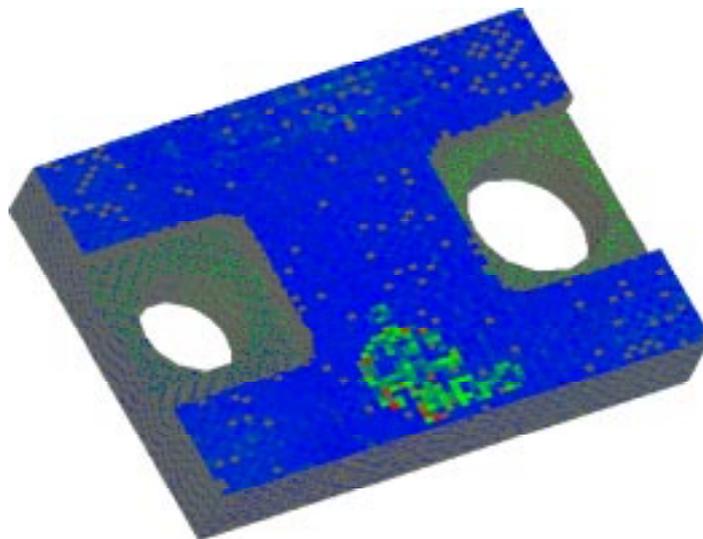


Figure VI.10 : Objet *Support* : Texture Mapping 1.

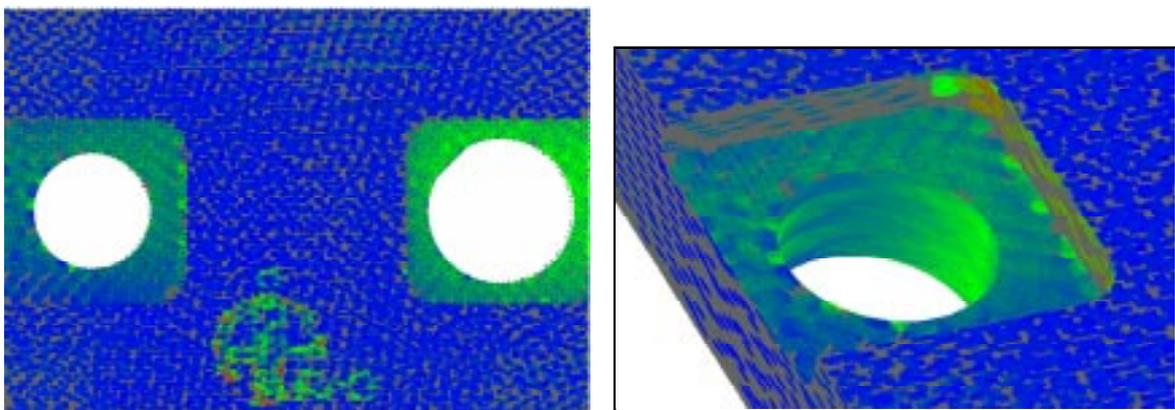


Figure VI.11 : Objet *Support* : Texture Mapping 2.

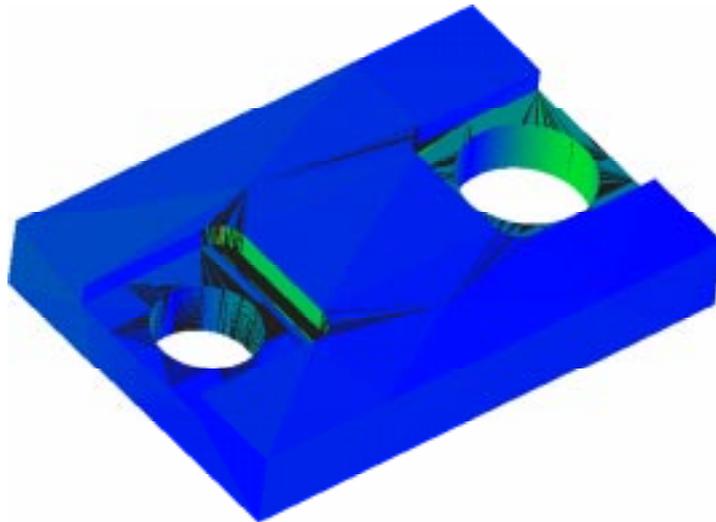


Figure VI.13 : Visualisation des erreurs avec les triangles colorés (erreur maximum : 1,02 mm).

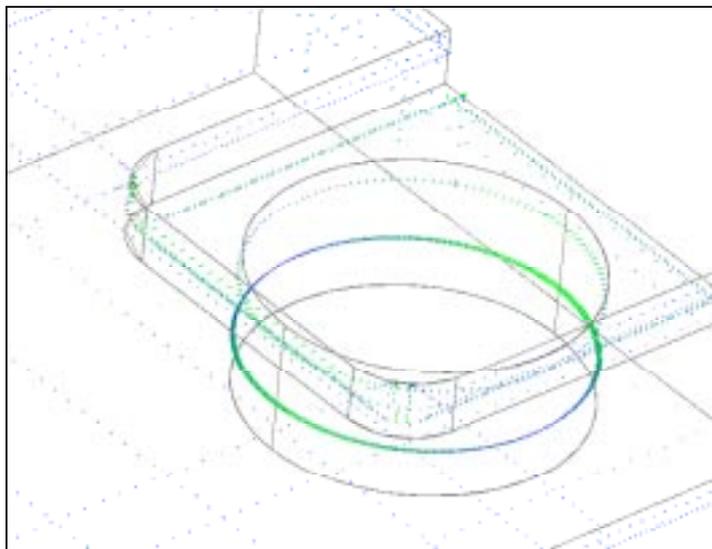


Figure VI.14 : Visualisation des segments : Inspection du perçage de droite (erreur maximale : 1,06 mm).

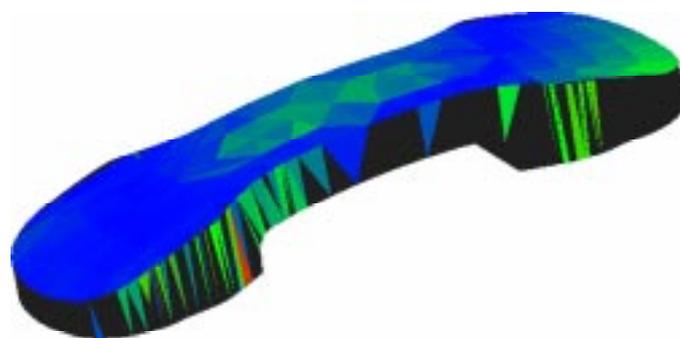


Figure VI.16 : Visualisation avec triangles colorés de la mise en correspondance de l'image 3D avec le modèle CAO STL (erreur maximum 0,26 mm).

La figure VI.8 (page 148) montre la représentation des erreurs avec les triangles colorés, la figure VI.9 (page 149) montre la représentation des erreurs avec les segments colorés et les figures VI.10 et VI.11 (page 149) avec les deux types de texture mapping.

Points 3D provenant de la machine à mesurer

Données 3D	: Objet "Support" numérisé à l'aide d'une machine à mesurer
Modèle STL	: 1636 triangles
N_t	: 5
N_s	: 200
Temps de calcul	: 11 mn 14 s

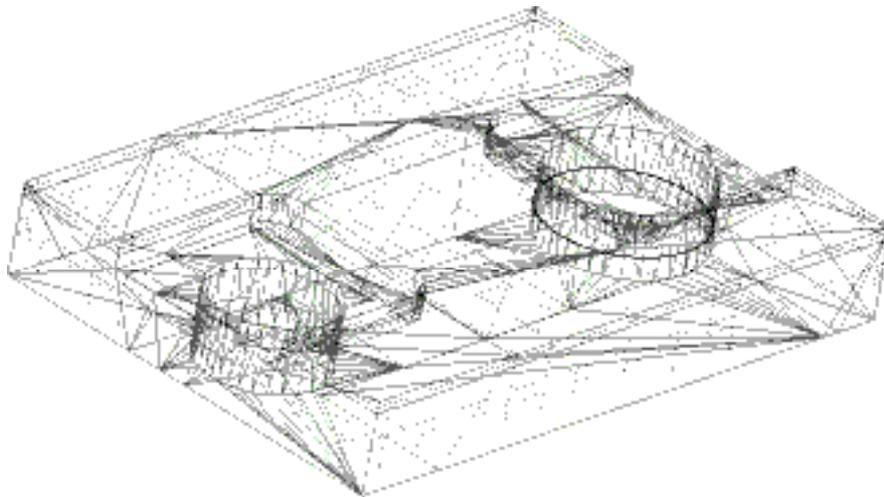


Figure VI.12 : Mise en correspondance points 3D/modèle CAO STL.

Les erreurs de fabrication sont mises en évidence dans les figures VI.13 et VI.14 (page 150) en utilisant les triangles et les segments colorés.

Conclusion

A partir des deux types de points 3D, nous retrouvons graphiquement sur cet objet les mêmes erreurs que celles signalées chapitre V :

- Inclinaison des deux lamages,
- Desaxage et forme elliptique des deux perçages,
- Mauvais positionnement en x des deux lamages.

Sur cet objet, on peut juger facilement de l'efficacité de chacune des méthodes de visualisation. En effet, sur la face supérieure de l'objet est gravé un logo. Celui-ci ne figure pas dans le modèle CAO. Il apparaît donc une erreur importante dans cette zone de gravure.

II.2.4. Téléphone

Données 3D	:	Objet "Téléphone" numérisé à l'aide d'un capteur télémétrique
Modèle STL	: 13760 triangles	
N_t	: 10	
N_s	: 30	

Pour cet objet, nous disposons de deux types de données : une seule image 3D, qui correspond à une seule vue numérisée par le capteur télémétrique et des données 3D qui résultent de la fusion de plusieurs images 3D (Chapitre III, paragraphe I).

Une seule image 3D



Figure VI.15 : Mise en correspondance de l'image 3D avec le modèle CAO STL.

On constate qu'une seule image 3D est facilement mise en correspondance avec le modèle STL (figure VI.15). La visualisation des triangles colorés (figure VI.16 page 150), fait apparaître une variation de couleur au centre de la face supérieure de la pièce, due à la présence d'un logo gravé à cet endroit.

Données 3D complètes

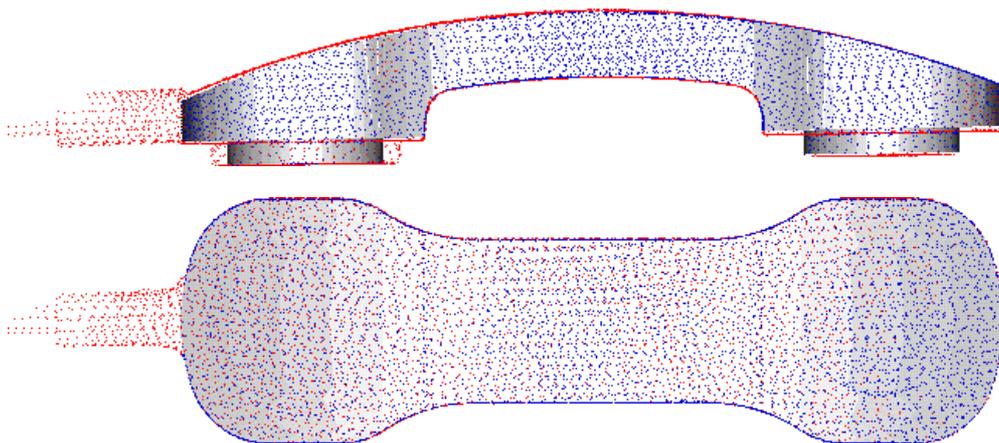


Figure VI.17 : Résultat de la mise en correspondance des points 3D avec le modèle CAO STL.

Mis à part les points qui appartiennent à la fixation de l'objet, on constate une erreur de 5 mm sur le diamètre de la partie cylindrique gauche. On vérifie également que les points 3D "parasites", correspondant à la tige de fixation de la pièce lors de sa numérisation, ne gênent en rien la mise en correspondance et prouvent la robustesse de la méthode.

II.2.5. IMS

Données 3D	:	Objet "IMS" numérisé à l'aide d'un capteur télémétrique
Modèle STL	:	39544 triangles
N_t	:	10
N_s	:	30
Temps de calcul		: 54 mn

Sur cette pièce, on peut constater que la plaque-base est trop courte sur sa largeur ainsi que sur sa longueur, et que les éléments figurant sur cette plaque (marches d'escalier...) référencés par rapport aux extrémités de la pièce se trouvent légèrement décalés (2,3 mm).

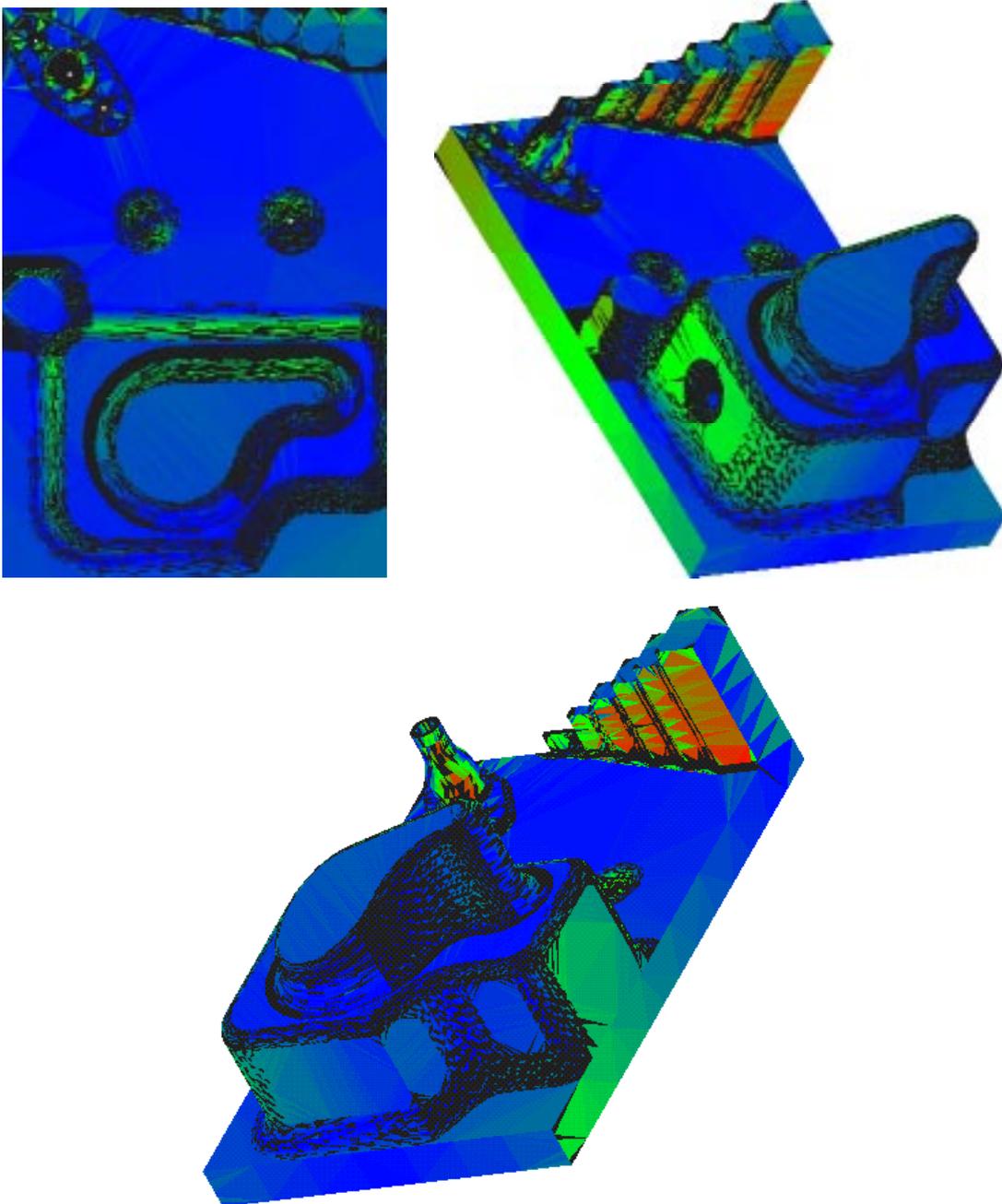


Figure VI.18 : Représentation par les triangles colorés.

II.2.6. Pièce1

{	Données 3D	:	Objet "Pièce1" numérisé à l'aide d'un capteur télémétrique
	Modèle STL	:	9826 triangles
	N_t	:	10
	N_s	:	30

On constate sur la figure ci-dessous, que la représentation par triangles colorés montre que cette pièce présente une erreur de mise en correspondance nulle à une extrémité et importante à l'autre extrémité.

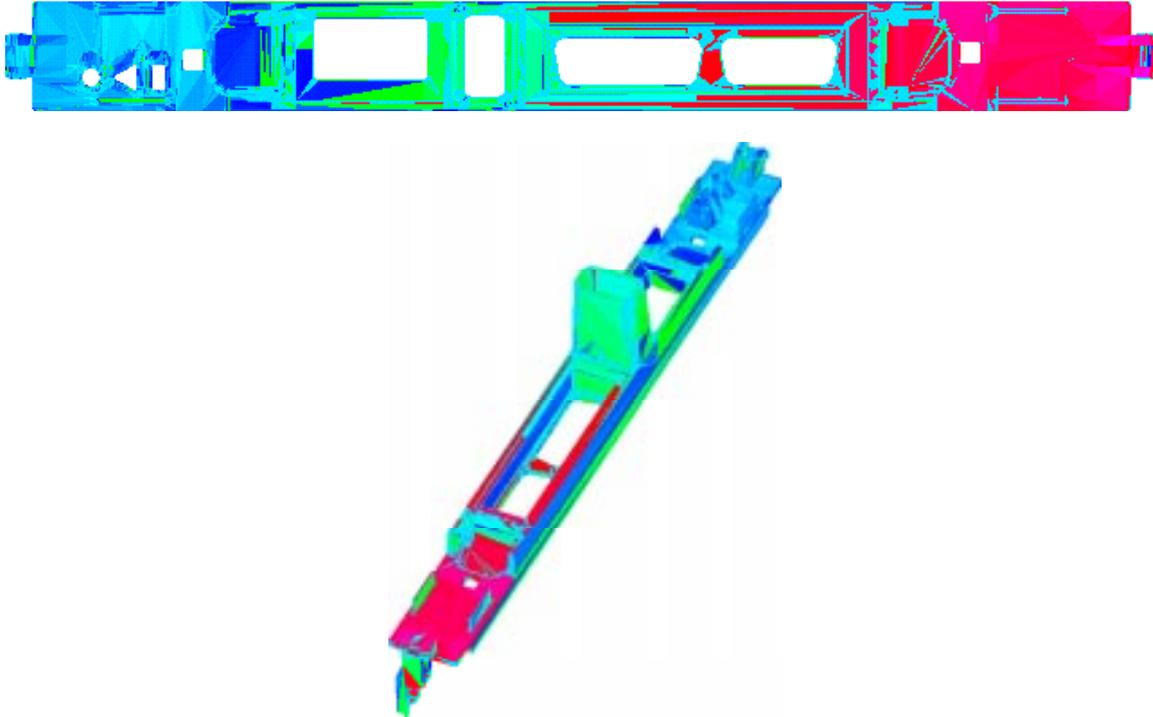


Figure VI.19 : Représentation par les triangles colorés sans affichage des points.

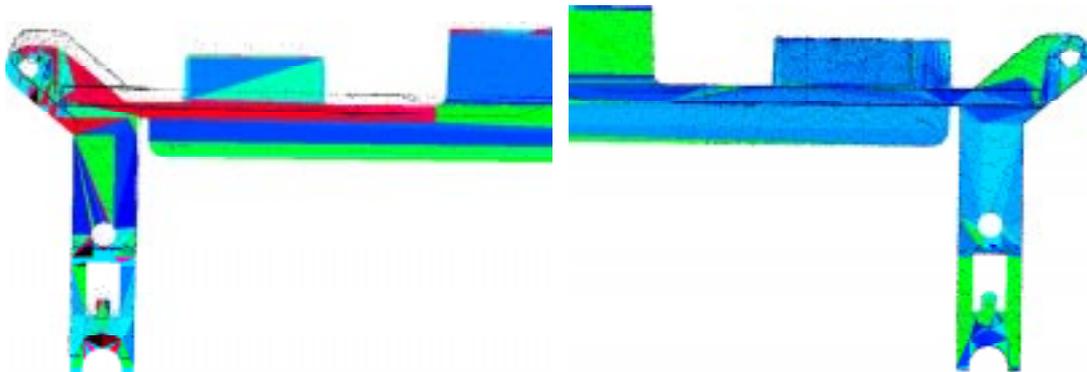


Figure VI.20 : Représentation par les triangles colorés avec affichage des points, extrémité de gauche (à gauche), extrémité de droite (à droite).

En visualisant la superposition des points aux deux extrémités, on s'aperçoit que la pièce est courbée de plusieurs mm dans le sens de sa longueur, défaut lié à sa fabrication.

III. Préparation à la finition d'une pièce ébauchée

Cette application particulière permet à partir de points 3D provenant de la numérisation d'une pièce ébauchée de vérifier qu'il reste suffisamment de matière, avant d'aborder la phase de finition (figure VI.21). Ceci nécessite, en plus des points 3D, de disposer du modèle CAO de la pièce finie et du programme de mise en correspondance où tous les points 3D sont utilisés pour la phase ICP. D'autre part, cette application est particulièrement intéressante pour des pièces de forme complexe, telle qu'une aube de turbine, difficiles à mesurer par des moyens classiques.

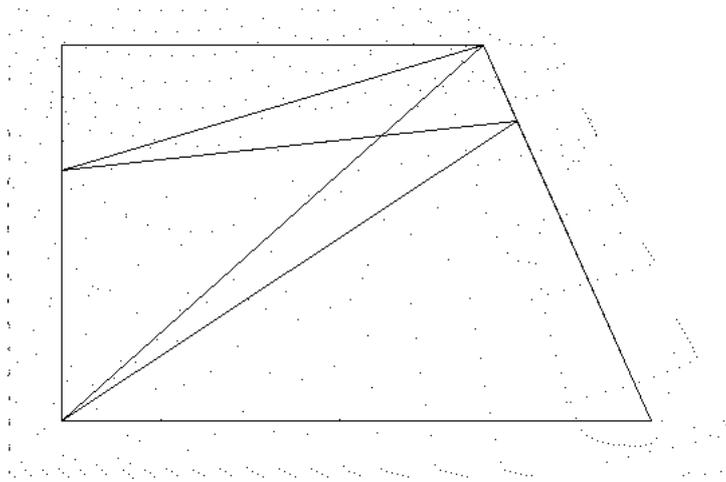


Figure VI.21 : Mise en correspondance des points de la pièce *polyèdre* ébauchée avec le modèle CAO de la pièce finie.

Conclusion

Nous avons montré dans ce chapitre que notre méthode de mise en correspondance, permet, en choisissant un classifieur simple, de faire de la reconnaissance de formes sur des objets 3D.

L'intérêt principal de notre méthode est de pouvoir superposer les données 3D de l'objet à contrôler avec le modèle géométrique, créé lors de sa conception, et de mettre ainsi en évidence les erreurs de fabrication. Cette phase d'inspection peut se faire de façon quantitative (chapitre V). Nous avons proposé également plusieurs techniques de visualisation qui permettent de localiser les défauts :

- Triangles colorés : faciles à mettre en oeuvre, mais nécessitent une triangulation adaptée pour mieux apprécier les défauts.

- Texture Mapping : Lourde à mettre en oeuvre, avec un compromis à trouver concernant la visibilité des taches d'erreur (compromis visibilité/ moyennage des erreurs).
- Segments d'erreur : Obtenus soit par seuillage, soit par échelle d'erreur colorée, cette méthode est la plus efficace car elle est vraiment locale (un segment par point) et son échelle peut varier de façon interactive selon les choix de l'opérateur.

Conclusion Générale et Perspectives

Le travail décrit dans ce mémoire porte sur l'inspection automatique d'objets comportant des surfaces complexes à partir de leur modèle CAO et de données 3D provenant d'un capteur télémétrique. Ce sujet étant nouveau pour l'équipe Vision Industrielle du LAI, il a nécessité un important travail de recherche bibliographique, dans des domaines tels que la mise en correspondance 3D, l'inspection automatique visuelle, les modèles déformables,...

Après avoir choisi deux types de modèles géométriques surfaciques représentant l'objet étudié, nous avons détaillé la chaîne d'acquisition 3D permettant d'obtenir les données 3D. Nous avons ensuite présenté notre méthode de mise en correspondance 3D générale, dont nous avons détaillé les performances. Finalement, nous présentons quelques applications à cette méthode, parmi lesquelles la Reconnaissance de formes, mais surtout l'Inspection automatique d'objets de formes complexes. Nous proposons une méthode d'inspection de ce type d'objet, d'une part quantitative avec une approche globale et locale, et d'autre part qualitative en utilisant différents procédés de visualisation.

Nous avons contribué à ce travail dans les domaines suivants :

- Une synthèse bibliographique exhaustive dans les domaines suivants :
 - Modélisation 3D (modèles géométriques, modèles vision et modèles déformables).
 - Mise en correspondance 3D
 - Inspection Visuelle Automatique
- Proposition d'une méthode de mise en correspondance générale entre des données 3D et le modèle CAO associé, qui a l'avantage d'être robuste (tolérant jusqu'à 50% de points parasites) automatique, et qui fonctionne quelque soit la position initiale des

données et du modèle. Cette méthode générale nécessite uniquement la définition de la distance entre une entité-donnée et une entité-modèle.

- Définition du calcul de distance accéléré entre un point et un ensemble de triangles, utilisant une subdivision de l'espace des triangles en voxels parallélépipédiques.
- Définition du calcul de distance entre un point et une surface NURBS découpée.
- A partir des modèles surfaciques NURBS découpées et triangulées, définition d'une représentation hiérarchisée du modèle géométrique pour accélérer le calcul de distance point/modèle.
- Développement d'outils pour l'inspection visuelle automatique d'objets de forme complexe ou non-complexe à partir de points 3D et d'un modèle géométrique.

La stratégie d'inspection que nous avons préconisée est la suivante :

- Vérification des tolérances globales :
 - * Tolérances de positionnement et d'orientation de chaque surface,
 - * Tolérances de forme globale de chaque surface.
- Vérification des tolérances locales
 - * Tolérance sur la normale,
 - * Tolérance des courbures.
- Localisation des erreurs de fabrication, mises en évidence par différents procédés de visualisation :
 - * Segments colorés,
 - * Triangles colorés,
 - * Texture Mapping.
- Comparaison sur une pièce réelle, à partir de données 3D provenant d'une machine à mesurer et d'un capteur télémétrique, de ces deux procédés d'acquisition.

Les algorithmes présentés dans ce mémoire, programmés en C++, constituent deux programmes : Un programme de mise en correspondance entre un modèle CAO STL et/ou un modèle CAO NURBS découpées et un ensemble de points 3D et un programme d'inspection à partir des points 3D mis en correspondance avec le modèle CAO.

En conclusion, les travaux que nous avons menés nous permettent de dire que :

- La méthode de mise en correspondance que nous avons développée présente une erreur calculatoire de $0,7 \mu\text{m}$. L'erreur qui prédomine est l'erreur de modélisation STL, qu'il convient de minimiser pour obtenir une meilleure précision finale (fonction de l'application et de l'erreur d'acquisition des points 3D).
- A partir de points 3D provenant d'une machine à mesurer, nous pouvons déceler des défauts de fabrication de l'ordre du centième de mm. Ils sont du dixième de mm, lorsque les points proviennent d'un capteur télémétrique.

Les perspectives de notre travail sont les suivantes :

- Réduction du temps de calcul de la méthode de mise en correspondance, afin que celle-ci puisse être raisonnablement intégrée dans un concept manufacturier. L'objectif de notre travail était avant tout, de démontrer le bon fonctionnement de la méthode que nous avons proposée, nous avons montré qu'il est possible de diminuer les temps de calcul, mais n'avons pas eu le temps d'implémenter les solutions.
- Améliorer la stratégie d'inspection,
- Concept de "station d'inspection intégrée" : Ceci consiste à intégrer ensemble le capteur laser 3D, la base de données CAO et un calculateur contenant les programmes de mise en correspondance, ainsi qu'un écran permettant de visualiser graphiquement les erreurs des objets inspectés.

Annexe A

A.I. Représentation d'une transformation rigide 3D

Voici les types de représentation d'une transformation rigide les plus utilisés :

A.I.1. Matrice de rotation et vecteur translation

L'équation $p \oplus R.p + T$ peut être développée :

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \oplus \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \cdot \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

où R doit être orthonormale, son déterminant doit être égal à 1.

L'intérêt d'utiliser cette représentation est que les matrices 3×3 sont très faciles à manipuler et que la composition de plusieurs rotations est obtenue par simple multiplication de leurs matrices.

A.I.2. Axe et angle de rotation

Les éléments de la matrice de rotation peuvent être exprimés sous forme d'un axe de rotation \vec{n} et d'un angle de rotation θ , ou $\vec{n}(n_1, n_2, n_3)$ est un vecteur unitaire (figure A.I.1):

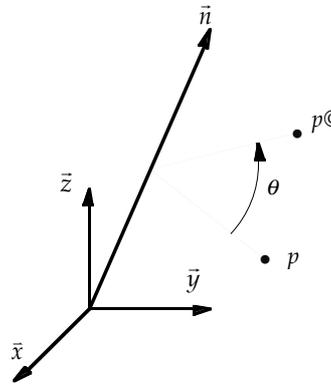


Figure A.I.1 : Transformation d'un point p en un point p' par la rotation d'angle θ autour de l'axe unitaire \vec{n} .

$$R = \begin{bmatrix} n_1^2 + (1 - n_1^2) \cos \theta & n_1 n_2 (1 - \cos \theta) + n_3 \sin \theta & n_1 n_3 (1 - \cos \theta) - n_2 \sin \theta \\ n_1 n_2 (1 - \cos \theta) - n_3 \sin \theta & n_2^2 + (1 - n_2^2) \cos \theta & n_2 n_3 (1 - \cos \theta) + n_1 \sin \theta \\ n_1 n_3 (1 - \cos \theta) - n_2 \sin \theta & n_2 n_3 (1 - \cos \theta) - n_1 \sin \theta & n_3^2 + (1 - n_3^2) \cos \theta \end{bmatrix}$$

Cette représentation ne permet pas de combiner simplement plusieurs rotations, bien qu'elle donne une idée plus intuitive de la transformation. Par ailleurs la rotation $(-\vec{n}, -\theta)$ est identique à (\vec{n}, θ) et dans le cas où $\theta = 0$, l'axe \vec{n} est indéterminé, ce qui pose des problèmes au niveau des calculs.

A.I.3. Angles d'Euler

La matrice de rotation peut également être exprimée à partir des angles d'Euler : θ, Φ, Ψ (figure A.I.2):

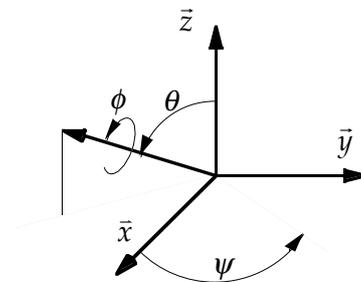


Figure A.I.2 : Définition des angles θ, Φ, Ψ

Dans ce cas :

$$R = \begin{bmatrix} \cos \Phi \cos \Psi + \sin \theta \sin \Phi \sin \Psi & -\cos \Phi \sin \Psi + \sin \theta \sin \Phi \cos \Psi & \sin \Phi \cos \theta \\ \cos \theta \sin \Psi & \cos \theta \cos \Psi & -\sin \theta \\ -\sin \Phi \cos \Psi + \sin \theta \sin \Phi \sin \Psi & \sin \Phi \sin \Psi + \sin \theta \cos \Phi \cos \Psi & \cos \Phi \cos \theta \end{bmatrix}$$

Il est montré que lorsque $\theta = \pm \frac{\pi}{2}$ les angles d'Euler ne peuvent être déterminés de façon unique et que les compositions de rotations ne sont pas simples à manipuler.

A.I.4. Représentation par quaternions

Un quaternion-unitaire est un vecteur à 4 dimensions :

$$q = (q_0, q_1, q_2, q_3) \text{ ou } q_0 \geq 0 \text{ et } q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$$

$$q = \begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right)n_1 \\ \sin\left(\frac{\theta}{2}\right)n_2 \\ \sin\left(\frac{\theta}{2}\right)n_3 \end{pmatrix} \quad R = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_3q_2 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_3q_2 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}$$

Une composition de rotations s'établit par multiplication des deux quaternions les représentant.

A.I.5. Décomposition en paramètres de vissage

Les six paramètres de vissage sont représentés (figure A.I.3) par $s_x, s_y, s_z, s_{ox}, s_{oy}, \theta$ et d où :

- $\begin{cases} s_x \\ s_y \\ s_z \end{cases}$ représente le vecteur-unitaire \mathcal{S} dans la direction de vissage,
- $\mathcal{S}'_o \begin{cases} s_{ox} \\ s_{oy} \\ s_{oz} \end{cases}$ est le vecteur construit de l'origine à l'axe de vissage et qui lui est orthogonal,
- θ est l'angle de vissage autour de \mathcal{S} ,
- d est la composante de translation le long de \mathcal{S} ,

La matrice de rotation dans ce type de représentation est la même que dans les autres représentations.

Le vecteur-translation est défini ainsi : $T = (I - R)s_o + d.s'$

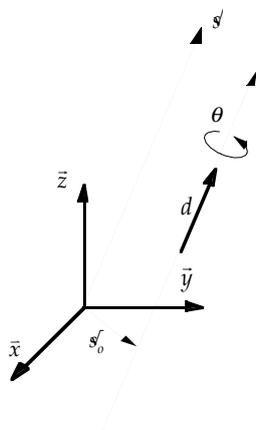


Figure A.I.3 : Paramètres de vissage

Cette représentation présente les mêmes avantages et inconvénients que la représentation par axe et angle de rotation.

A.II : Calculs de distances inter-primitives

A.II.1. Calcul de distance entre deux ensembles de points

D'une manière générale, la formule de calcul de distance entre deux ensembles de points $\{x_i\}$ et $\{x_j\}$ décrivant deux surfaces S et S' est :

$$C(R, t) = \sum_{i=1}^m d^2(Rx_i + t, S) + \sum_{j=1}^n d^2(R^T x_j + R^T t, S)$$

où $d(x, S)$ représente la distance d'un point x à une surface S . Cette formule est valable dans les cas où chaque point de S a un point correspondant dans S' et inversement. La formule décrite ci-dessus étant symétrique, et aucune des deux formes n'étant prédominante, afin d'économiser les calculs, la deuxième partie de la formule peut être omise.

Dans le cas où il y a recouvrement partiel des deux ensembles de points, donc dans le cas où chaque point du premier ensemble n'a pas nécessairement de point associé dans le deuxième, il est nécessaire de faire appel à un calcul de distance global, éliminant les distances locales parasites causées par ces absences de correspondance.

Les méthodes détaillées (ci-dessous) proposent des solutions à ce problème (choix du seuil de distance) et présentent l'avantage d'utiliser directement des ensembles de points, sans extraction de caractéristiques. Leur inconvénient général est qu'il est nécessaire de connaître une bonne estimation de la transformation initiale, et/ou de n'avoir à déterminer qu'une petite transformation. Ces méthodes sont souvent utilisées pour des applications médicales ou pour la construction de modèles 3D.

Approche de Zhang [Zhang 94]

Zhang a pris en compte le fait qu'un point d'une surface n'a pas nécessairement de correspondant sur l'autre surface en pondérant la formule de distance précédente :

$$C(R, t) = \frac{1}{\sum_{i=1}^m p_i} \sum_{i=1}^m p_i d^2(Rx_i + t, S) + \frac{1}{\sum_{j=1}^n q_j} \sum_{j=1}^n q_j d^2(R^T x_j + R^T t, S)$$

p_i (resp. q_j) prend la valeur 1 si le point x_i (resp. x_j) peut être mis en correspondance avec un point de S' (resp. S), et 0 dans le cas contraire.

L'avantage de cette méthode est que les points aberrants (apparition et disparition d'éléments, dans le cas de la navigation visuelle où les objets entrent et sortent de la scène, ou dans le cas d'occlusions), sont automatiquement éliminés.

Zhang propose une méthode robuste permettant de sélectionner les points à utiliser après appariement pour déterminer la transformation rigide, en exploitant les statistiques sur les distances et en assumant que la répartition des distances est gaussienne :

A l'itération I : $\{p_i\}$ et $\{p_i^{\odot}\}$ sont les deux ensembles de points appariés, d_i leurs distances respectives, et D_{\max}^I la distance maximum tolérée.

Soit N le nombre de points, μ la moyenne et σ l'écart-type des distances, définis par :

$$\mu = \frac{1}{N} \sum_{i=1}^N d_i \quad \sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (d_i - \mu)^2}$$

Soit D un paramètre de distance moyenne fixé par l'utilisateur indiquant si la mise en correspondance peut être considérée comme bonne. Le choix de D a une très forte influence sur la convergence de la méthode, si D est trop petit beaucoup d'itérations sont nécessaires pour converger, au contraire si la valeur de D est trop élevée la méthode n'est pas sûre de converger.

A partir des valeurs de μ et D , on peut déterminer D_{\max}^I :

Condition	Mise en correspondance	D_{\max}^I
Si $\mu < D$	Bonne	$D_{\max}^I = \mu + 3\sigma$
Si $D < \mu < 3D$	Encore bonne	$D_{\max}^I = \mu + 2\sigma$
Si $3D < \mu < 6D$	Pas trop mauvaise	$D_{\max}^I = \mu + \sigma$
Si $\mu > 6D$	Mauvaise	$D_{\max}^I = \xi$

Choix de ξ :

En exploitant l'histogramme des distances, ξ est choisi comme étant la valeur de distance correspondant à la vallée après le pic le plus élevé. Le choix de ce seuil part du principe que le pic le plus élevé correspond aux points correctement mis en correspondance, contrairement aux points situés au-delà de la vallée. Cette hypothèse n'est valable que dans le cas où une bonne estimation de la transformation initiale est connue; dans le cas contraire l'histogramme des distances est difficilement exploitable et la méthode proposée moins fiable.

Ainsi, cette méthode est efficace pour trouver la transformation entre deux ensembles de points se recouvrant partiellement, mais nécessite une bonne évaluation de la transformation initiale.

Approche de Blais [Blais 95]

Blais introduit un facteur de recouvrement Ω . Le recouvrement généré par une transformation T est défini comme étant le nombre de correspondances établies $\|S_c(T)\|$ divisé par le nombre de points de contrôle $\|S_c\|$. τ est une valeur de seuil au delà de laquelle la distance n'est pas prise en compte.

Le critère de distance C est alors défini par :

$$C = \begin{cases} \frac{\sum_{x_i} d(T(x_i), y_i)}{\|S_c(T)\|} & \text{si } \frac{\|S_c(T)\|}{\|S_c\|} \geq \Omega \\ \tau \cdot \|S_c\| & \text{si } \frac{\|S_c(T)\|}{\|S_c\|} < \Omega \end{cases}$$

Cette méthode permet de mettre en correspondance les images de profondeur d'un objet afin de construire son modèle surfacique. L'originalité de l'approche de Blais réside également dans l'établissement des correspondances entre points utilisant l'inversion du procédé de calibrage du capteur de profondeur.

Approche de Bittar [Bittar 93]

Cette approche est également utilisée pour la construction d'un modèle de surface à partir d'images de profondeur d'un objet. La fonction de distance présentée élimine les points qui n'appartiennent pas à la région partagée par deux ensembles de points qui se recouvrent.

Toutes les distances au modèle à partir des points D_i sont établies, puis triées; seules les $p\%$ meilleures distances sont conservées pour le calcul de la distance moyenne. p est choisi initialement égal à 100% et décroît de 10% à chaque itération jusqu'à la valeur minimum $p_{\min}=30\%$, pourcentage de recouvrement réel généralement constaté pour les applications étudiées. Cette méthode nécessite encore une bonne estimée de la transformation initiale.

A.II.2. Distance non-signée ou signée entre un ensemble de segments et un ensemble de points [Lavallée 95]

Afin de mettre en correspondance des images 2D obtenues en Imagerie X avec des modèles 3D, pour la chirurgie assistée par ordinateur et robotique, il est nécessaire de définir des distances entre un ensemble de segments 2D et une ensemble de points 3D.

La distance non-signée entre un point r et une surface S constituée de I points s_i est :

$$d_E(r) = \min_{s_i} d(r, s_i)$$

où d étant la distance euclidienne.

La distance non-signée entre une k -ième ligne l_k constituée de j points r_j et S est :

$$d_E(l_k) = \min_{r_j, s_i} d(r_j, s_i)$$

L'inconvénient d'utilisation de la distance non-signée est qu'une ligne qui traverse la surface aura une distance minimum nulle, bien qu'elle ne soit jamais tangente à la surface. Une distance signée $\hat{d}(r)$ entre un point r et une surface S donne une valeur négative lorsque r est à l'intérieur de la surface et positive dans le cas contraire.

A.II.3. Distance points de contrôle/plans [Chen 92]

Contrairement à d'autres techniques où toutes les données disponibles sont utilisées, ici seuls quelques points appelés points de contrôle sont pris en considération, ces points étant choisis dans les zones lisses des surfaces. Chen propose une méthode permettant de minimiser la distance entre les points de contrôle dans une vue P et les plans interpolant leurs points de contrôle associés dans une autre vue Q .

Le critère à minimiser est $e^k = \sum_{i=1}^N d_s^2(T^k p_i, S_i^k)$ où :

- T^k est la transformation 3D appliquée au point de contrôle $p_i \in P$ à l'itération k , l_i est la ligne normale à P en p_i ,
- $q_i^k = (T^k l_i) \cap Q$ est le point d'intersection de la surface Q avec la ligne transformée $T^k l_i$,
- $n_{q_i^k}^k$ est la normale à Q en q_i^k ,
- $S_i^k = \{s | n_{q_i^k}^k \cdot (q_i^k - s) = 0\}$ est le plan tangent à Q en q_i^k ,
- d_s est la distance signée d'un point à un plan.

La particularité de cette méthode est qu'elle n'utilise pas la correspondance entre points. Ses inconvénients sont qu'elle ne peut pas traiter des objets trop complexes et qu'elle nécessite une bonne estimation de la transformation initiale.

A.II.4. Calcul de distance entre triangles [Bergevin 95]

Bergevin réalise la mise en correspondance entre deux images de profondeur en représentation triangulée hiérarchique. Le calcul de distance s'effectue entre les points de contrôle de la première forme et les triangles de la deuxième. Les points de contrôle sont choisis dans un premier temps comme étant les sommets des gros triangles, puis comme étant les sommets des petits triangles proches des contours.

Annexe A.III : Estimateurs plus spécifiques

A.III.1. Moindres carrés MS [Zhuang 94]

L'estimateur moindres carrés va minimiser la somme des carrés des distances aux primitives, pondérée par le nombre de points. cet estimateur n'est absolument pas robuste à la présence de points aberrants.

$$C = \frac{1}{N} \sum_{i=1}^N \|T(p_i) - p_i^{\odot}\|^2$$

A.III.2. Estimateur LMS

Cet estimateur introduit par Rousseuw et Leroy en 1987 [Rousseuw 87] minimise la distance médiane entre les deux formes à mettre en correspondance et présente ainsi une robustesse à 50% de points aberrants.

A.III.3. Estimateur de variance minimum (MVE) [Doraï 94] [Lavallée 95]

$$e^k = \sum_{i=1}^N \frac{1}{\sigma_{d_s}^2} d_s^2$$

ou $\sigma_{d_s}^2$ est la variance du bruit de mesure estimée à une distance d_s . L'un des avantages de cet estimateur est qu'il n'est pas nécessaire de connaître l'exacte distribution du bruit. Par contre, il est nécessaire de connaître sont les statistiques du deuxième ordre de la distribution du bruit, qui peuvent pratiquement être facilement évalués.

A.III.4. Estimateur MF prise en compte des erreurs de mesure [Zhang 94]

Les techniques robustes se caractérisent par leur point d'arrêt ("Break-Point"), qui est le pourcentage maximum de points aberrants qui peut être toléré sans corrompre le résultat de la transformation recherchée. Dans le cas d'un estimateur aux moindres carrés, ce point d'arrêt est à 0%, et dans le cas d'un estimateur médian à 50%.

Les techniques de vision robustes peuvent nécessiter des estimateurs plus robustes, ayant un point d'arrêt supérieur à 50 %, lorsque :

- plus de 50 % des données sont mauvaises,
- les données appartiennent à plusieurs surfaces et que moins de 50% des points appartiennent à la surface d'intérêt,
- il y a superposition de surfaces.

L'estimateur introduit par Zhang présente l'avantage d'être hautement robuste, non affecté par la présence d'une large proportion de points aberrants ou de correspondances erronées, tant qu'il y a suffisamment de points non-aberrants pour fournir une solution correcte. Par ailleurs, il est indépendant des recherches initiales, sa complexité de calcul est raisonnable et facilement parallélisable. Aucun estimateur global ne peut avoir un point d'arrêt supérieur à 50 %. Afin d'obtenir un point d'arrêt supérieur, la seule possibilité est de réaliser une modélisation partielle.

La mise en correspondance consiste à déterminer les sept paramètres suivants : 6 paramètres de transformation rigide β classiques + un paramètre supplémentaire de variance gaussienne σ^2 .

Le critère à minimiser devient alors le suivant :

$$\max_{\beta, \sigma} q(\beta, \sigma; t) = \sum_{n=1}^N \log \left\{ \frac{1}{(\sqrt{2\pi}\sigma)^3} \exp\left(-\frac{\|r_n\|^2}{2\sigma^2}\right) + t \right\} \quad \text{avec } r_n = y_n - (R \cdot x_n + T)$$

où r_n est l'erreur de la paire $\{x_n, y_n\}$ et où t choisit un modèle partiel et partage l'ensemble des points en points aberrants et non-aberrants.

Pour chaque t , après avoir résolu $\max_{\beta, \sigma} q(\beta, \sigma; t)$ pour $R(t), T(t), \sigma(t)$, un sous-ensemble $G(t)$ est construit :

$$G(t) = \left\{ (x_n, y_n) : \frac{1}{(\sqrt{2\pi}\sigma)^3} \exp\left(-\frac{\|r_n\|^2}{2\sigma^2}\right) > t \right\}$$

le test de normalité est appliqué sur l'ensemble des résidus des points non-aberrants :

$$G_0(t) = \{r_n : r_n = y_n - (R(t) \cdot x_n + T(t)), (x_n, y_n) \in G(t)\}$$

La maximisation s'arrête pour t^* lorsque le test de normalité pour $G_0(t^*)$ est passée ; la transformation recherchée est alors $R(t^*), T(t^*)$.

A.III.5. Estimateur MINPRAN [Stewart 95]

Comme l'estimateur précédent, l'estimateur MINPRAN (MINimizing the probability of RANdomness) est capable de trouver la transformation même lorsqu'il y a plus de 50% de points aberrants. Les points aberrants sont distribués aléatoirement selon le bruit du capteur. Dans ce cas, le critère à minimiser (en deux dimensions) est $H(\phi, N) = \min_r F(r, k_{\phi, r}, N)$, où ϕ est le vecteur-paramètres, $\{\phi - r, \phi + r\}$ est l'intervalle dans lequel le point est jugé non-aberrant, N le nombre de points donnés, H est appelé "Probability of randomness".

Annexe B

Méthodes de Mise en Correspondance 3D

B.1. Méthodes avec correspondance de points 3D

Toutes ces méthodes s'appliquent à deux ensembles de points/primitives appairés et de même taille. Nous présentons dans un premier temps des méthodes non-linéaires simples, mais souvent mal adaptées au problème traité. Suivent ensuite les méthodes d'optimisation non-linéaires, puis des méthodes plus appropriées à la mise en correspondance de nuages de points 3D.

Méthodes linéaires directes [Sabata 91]

Il est possible de déterminer directement les paramètres de transformation en résolvant le système d'équations linéaires $M = R.D + T$. On dénombre alors 12 variables (9 de rotation et 3 de translation), qui peuvent être déterminées par la connaissance de quatre points indépendants.

Cette méthode n'est pas envisageable, car elle n'utilise qu'une partie des données disponibles et, comme celles-ci sont généralement bruitées, la transformation obtenue serait alors peu crédible.

Méthode basée sur les invariants en translation [Sabata 91]

Du fait que les vecteurs sont invariants en translation, la composante de rotation de la transformation peut être calculée entre deux vecteurs appairés de D et M. Comme pour la méthode précédente, seulement une petite partie des données est utilisée et le résultat demeure extrêmement sensible au bruit.

Détermination de transformation à partir des coordonnées homogènes (Aggarwal, Magee - 1986) [Sabata 91]

En coordonnées homogènes, les points de données p_i et les points de modèles $p_i^{\mathbb{C}}$ s'expriment :

$$p_i = \begin{pmatrix} x_i \\ y_i \\ z_i \\ 1 \end{pmatrix} \quad \text{et} \quad p_i^{\mathbb{C}} = \begin{pmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{pmatrix}$$

La matrice de transformation T peut être déterminée par une simple inversion de matrice : $T = sS^{-1}$, avec :

$$S = \begin{pmatrix} \sum_{i=1}^n X_i^2 & \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n X_i Z_i & \sum_{i=1}^n X_i \\ \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n Y_i^2 & \sum_{i=1}^n Y_i Z_i & \sum_{i=1}^n Y_i \\ \sum_{i=1}^n X_i Z_i & \sum_{i=1}^n Y_i Z_i & \sum_{i=1}^n Z_i^2 & \sum_{i=1}^n Z_i \\ \sum_{i=1}^n X_i & \sum_{i=1}^n Y_i & \sum_{i=1}^n Z_i & n \end{pmatrix} \quad s = \begin{pmatrix} \sum_{i=1}^n X_i x_i & \sum_{i=1}^n Y_i x_i & \sum_{i=1}^n Z_i x_i & \sum_{i=1}^n x_i \\ \sum_{i=1}^n X_i y_i & \sum_{i=1}^n Y_i y_i & \sum_{i=1}^n Z_i y_i & \sum_{i=1}^n y_i \\ \sum_{i=1}^n X_i z_i & \sum_{i=1}^n Y_i z_i & \sum_{i=1}^n Z_i z_i & \sum_{i=1}^n z_i \\ \sum_{i=1}^n X_i & \sum_{i=1}^n Y_i & \sum_{i=1}^n Z_i & n \end{pmatrix}$$

Comme il n'existe aucune contrainte sur les paramètres de la transformation, la transformation trouvée n'est la plupart du temps pas rigide.

Méthode de projection sphérique adaptée [Sabata 91]

La rotation est représentée par l'axe \vec{n} et l'angle de rotation θ . A l'aide de trois vecteurs non-colinéaires $\vec{m}_1, \vec{m}_2, \vec{m}_3$ construits à partir de trois points de D et des trois vecteurs correspondants dans M ($\vec{m}_1^{\mathbb{C}}, \vec{m}_2^{\mathbb{C}}, \vec{m}_3^{\mathbb{C}}$), on peut déterminer les paramètres de la transformation :

$$\vec{n} = (\vec{m}_2^{\mathbb{C}} - \vec{m}_1) \times (\vec{m}_3^{\mathbb{C}} - \vec{m}_1) \quad \text{et} \quad \cos \theta = \frac{(k_2 - k_1)}{(1 - k_1)} \quad \text{avec} \quad k_1 = (\vec{n} \cdot \vec{m}_3^{\mathbb{C}})^2 \quad \text{et} \quad k_2 = \vec{m}_3 \cdot \vec{m}_3^{\mathbb{C}}$$

Cette procédure n'utilisant que trois points de D et les trois points correspondants dans M n'est pas robuste et est très affectée par la présence de bruit sur les points. Par ailleurs cette méthode est très instable autour des points singuliers liés à la représentation des rotations.

Transformation à partir de la décomposition en vissage (Blostein et Huang - 1984)

En sélectionnant trois points p, q et r de D et leurs points associés dans M (p', q' , et r'), les paramètres de la transformation en vissage sont :

$$\tan\left(\frac{\theta}{2}\right) s' = \frac{[(r \ominus q \oplus \mathcal{Q}) - (r - q)] \times [(p \ominus q \oplus \mathcal{Q}) - (p - q)]}{[(r \ominus q \oplus \mathcal{Q}) - (r - q)] \cdot [(p \ominus q \oplus \mathcal{Q}) - (p - q)]} \quad s'_0 = \frac{1}{2} \left[p + p \ominus \frac{(s' \times (p \ominus p))}{\tan\left(\frac{\theta}{2}\right)} - s' \cdot (p \ominus p) s' \right]$$

$$d = s' \cdot (p - p \oplus \mathcal{Q})$$

Cette méthode n'est toujours pas robuste, du fait de l'utilisation d'une petite partie des données, de son comportement numérique instable suivant θ et de sa grande sensibilité au bruit.

Moindres carrés récursifs [Faugeras 87] [Chen 92] [Sabata 91]

Le critère à minimiser est le suivant :

$$C = \frac{1}{N} \sum_{i=1}^N \|T(p_i) - p'_i\|^2$$

Si p est le centre de gravité des $\{p_i\}$ et p' le centre de gravité des $\{p'_i\}$, la rotation et la translation peuvent être découplés. L'erreur à minimiser devient :

$$E = \sum_{i=1}^n \|(p'_i - p'_i \oplus \mathcal{Q}) - R \cdot (p_i - p)\|^2$$

La composante translation idéale est : $T = p'_i \oplus R \cdot p$. Pour la rotation, il faut fixer deux des trois angles et minimiser le critère C en fonction du troisième.

Cet algorithme utilise toutes les données disponibles, par contre il n'est pas sûr que la solution obtenue soit la solution optimale, à cause de la sensibilité de cette méthode au moindre point aberrant.

Méthodes d'optimisation non-linéaire classiques

Toutes les techniques d'optimisation non-linéaires peuvent être utilisées ; en général, il faut une transformation initiale proche de la transformation idéale et la méthode n'est pas sûre de converger vers un minimum global. Ces méthodes d'optimisation non-linéaire peuvent être classées en deux catégories, selon qu'elle requiert ou non la connaissance de la dérivée de la fonction-distance à minimiser, auquel cas elle sont bien plus efficaces.

Quelques méthodes d'optimisation non-linéaires classiques sans contrainte :

- Méthodes de gradient conjugué [Press 92]

Celles-ci requiert un calcul de dérivée de la fonction à minimiser

- Méthode du Simplex [Press 92]

Elle ne nécessite pas la connaissance de la dérivée. Cette méthode est plutôt rapide pour des problèmes peu complexes.

- Levenberg Marquadt [Lavallée 95] [bittar 93],

Minimisation aux moindres-carrés non-linéaire rapide et robuste.

- Very Fast Simulated Annealing [Blais 95]

Seule méthode d'optimisation non-linéaire évitant les minimums locaux, mais coûteuse en temps de calcul.

- ...

Méthodes d'optimisation non-linéaires adaptées au problème de mise en correspondance

Pour le problème très particulier de la mise en correspondance, il existe des techniques plus appropriées que les méthodes précédemment citées :

- Méthode des quaternions (Faugeras et Hebert - 1986, Horn - 1987)

Voir paragraphe IV.4

- Décomposition en valeurs singulières (SVD) (Arun - 1987)
- Méthode des quaternions doubles [Faugeras 87] [Zhang 94]
- Utilisation de matrices orthonormales [Sabata 91]

- Réseaux de neurones, Réseaux de Kohonen [Wirtz 91]

Comparaison de l'approche par réseaux de neurones avec les approches traditionnelles :

Solutions itératives : Leurs avantages sont d'être indépendantes du modèle, de ne pas nécessiter d'apprentissage ; la précision peut être ajustée par les paramètres au cas par cas,

Leur inconvénient est que le choix des conditions initiales est délicat et qu'il peut mener à la non-convergence de la méthode. La résolution requiert parfois des calculs complexes.

Réseaux de neurones : L'avantage est qu'ils garantissent toujours une solution, que les calculs sont très simples et qu'ils sont très adaptatifs.

L'inconvénient est la longue phase d'apprentissage pour chaque modèle et que la précision souhaitée est fonction du nombre de neurones.

B.2. Méthodes sans correspondance de points

Ces méthodes partent du principe que l'on dispose de deux ensembles de points non appariés $\{p_i\}$ et $\{p_j\}$, en supposant que les mêmes points sont visibles dans les deux ensembles. Le vecteur translation se construit ainsi entre les centres de gravité des deux ensembles.

La rotation peut être trouvée à partir de deux vecteurs non-parallèles $\{n, v\}$ et les vecteurs correspondants dans le deuxième ensemble de points $\{n', v'\}$. Un troisième vecteur w , linéairement indépendant des deux autres peut être construit :

$$w = \frac{n \times v}{\|n \times v\|}, \quad w' = \frac{n' \times v'}{\|n' \times v'\|} \text{ et en utilisant le fait que : } n \in R.n', \quad v \in R.v', \quad w \in R.w' \text{ et les}$$

contraintes sur R , les paramètres de la matrice de rotation peuvent être facilement trouvés.

Le problème réside donc dans la recherche des vecteurs $\{n, v\}$ et $\{n', v'\}$.

Transformation de carreaux planaires rigides (Lin - 1986)

Cette méthode consiste à faire passer un plan au travers des points de chaque ensemble de points, le premier vecteur est la normale à ce plan, le second est le vecteur de direction moyenne \vec{M} , obtenu en sommant tous les vecteurs unitaires construits entre le centre de gravité et chacun des points de l'ensemble.

Transformation de Fourier (Lin - 1986)

Cette méthode se base sur le fait qu'une fonction définie sur un ensemble de points et sa transformée de Fourier décrivent la même rotation. Les points sont d'abord exprimés dans un repère local centré sur leur centre de gravité :

$$q_{1i} = p_i - \frac{1}{N} \sum_{k=1}^N p_k \quad q_{2i} = p_i^{\odot} - \frac{1}{N} \sum_{k=1}^N p_k^{\odot}$$

Ainsi : $q_{2i} = R \cdot q_{1i}$

Soient les fonctions suivantes : $g_1(q) = \sum_{i=1}^N \delta(q - q_{1i})$, $g_2(q) = \sum_{i=1}^N \delta(q - q_{2i})$

qui ont pour transformées de Fourier :

$$G_1(f) = \sum_{i=1}^N \exp(-j2\pi f^T q_{1i}) \quad G_2(f) = \sum_{i=1}^N \exp(-j2\pi f^T q_{2i})$$

La recherche de l'axe de rotation part du fait que la valeur de la transformée de Fourier est identique tout le long de l'axe de rotation. Une fonction de coût prenant en compte le cosinus directeur de l'axe de rotation est utilisée.

Pour l'angle de rotation , une autre fonction de coût basée sur la corrélation entre les deux transformées de Fourier est évaluée, le maximum de cette fonction de coût donne l'angle optimal.

B.3. Méthode avec correspondance de primitives

Afin de résoudre le problème de mise en correspondance, il est souvent nécessaire d'établir une correspondance entre caractéristiques de D / caractéristiques de M.

Une mesure de qualité $p(M_i, D_i)$, ainsi qu'une mesure de cohérence du n-uplet $c(M_1, D_1, \dots, M_n, D_n)$, sont réalisées.

Techniques de mise en correspondance pouvant être utilisées :

- Techniques de relaxation : leur but est de réduire itérativement l'ambiguïté et l'incohérence des correspondances initiales. Il existe deux types de techniques de relaxation :
 - Techniques de relaxation discrètes : p et c ne peuvent être égaux qu'à 1 ou 0
 - Techniques de relaxation continues : p et c peuvent prendre des valeurs réelles ; $p(M_i, D_i)$ équivaut à la probabilité que M_i corresponde à D_i .
- Transformée de Hough

- Recherche par analyse arborescente

Nous présentons deux types de méthodes utilisant les correspondances entre primitives : méthode de mise en correspondance de surfaces et méthodes de mise en correspondance de représentations sphériques.

B.3.1. Méthodes avec correspondance de surfaces

Les points et lignes sont des caractéristiques locales qui sont très sensibles au bruit et aux erreurs de quantification. Par ailleurs, établir des correspondances entre points lorsqu'il y a des occlusions dans la scène est délicat ; de plus, il peut y avoir des erreurs de correspondance (bord de cylindre, ...). Utiliser les surfaces pour la mise en correspondance permet d'éviter les problèmes mentionnés ci-dessus.

Utilisation des quaternions (Faugeras et Hebert - 1983)

Chaque forme à mettre en correspondance est un ensemble de carreaux plans, où chaque plan est défini par sa normale et sa distance à l'origine. $P_i(\vec{n}_i, d_i)$ et $P_{i'}(\vec{n}_{i'}, d_{i'})$ sont respectivement le premier et le deuxième ensemble.

Le critère à minimiser est :
$$\text{Min} \left(\sum_{i=1}^N \|\vec{n}_{i'} - R \cdot \vec{n}_i\|^2 \right)$$

La représentation de la rotation se fait par quaternion ; il faut donc retrouver le quaternion qui minimise le critère ci-dessus.

Utilisation des centres de gravité (Kehtarnavaz et Mohan - 1989)

On dispose de deux ensembles de surfaces, extraites d'images de profondeur ; dans cette méthode, seuls les centres de gravité de chaque surface sont considérés, on retrouve ainsi la mise en correspondance de deux ensembles de points. Cette méthode allie à la fois la simplicité de la représentation des surfaces par un ensemble de points et la robustesse de l'approche par surfaces. En revanche elle est sensible aux occlusions et n'impose aucune contrainte de rigidité sur la transformation recherchée.

Optimisation génétique, algorithme génétique canonique (cGA) [Jacq 95]

L'algorithme cGA utilise une fonction de distance stochastique, appliquée sur des sous-ensembles extraits aléatoirement de l'ensemble de données. Une technique adaptative de division de l'espace de recherche est utilisée avec des activations successives de l'algorithme cGA, ce qui diminue la complexité de l'algorithme global et procure une bonne fiabilité à la transformation finale trouvée.

B.3.2. Mise en correspondance de représentations sphériques [Higuchi 94]

Higuchi réalise la mise en correspondance à partir des représentations sphériques de différentes vues de l'objet.

La représentation sphérique est obtenue en plaçant sur l'objet un maillage élastique. En chaque noeud, est calculée la variation de courbure gaussienne appelé angle simplex (SAI), ces informations sont rapportées sur une sphère-unité. La SAI a l'avantage de pouvoir représenter des objets non-convexes. La transformation est obtenue en comparant les angles simplex en chaque noeud de la sphère-unité. Il n'y a pas de transformation initiale à estimer ; la méthode est déterministe et elle garantit de trouver la meilleure représentation. Par ailleurs, il n'est pas nécessaire d'extraire des caractéristiques, ni de les mettre en correspondance.

La recherche de transformation s'effectue en utilisant les axes principaux des deux sphères, par discrétisation grossière puis locale pour trouver le minimum global.

B.3.3. Méthodes avec correspondance de primitives 3D à partir de données 2D

Ce type de méthode peut être divisé en deux parties : une pour reconstruire la troisième dimension à partir des données 2D et une de recherche de transformation. Des caractéristiques locales, telles que des points ou des lignes, utilisées pour la reconstruction 3D, sont également utilisées pour la mise en correspondance.

Moindres carrés itératifs (Blostein et Huang - 1984) [Sabata 91]

Dans une première partie, les coordonnées 3D des points caractéristiques de la scène sont calculées. Si les m_i et m_i^c sont les points associés, l'erreur $E_i = \|m_i^c - Rm_i\|$ peut être calculée. A chaque itération, la contribution de l'observation pour l'estimation de la rotation est pondérée avec l'inverse de l'erreur de l'observation. Dans cette méthode toutes les données disponibles sont utilisées, mais les contraintes de transformation rigide ne sont pas respectées. Les résultats montrent que les erreurs de reconstruction 3D priment sur la détermination de la transformation.

Transformation à partir d'une correspondance point/ligne (Kim et Aggarwal - 1987) [Sabata 91]

La reconstruction 3D s'effectue par l'utilisation de techniques de relaxation sur les points et lignes extraits des deux images 3D. La transformation est estimée à partir, soit de la correspondance entre deux lignes et un point, soit de la correspondance de trois points.

La translation est déterminée à partir des correspondances de points et la rotation en utilisant les lignes (l_i, l_j) et les lignes associées (l_i^c, l_j^c) .

$$\vec{n}_{ij} = \frac{(l_i - l_{\odot}) \times (l_j - l_{\odot})}{\|(l_i - l_{\odot}) \times (l_j - l_{\odot})\|} \text{ et } \cos \theta_{ij} = \frac{(\vec{n}_{ij} \times l_i) \cdot (\vec{n}_{ij} \times l_{\odot})}{\|\vec{n}_{ij} \times l_i\| \|\vec{n}_{ij} \times l_{\odot}\|}$$

Toutes les combinaisons de lignes (l_i, l_j) associées à (l_{\odot}, l_{\odot}) sont utilisées pour déterminer la transformation. Cette méthode utilise toutes les données disponibles et respecte les contraintes de transformation rigide.

Transformation itérative et estimation de profondeur (Kiang - 1987)

Dans les méthodes précédentes, l'erreur de reconstruction 3D est plus importante que celle introduite dans la détermination de la transformation. L'idée est donc de ne plus effectuer séquentiellement ces deux étapes, mais de les exécuter en même temps. Un modèle d'erreur 3D, sous forme d'un polyèdre d'incertitude autour de la position estimée d'un point 3D, est introduit.

Les paramètres de la transformation sont estimés utilisant une procédure aux moindres carrés pondérée, où l'erreur est donnée par :

$$E = \sum_{i=1}^N W_i ((Rp_i + T) - p_{\odot})^T ((Rp_i + T) - p_{\odot})$$

le facteur de pondération W_i , pour le i -ème point est fonction de l'inverse de la taille de sa région d'incertitude. Cette méthode demeure sensible au bruit.

B.4. Algorithme ICP (Iterative Closest Point)

Cette méthode couramment utilisée permet d'appairer les points/primitives de D avec ceux de M. Dans cet algorithme, la forme modèle (M) n'est pas nécessairement représentée par un ensemble de points. Ainsi à chaque point de l'ensemble de données (D) correspond un point (point le plus proche) dans la forme modèle. L'idée de base de l'algorithme ICP est que, dans certaines conditions, l'ensemble des points les plus proches appartenant au modèle représente une approximation raisonnable de la vraie correspondance. Besl et Mc Kay [Besl 92] ont prouvé qu'en déterminant les points les plus proches et en cherchant la transformation entre les deux ensembles de points appairés, la solution converge obligatoirement vers un minimum local.

Algorithme ICP de base

- a. Pour chaque point de D, chercher son point le plus proche dans M,
- b. Avec la correspondance établie en a, calculer la transformation qui minimise le critère de distance,

- c. Appliquer la transformation déterminée en b sur les données D ,
- d. Calculer les changements sur l'erreur globale aux moindres carrés. Si ce changement est inférieur à un certain seuil, alors arrêter, sinon aller en a .

Extension de l'algorithme ICP : Iterative Closest Compatible Point (ICCP) [Godin 94] [Godin 95]

Une fonction de compatibilité est proposée permettant d'appairer les points compatibles, c'est-à-dire les points pour lesquels la fonction de compatibilité est supérieure à une valeur seuil. L'algorithme de détermination de transformation est ainsi restreint à l'utilisation de paires de points compatibles, c'est-à-dire de même étiquette (utilisation des courbures), ou associés à des informations d'intensité compatibles.

- Utilisation des courbures [Godin 95] :

En chaque point sont calculées les courbures moyenne et gaussienne de la surface indépendamment du point de vue. Ces courbures sont obtenues en faisant passer par le point d'intérêt et ses points voisins (pondérés et compris dans une fenêtre locale) une surface polynomiale bicubique. Ceci permet d'éviter l'influence du bruit et des discontinuités de profondeur ou d'orientation sur le calcul des courbures. Les valeurs des deux courbures en un point permettent de classer ce point dans une des 8 catégories de surface locale possibles et de lui attribuer un label.

- Utilisation des informations d'intensité ou de couleurs [Godin 94] :

Ce type d'informations, dépendant des conditions de mesure telles que la distance ou l'orientation, est converti en une primitive indépendante du point de vue.

Une version robuste de cet algorithme ICCP a également été mise au point en ne considérant que les points présents dans les deux images.

Accélération de l'algorithme ICP [Besl 92]

L'algorithme ICP classique produit une succession de vecteurs à 6 dimensions (nombre de paramètres de la transformation rigide) qui définissent un chemin dans l'espace de mise en correspondance, allant de la transformation initiale (généralement identité) jusqu'à la transformation optimale locale. Afin d'accélérer l'algorithme, Besl réalise une interpolation des trois derniers points déterminés par ICP à l'aide d'une parabole, afin de réaliser une prédiction sur le point suivant.

Annexe C

Données 3D et Modèles CAO

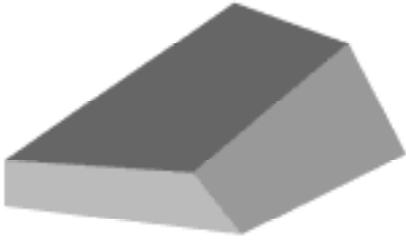
Nous allons présenter dans cette annexe toutes les données 3D et tous les modèles CAO des objets que nous avons utilisés dans cette étude. Ces objets ont été numérisés à l'aide du capteur télémétrique Hyscan 45c de la société Hymarc Ltd, les différentes vues étant fusionnées par mise en correspondance mécanique. Les modèles CAO ont été fournis au format IGES pour être convertis en modèle STL à l'aide du logiciel IGtoSTL de Brock Rooney.

Les objets utilisés peuvent être de forme simple (polyédrique, cylindrique,...), utilisés principalement pour tester les algorithmes, jusqu'à des formes complexes pour des objets d'origine industrielle, permettant d'avoir des applications plus concrètes.

Certains de ces objets ont été conçus en CAO à l'AIPRAO (Atelier Interétablissements de Productique Rhone Alpes Ouest). A partir de cette description CAO, a été généré le code de CFAO qui a servi à leur usinage sur le centre d'usinage de l'AIPRAO. D'autres objets ont été réalisés par différentes techniques de fabrication (prototypage rapide, fonderie, injection plastique,...).

Voici les différentes catégories d'objets utilisés dont les données vont être présentées dans cette annexe. Le nom attribué à chacun de ces objets figure en italique :

- Objets polyédrique (*polyèdre et plaque*)
- Objet cylindrique (*cylindre*)
- Objets cylindriques et polyédriques (*support et téléphone*)
- Objets industriels (*ims et pièce1*)



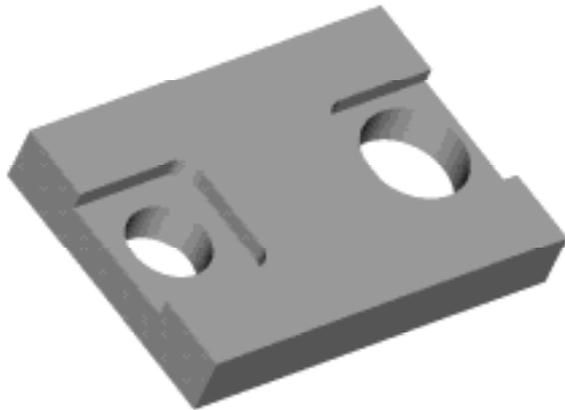
Polyèdre



Plaque



Cylindre



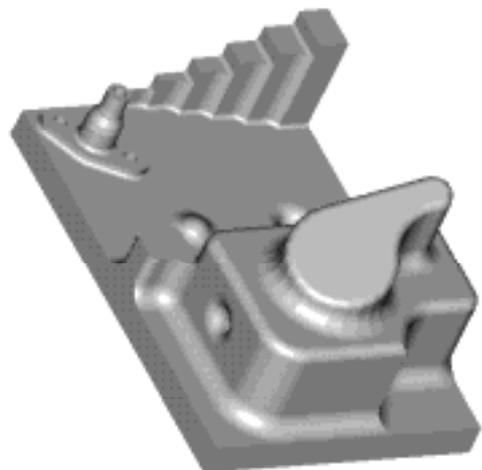
Support



Téléphone



Pièce1



Ims

C.I. Objet polyédrique N° 1 : *Polyèdre*

Cet objet est un polyèdre à 6 faces. Il a été usiné à partir de sa définition CAO à l'AIPRAO (Atelier Interétablissements de Productique Rhone-Alpes Ouest).

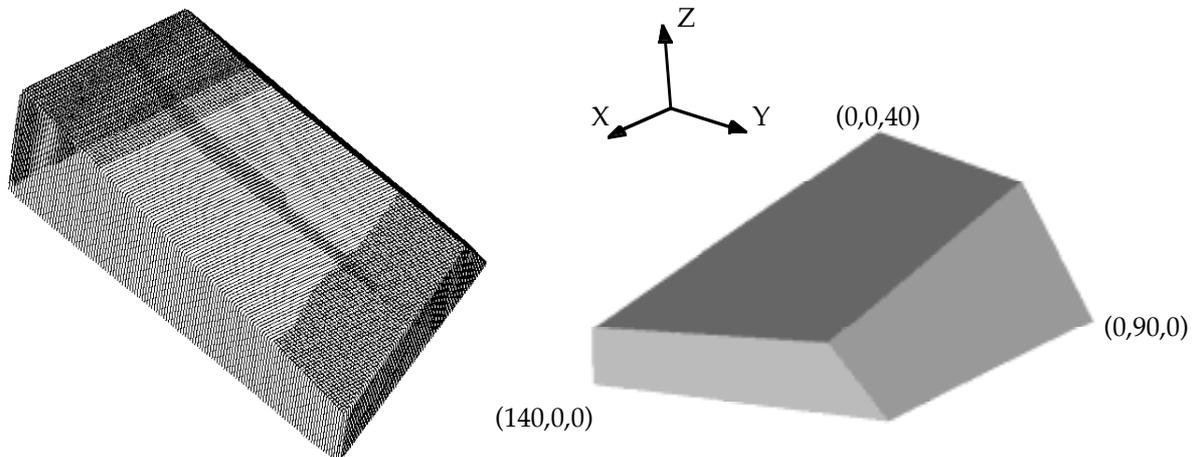


Figure C.I.1 : Ensemble de points 3D fournis par le capteur télémétrique (153291 pts) (à gauche)

Modèle CAO (à droite), les dimensions sont données en mm.

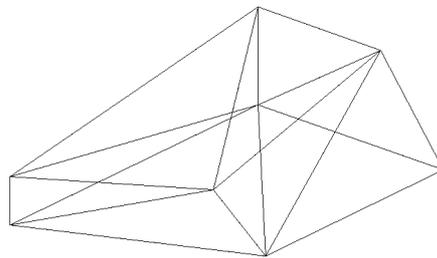


Figure C.I.2 : Triangulation STL (12 triangles).

Ce polyèdre appartient à une famille de polyèdres paramétriques, dont il est le plus petit. Ces polyèdres sont utilisés au chapitre VI, pour l'application reconnaissance de formes.

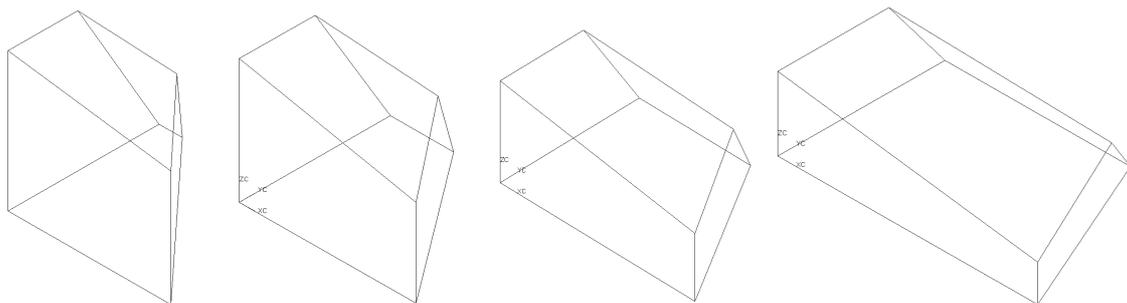


Figure C.I.3 : Famille de polyèdres paramétriques (*polyèdre1*, *polyèdre2*, *polyèdre3*, *polyèdre4*).

C.II. Objet parallélépipédique N° 2 : *Plaque*

Cet objet parallélépipédique présente un trou de forme rectangulaire ainsi qu'un trou de forme circulaire. Il a été réalisé en prototypage rapide (stéréolithographie).

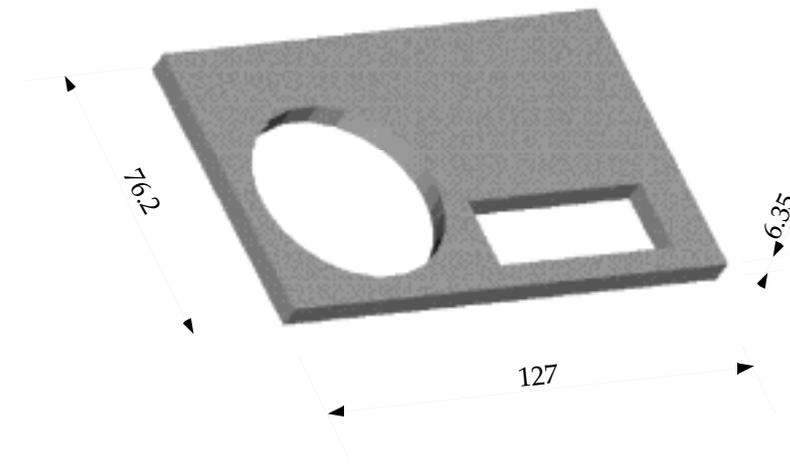


Figure C.II.1 : Modèle CAO, les dimensions sont données en mm.

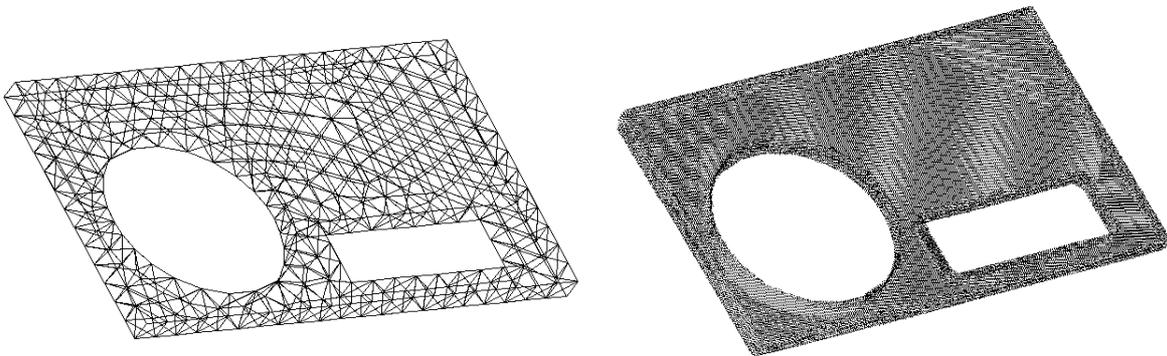


Figure C.II.2 : Triangulation STL (994 triangles) (à gauche)

Ensemble de points 3D (70117 points) (à droite).

C.III. Objet de forme cylindrique : *Cylindre*

Ce cylindre a une longueur de 100 mm pour un diamètre de 20 mm. Il a été usiné en usinage conventionnel.

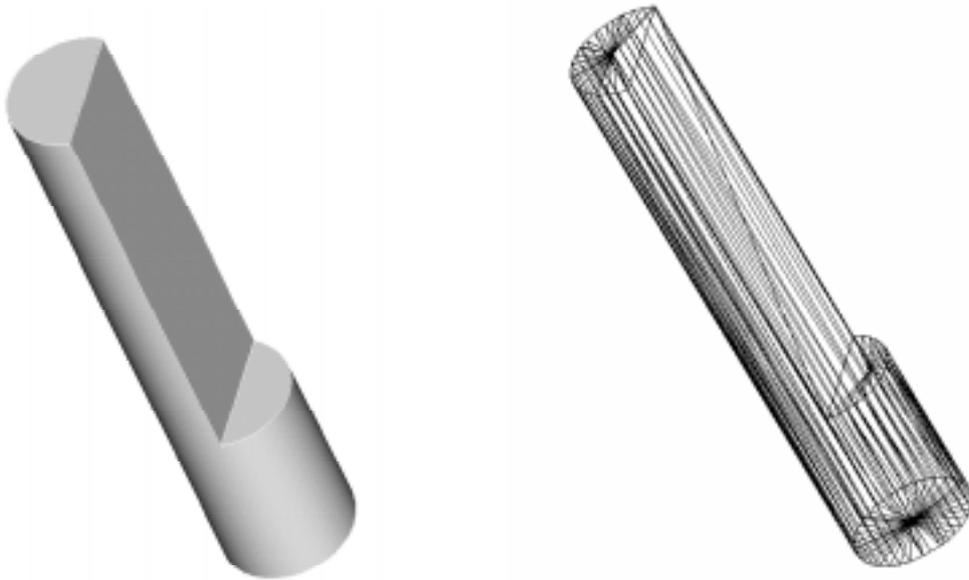


Figure C.III.1 : Modèle CAO de l'objet (à gauche)

Un exemple de triangulation STL (218 triangles)(à droite).

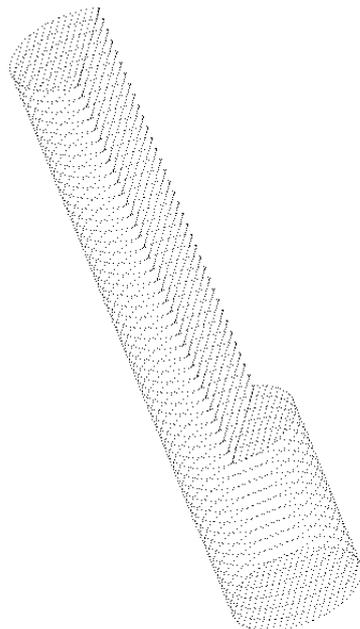


Figure C.III.2 : Ensemble de points idéaux (simulés sur la surface) (4399 points).

C.IV. Objet *Support*

Cet objet de base parallélépipédique présente deux perçages ainsi que deux lamages. Il a été conçu et usiné à l'AIPRAO.

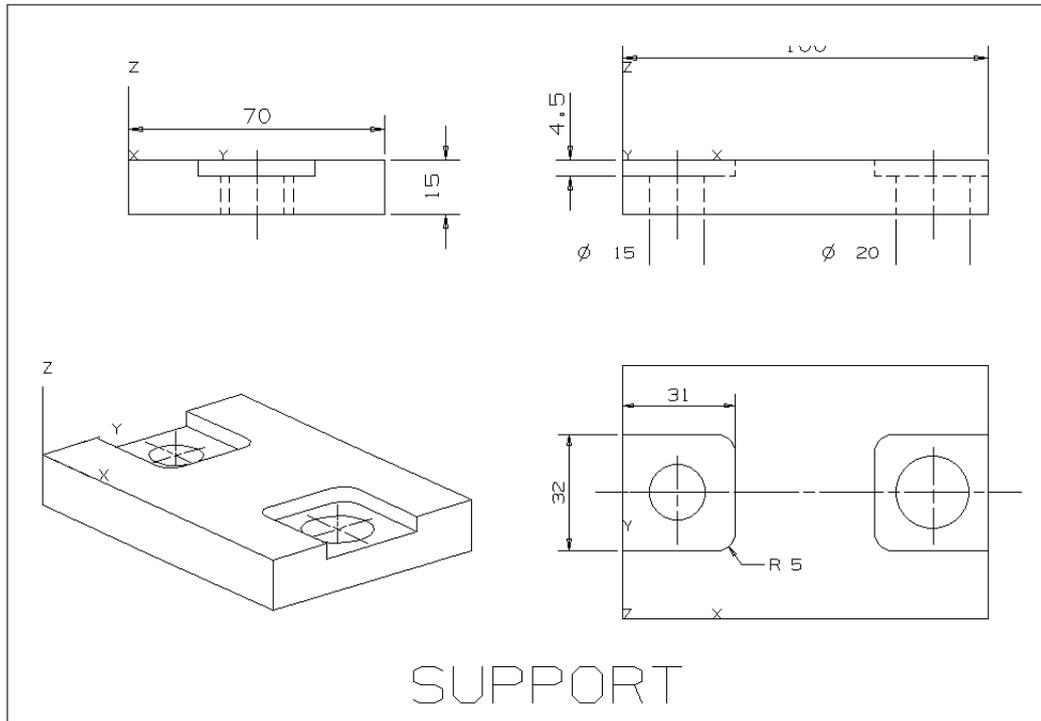


Figure C.IV.1 : Définition CAO et dimensions de l'objet *Support*

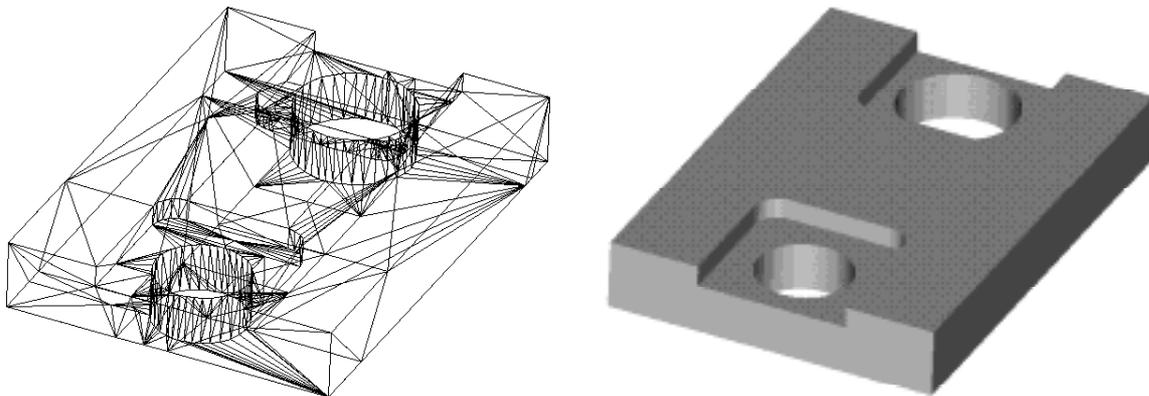


Figure C.IV.2 : Modèle CAO STL filaire (536 triangles) (à gauche)

Modèle CAO STL texturé (à droite).

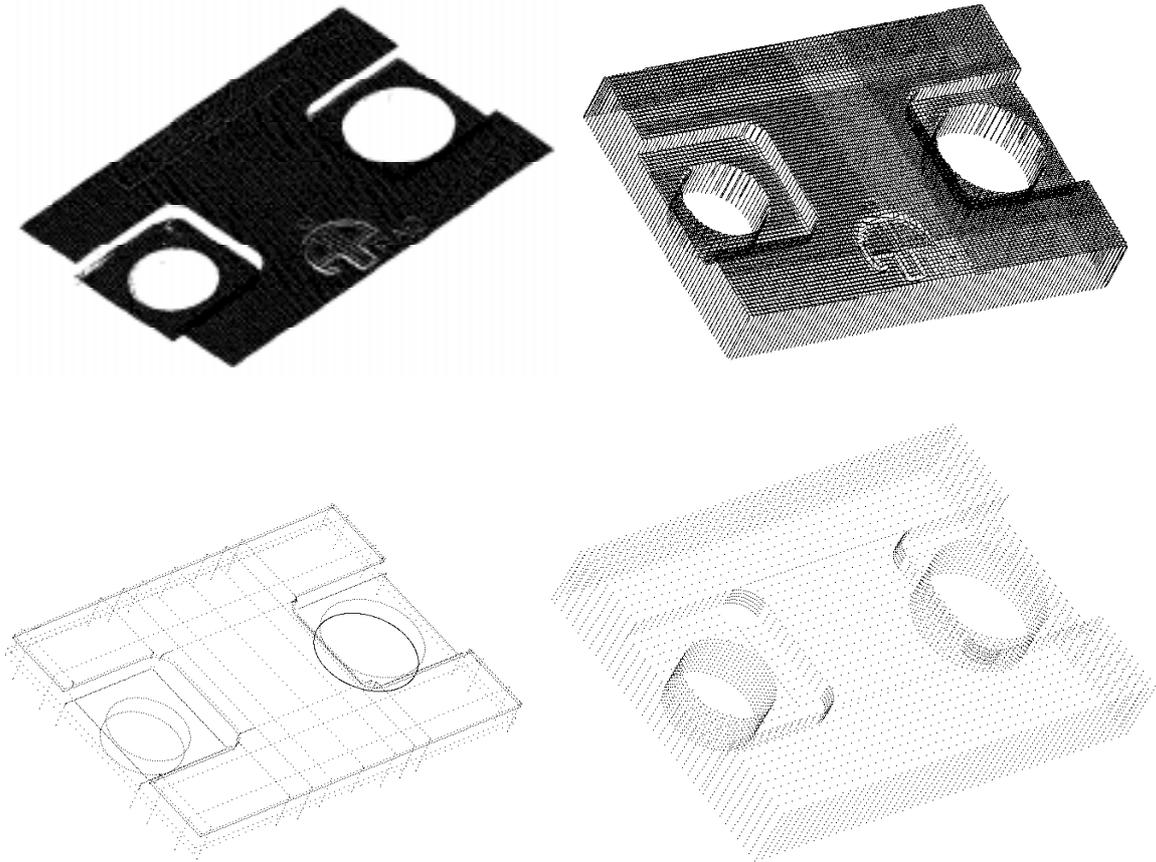


Figure C.IV.3 :Données 3D : De gauche à droite et de haut en bas :

Image 3D de l'objet réel fournis par un capteur télémétrique (5069 points)

Ensemble de points 3D de l'objet réel fournis par un capteur télémétrique (192655 points)

Ensemble de points 3D de l'objet réel fournis par une machine à mesurer mécanique (7743 points)

Ensemble de points idéaux (6713 points).

Cet objet a été utilisé au chapitre VI pour une application de reconnaissance de formes, parmi une famille de supports ou seuls les diamètres des deux perçages différent. Cette famille de supports est constituée de trois éléments comportant deux perçages de diamètres différents, parmi trois diamètres possibles (15,20 ou 25 mm)

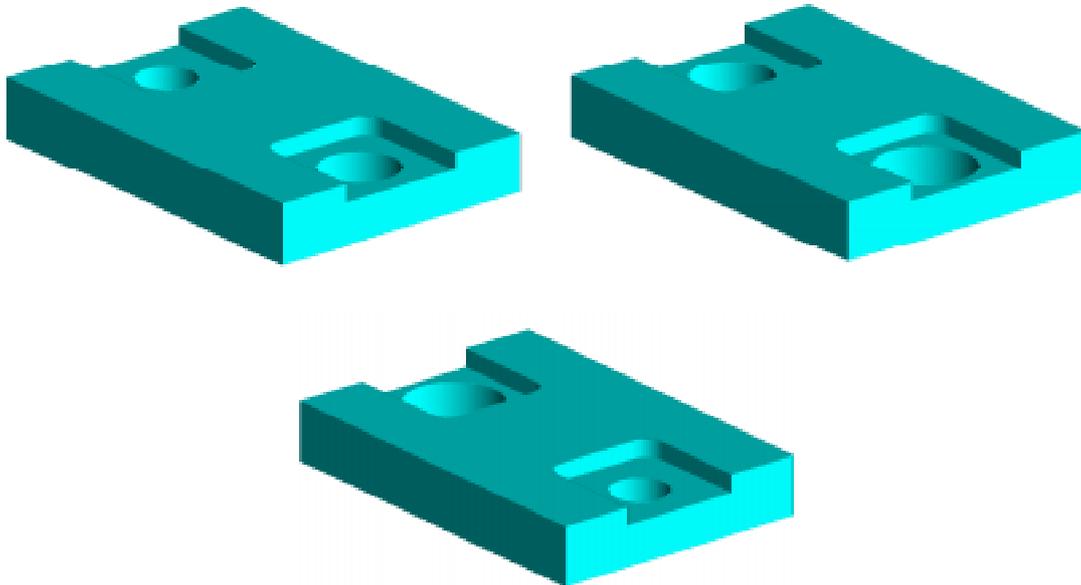


Figure C.IV.4 : Modèles CAO des trois supports : De gauche à droite et de haut en bas : *Support1*, *Support2*, *Support3*, l'objet *Support* étudié étant identique au modèle *Support2*.

C.V. Objet Téléphone

Cet objet a été conçu et usiné à l'AIPRAO.

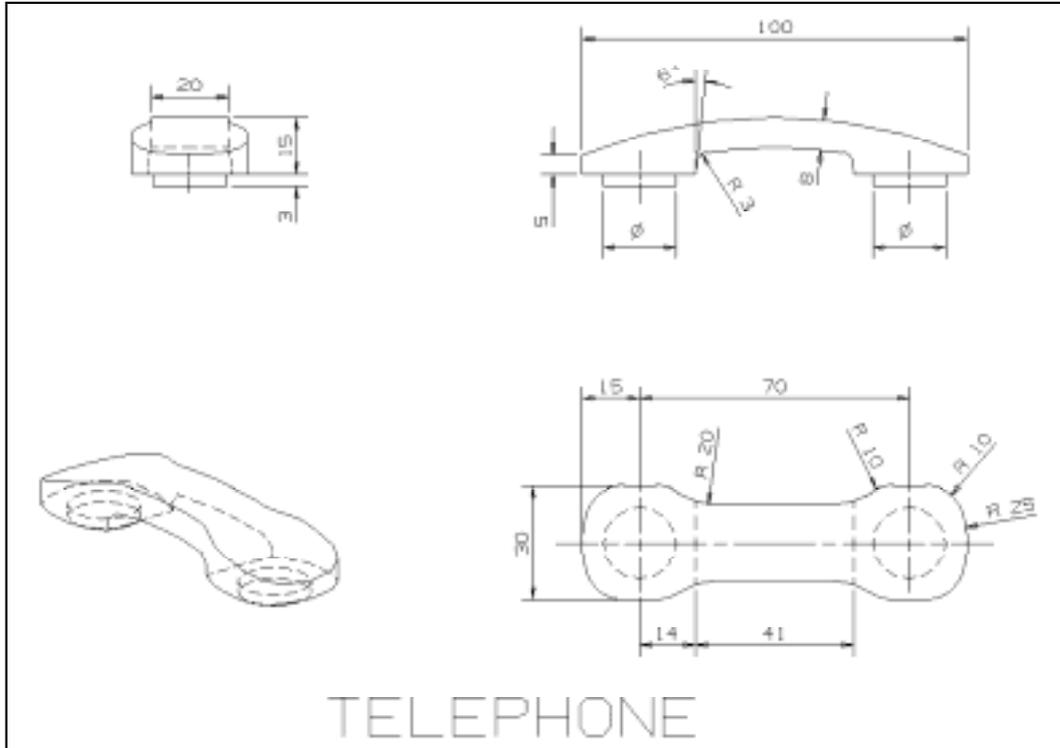


Figure C.V.1 : Modèle CAO de l'objet, les dimensions sont en mm.

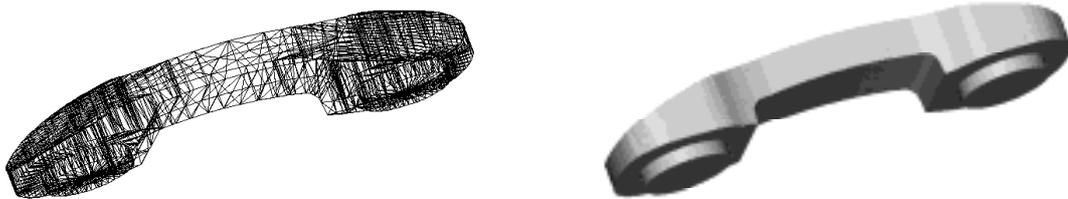


Figure C.V.2 : Modèles STL (13760 triangles) en représentation filaire (à gauche) et texturée (à droite).



Figure C.V.3 : Ensemble de points 3D de la pièce réelle fournis par un capteur télémétrique (115753 points) (à gauche), image 3D seule (5289 points).

C.VI. Objet *Pièce1*

Cet objet est d'origine industrielle à une longueur de 30 cm et a été réalisé en injection plastique.

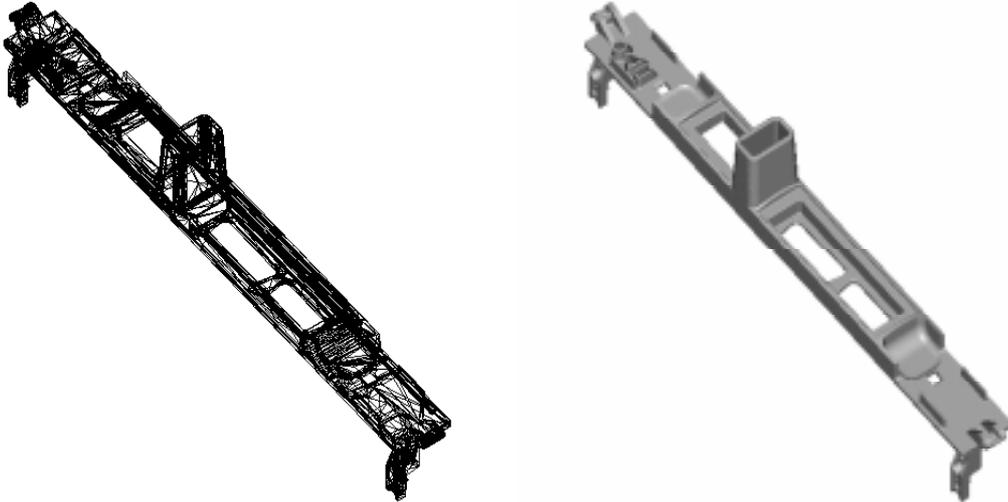


Figure C.VI.1 : Modèles STL en représentation filaire (9826 triangles) (à gauche) et en représentation texturée (à droite).

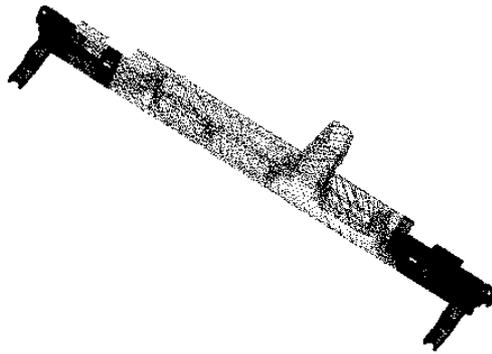


Figure C.VI.2 : Ensemble de points 3D fournis par le capteur télémétrique.

C.VII Objet IMS

Cet objet a été usiné en usinage conventionnel dans le cadre du projet international IMS (International Manufacturing System), auquel a participé le CNRC. Sa référence dans le projet IMS était IMS-TPC1.

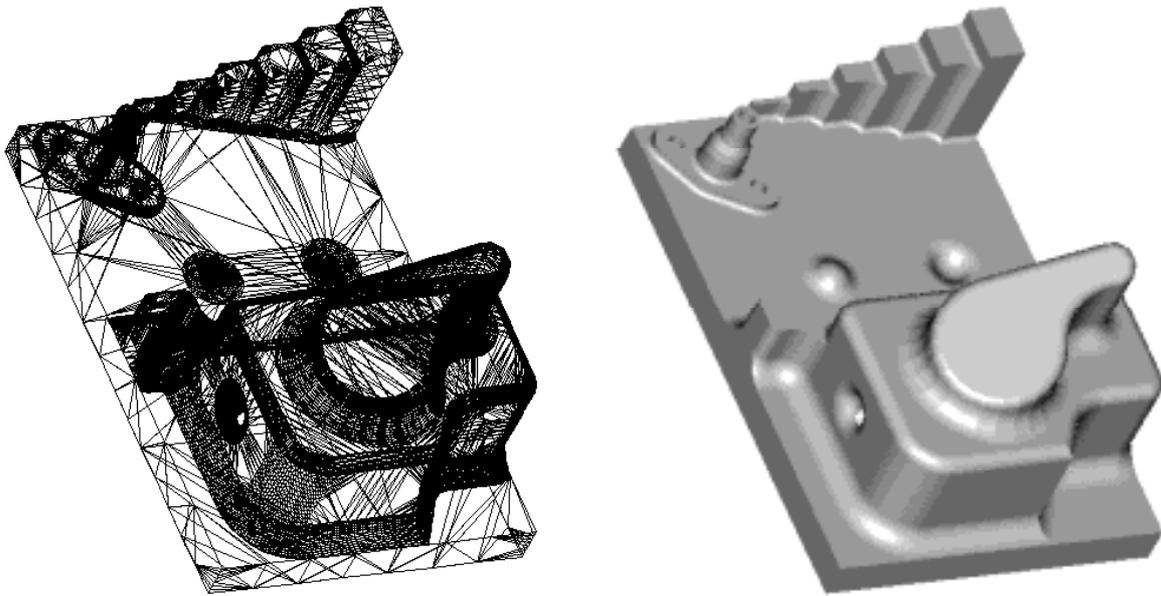


Figure C.VII.1 :Modèle STL (39544 triangles) en représentation filaire (à gauche) et texturée (à droite)

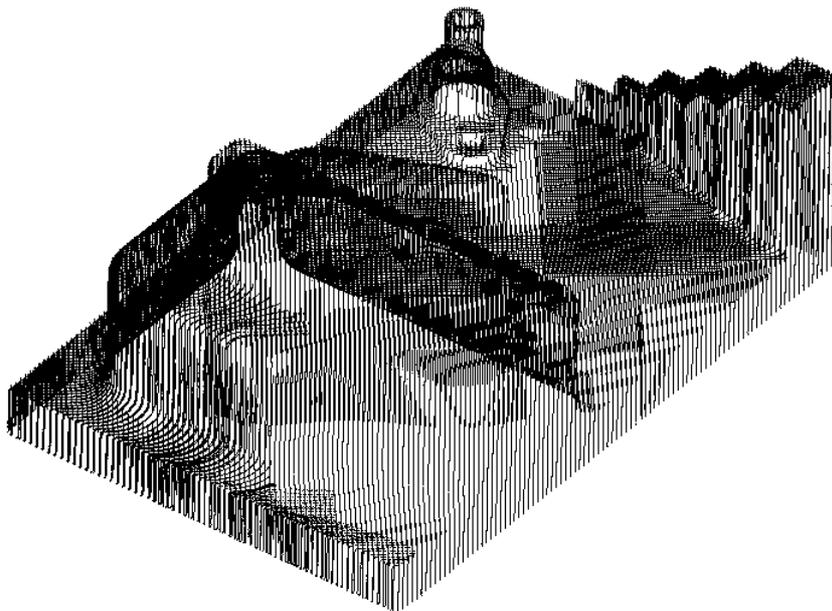


Figure C.VII.2 : Ensemble de points 3D fournis par un capteur télémétrique (495038 points).

Annexe D

Exemples d'Inspection Automatique Visuelle

D.1. Inspection de pièces moulées à partir de leur modèle CAO et d'une image de profondeur

Newman [Newman 94] présente une méthode d'inspection d'objets moulés présentant des surfaces planes ou quadriques, à partir de leur modèle CAO et d'une image de profondeur, et capable de traiter 90% des cas. La technique présentée permet de détecter certains types de défauts de moulage, inspection de caractéristiques de fabrication et de tolérances dimensionnelles.

Cette méthode consiste dans un premier temps à segmenter l'image de profondeur en régions et classer celles-ci dans une des catégories suivantes : plan, sphère, cylindre ou conique, puis à inspecter pour chacune d'elles leur déviation de planéité ou de cylindricité.

La planéité est vérifiée de la façon suivante : chaque région planaire est divisée en sous-régions de taille suffisamment petite, dans lesquelles on essaye de faire passer un plan, par itérations successives éliminant les points trop éloignés. Si un des points de la région est alors plus éloigné du plan que la tolérance admise, il est considéré comme point défectueux et la pièce est rejetée.

La cylindricité est vérifiée en faisant passer un cylindre idéal par les points, si deux ou plusieurs points sont éloignés de plus de la tolérance admise, la pièce est rejetée.

En plus des tolérances de forme que sont la planéité et la cylindricité, sont vérifiées quelques tolérances dimensionnelles, comme la circularité des trous, les rayons, la distance entre plans parallèles,...

D.2. Inspection de pièces mécaniques à partir d'un modèle géométrique et d'images de profondeur

Dans [Newman 90], les données de profondeur sont segmentées afin d'en extraire : des points de surface orientés (points dans l'espace avec normale de surface), des arêtes de l'objet, des descriptions de primitives de surface telles que des plans, cylindres, sphères, quadriques... permettant d'effectuer une bonne mise en correspondance.

L'inspection est menée par la vérification des points suivants : dimensions, positionnement, et positionnement relatif des caractéristiques.

Newman présente également une étude sur l'origine des erreurs d'inspection, à partir des erreurs introduites au niveau du calibrage du capteur, de l'acquisition des images, de la segmentation et des erreurs numériques liées aux méthodes numériques. Avec le système qu'il a utilisé, une tolérance d'inspection de 0,25 mm est réalisable mais il n'est pas envisageable d'obtenir une précision de 0,025 mm.

D.3. Inspection de pièces mécaniques définies en CAO avec une image 2D

Les particularités de cette méthode [Modayur 92] sont les suivantes : définition précise de tolérances géométriques pour l'utilisation d'images, le développement d'algorithmes de mesure correspondant directement à ces définitions, évaluation des erreurs de la phase de mesure, utilisation de ces informations d'erreurs pour la prise de décision. Ceci permet de vérifier les tolérances de formes sur les arêtes droites, et les angles entre arêtes.

D.4. Vérification de cylindricité avec une machine à mesurer

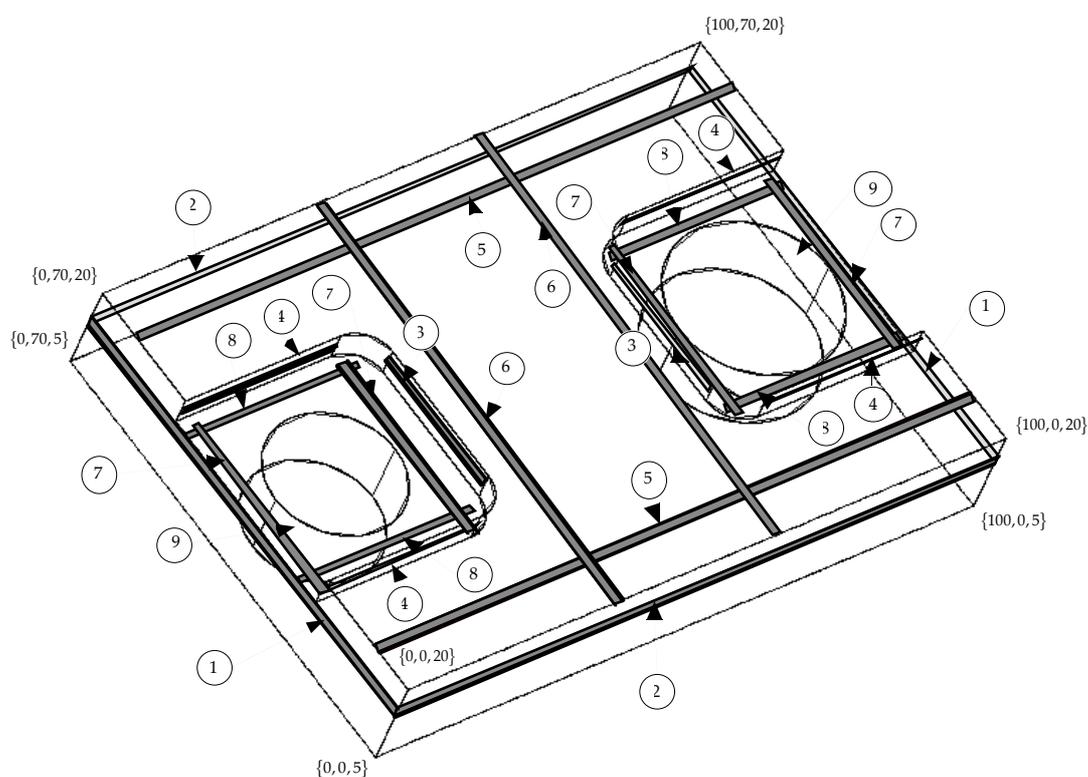
Kurfess [Kurfess 95] a développé une méthode permettant de vérifier la cylindricité de pièces à partir de données fournies par une machine à mesurer. A partir des points de mesure sélectionnés, faire passer un cylindre idéal, puis des cylindres déformés comportant plus de paramètres. Lorsqu'un cylindre correct (donnant un degré de satisfaction correct) est trouvé, vérifier les points qui sont dans l'intervalle de tolérance des paramètres du cylindre.

D.5. Inspection à partir d'un modèle CAO et d'une machine à mesurer

Dans cette méthode [Menq 92], à partir du modèle CAO, un plan d'inspection est généré pour mesurer à l'aide d'une machine à mesurer un nombre de points uniformément répartis sur la surface à inspecter. Après mise en correspondance des points mesurés avec les points idéaux, les points qui sont jugés trop éloignés sont écartés.

Annexe E

Inspection du Support



1 : Planéité des plans $x=0$ et $x=100$

2 : Planéité de plans $y=0$ et $y=70$

3 : Planéité du lamage pour $x=31$ et $x=69$

4 : Planéité du lamage pour $y=19$ et $y=51$

5 : Planéité pour $z=20$ suivant x

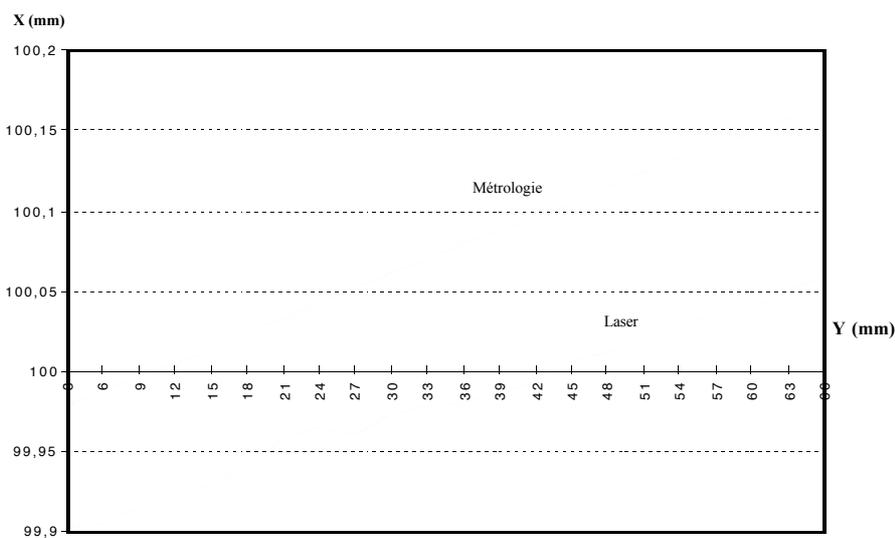
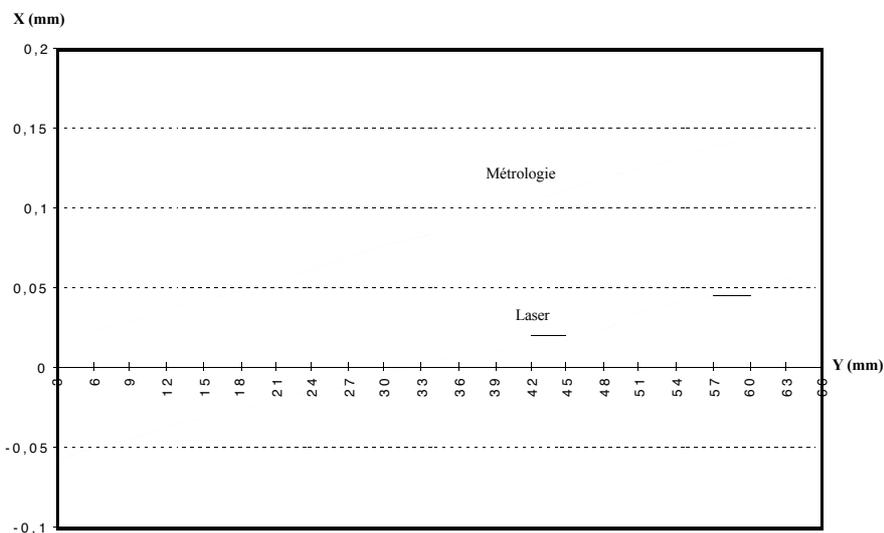
6 : Planéité pour $z=20$ suivant y

7 : Planéité pour $z=15,5$ suivant y

8 : Planéité pour $z=15,5$ suivant x

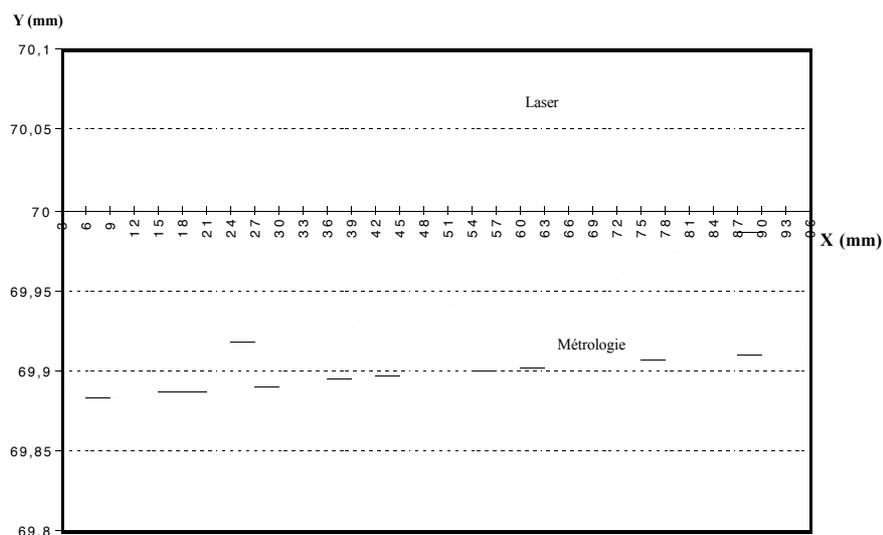
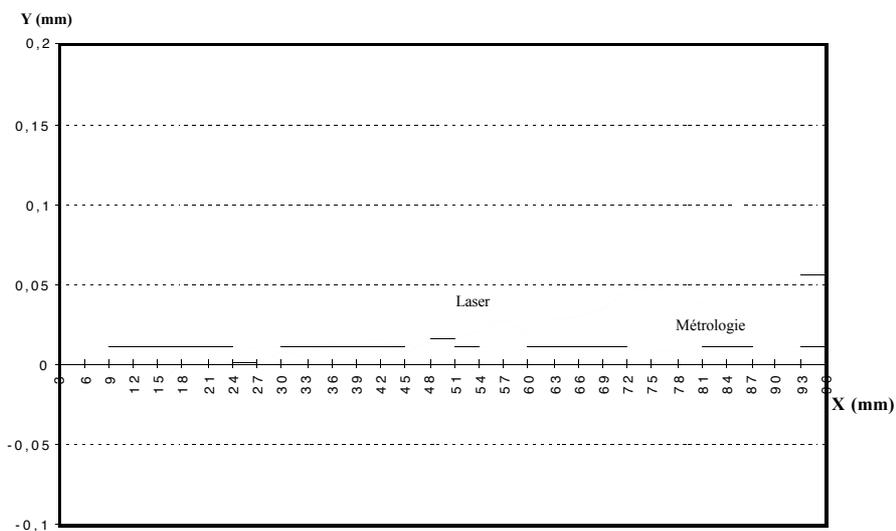
9 : Circularité des deux perçages

E.1 : Planéité pour $x=0$ et $x=100$



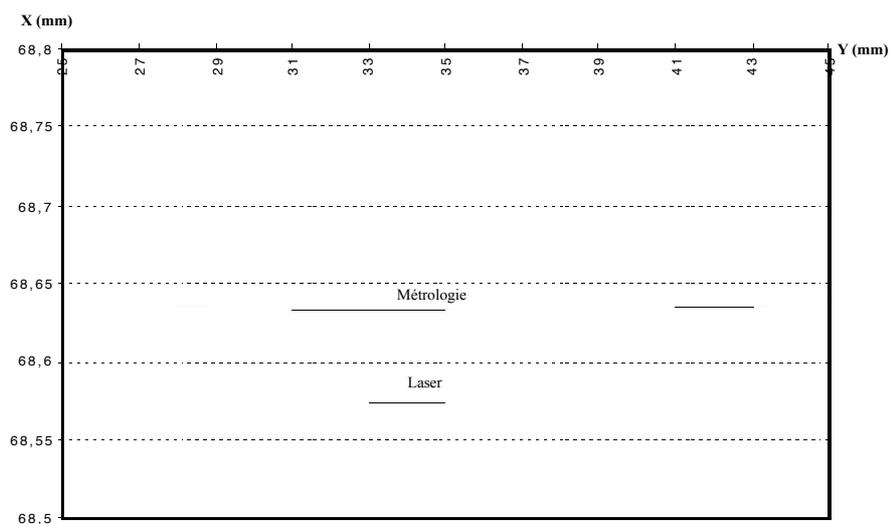
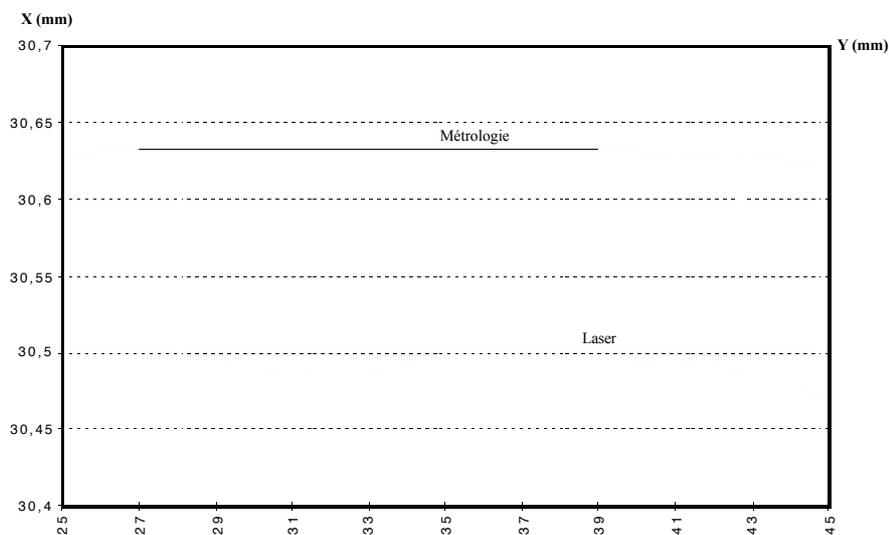
	Planéité pour $x=0$		Planéité pour $x=100$	
	Laser	Métrologie	Laser	Métrologie
Moyenne	0,002626	0,086659	99,979959	100,073682
Ecart.Moyen	0,029224	0,037432	0,038905	0,049682
Ecartype	0,033890	0,043008	0,046621	0,057099
Maximum	0,056418	0,157000	100,059000	100,167000
Minimum	-0,057640	0,016000	99,901600	99,979000
Médiane	0,004566	0,086250	99,979550	100,075000
Somme.Carrés.Ecarts	0,025268	0,040693	0,047818	0,071727
Var.P	0,001149	0,001850	0,002174	0,003260
Pente	0,001773	0,002259	0,002439	0,003000

E.2 : Planéité pour y=0 et y=70



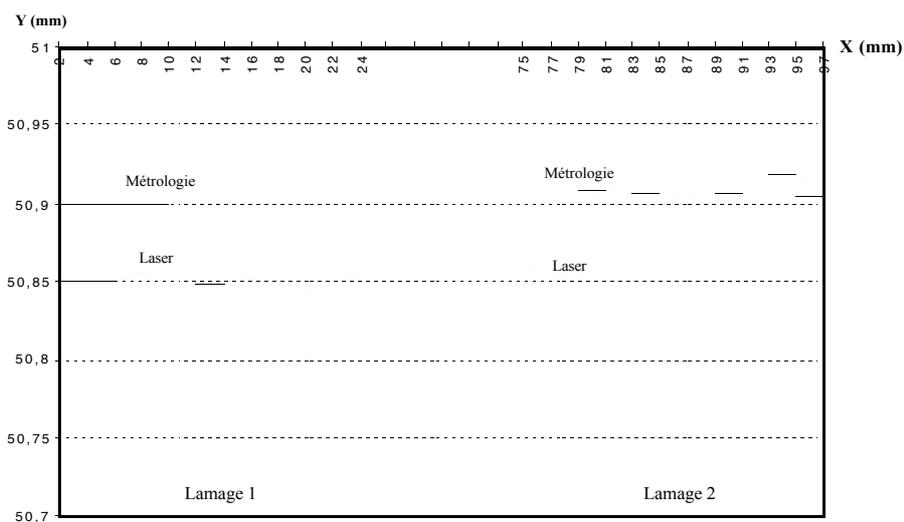
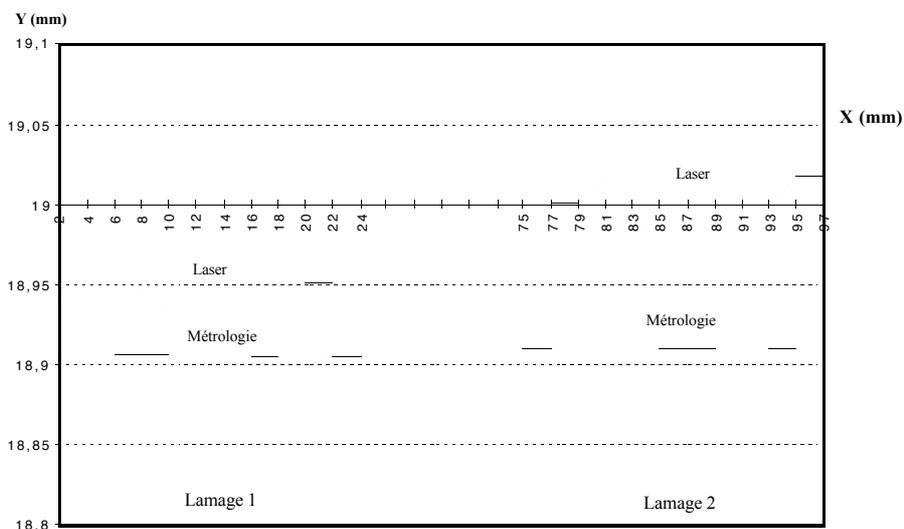
	Planéité pour y=0		Planéité pour y=100	
	Laser	Métrologie	Laser	Métrologie
Moyenne	0,020035	0,011281	69,943500	69,897688
Ecart.Moyen	0,018804	0,000646	0,028656	0,007949
Ecartype	0,022028	0,000819	0,035476	0,009604
Maximum	0,056793	0,013000	70,054200	69,918500
Minimum	-0,014476	0,009000	69,889600	69,882000
Médiane	0,016856	0,011000	69,942150	69,897250
Somme.Carrés.Ecarts	0,015527	0,000021	0,040274	0,002951
Var.P	0,000485	0,000001	0,001259	0,000092
Pente	0,000780	-0,000014	0,001109	0,000342

E.3 : Planéité du lamage pour $x=31$ et $x=69$



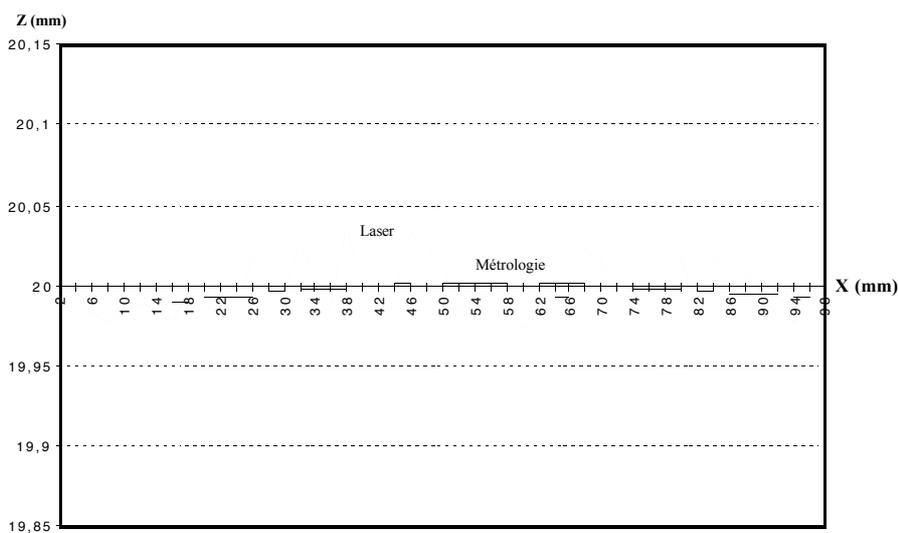
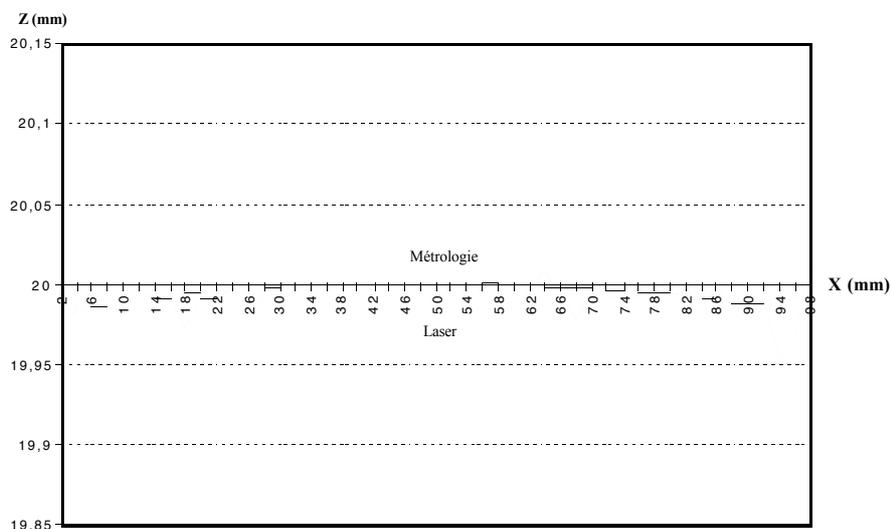
	Planéité lamage $x=31$		Planéité lamage $x=69$	
	Laser	Métrologie	Laser	Métrologie
Moyenne	30,491100	30,631118	68,573809	68,634836
Ecart.Moyen	0,006218	0,002519	0,003631	0,001671
Ecartype	0,007977	0,003980	0,004881	0,002096
Maximum	30,499900	30,633800	68,584800	68,639500
Minimum	30,470900	30,619000	68,566500	68,633000
Médiane	30,494000	30,632700	68,573100	68,633800
Somme.Carrés.Ecarts	0,000700	0,000174	0,000262	0,000048
Var.P	0,000064	0,000016	0,000024	0,000004
Pente	-0,000675	-0,000289	-0,000241	-0,000059

E.4 : Planéité pour y=19 et y=51



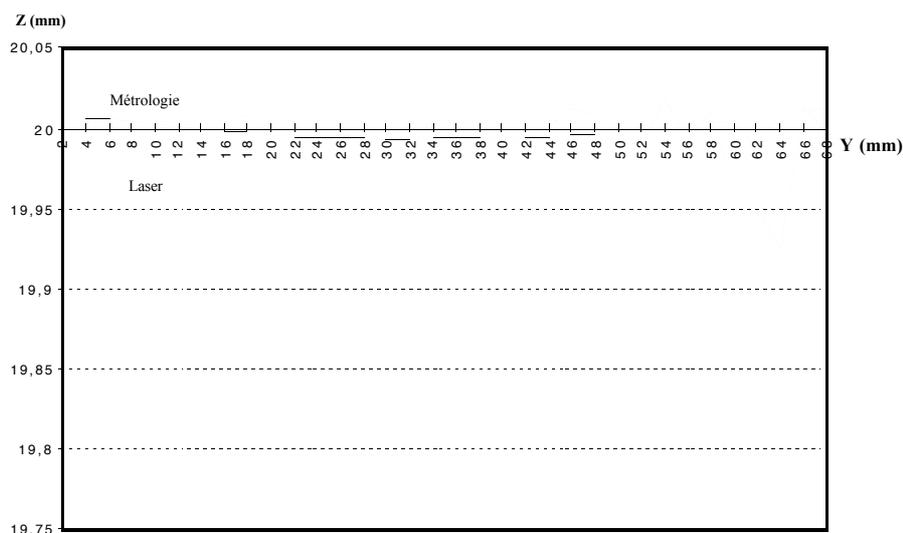
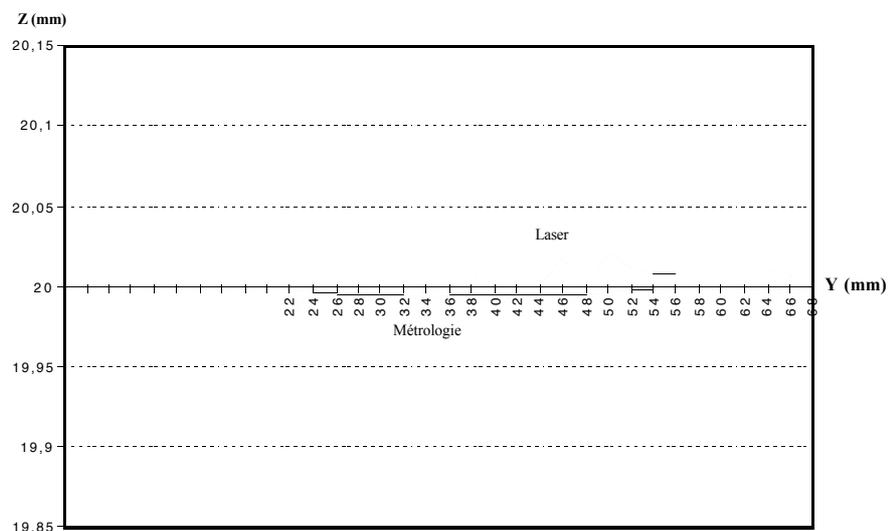
	Lamage 1 pour y=19		Lamage 2 pour y=19		Lamage 1 pour y=51		Lamage 2 pour y=51	
	Laser	Méto.	Laser	Méto.	Laser	Méto.	Laser	Méto.
Moyenne	18,946317	18,905167	19,007692	18,910217	50,848800	50,899450	50,901342	50,904850
Ecart.Moyen	0,005244	0,001317	0,006139	0,000586	0,003683	0,001067	0,012354	0,001492
Ecartype	0,006565	0,001606	0,007267	0,000654	0,005520	0,001931	0,018927	0,002214
Maximum	18,957700	18,908500	19,018200	18,911200	50,855400	50,900500	50,919100	50,908000
Minimum	18,931900	18,903000	18,997100	18,909300	50,834000	50,893200	50,846500	50,899200
Médiane	18,948050	18,904950	19,006700	18,910000	50,849550	50,900050	50,905500	50,905400
Σ Carrés.Ecarts	0,000517	0,000031	0,000634	0,000005	0,000366	0,000045	0,004299	0,000059
Var.P	0,000043	0,000003	0,000053	0,000000	0,000030	0,000004	0,000358	0,000005
Pente	0,000629	-0,000171	0,000480	0,000042	-0,000154	-0,000129	0,001863	-0,000086

E.5 : Planéité pour z=20 suivant x



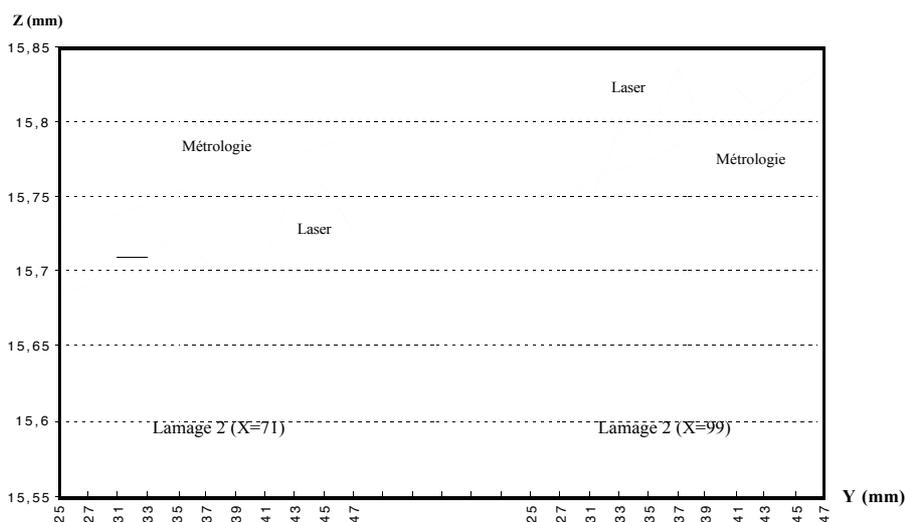
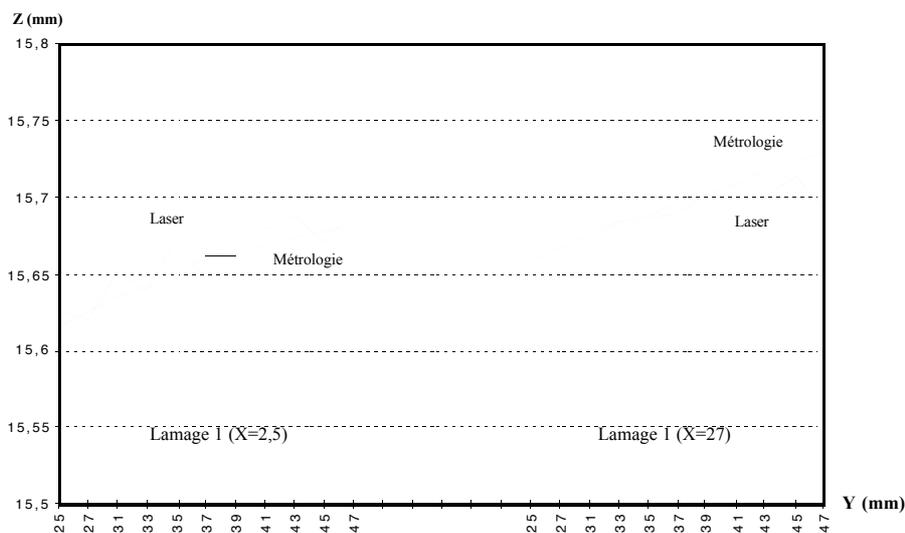
	Planéité pour z=20		Planéité pour z=20	
	Laser	Métrologie	Laser	Métrologie
Moyenne	19,974857	19,997622	19,999427	19,995327
Ecart.Moyen	0,026198	0,010125	0,012002	0,005144
Ecartype	0,041613	0,020780	0,015149	0,006400
Maximum	20,014400	20,112000	20,035000	20,002500
Minimum	19,760200	19,952000	19,973500	19,977000
Médiane	19,989300	19,995000	19,997900	19,998000
Somme.Carrés.Ecarts	0,084849	0,021158	0,011246	0,002007
Var.P	0,001732	0,000432	0,000230	0,000041
Pente	0,000111	-0,000050	0,000046	0,000120

E.6 : Planéité pour z=20 suivant y



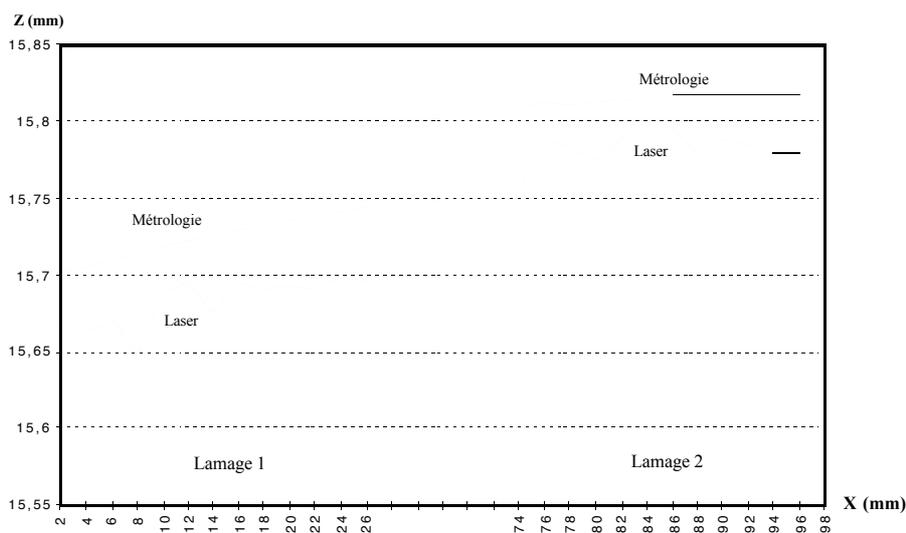
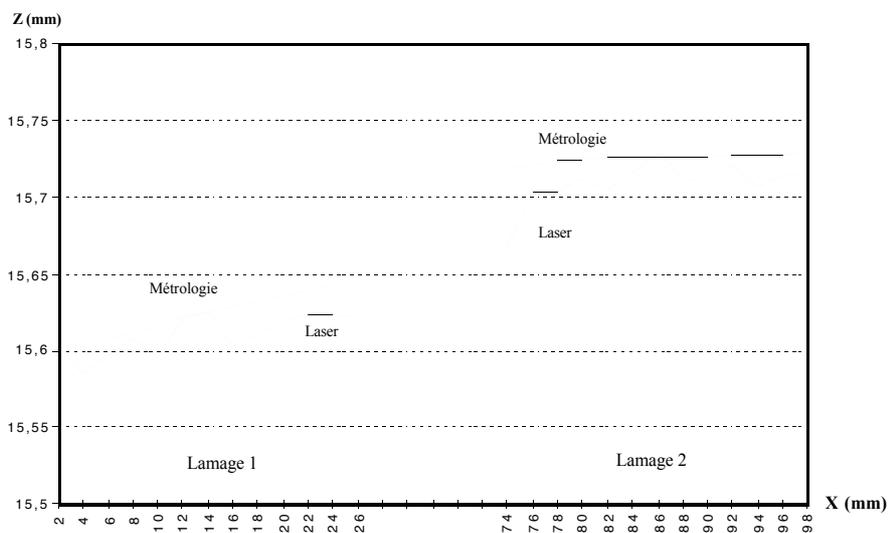
	Planéité pour z=20		Planéité pour z=20	
	Laser	Méetrologie	Laser	Méetrologie
Moyenne	19,979697	20,015221	19,992529	19,998765
Ecart.Moyen	0,037017	0,026510	0,012881	0,004856
Ecartype	0,070852	0,043807	0,018341	0,005819
Maximum	20,021100	20,194000	20,017400	20,013000
Minimum	19,664000	19,993000	19,924800	19,989500
Médiane	20,000250	19,998500	19,995900	19,997250
Somme.Carrés.Ecarts	0,170680	0,065247	0,011437	0,001151
Var.P	0,005020	0,001919	0,000336	0,000034
Pente	0,001359	-0,000977	-0,000077	0,000019

E.7 : Planéité pour z=15,5 suivant y



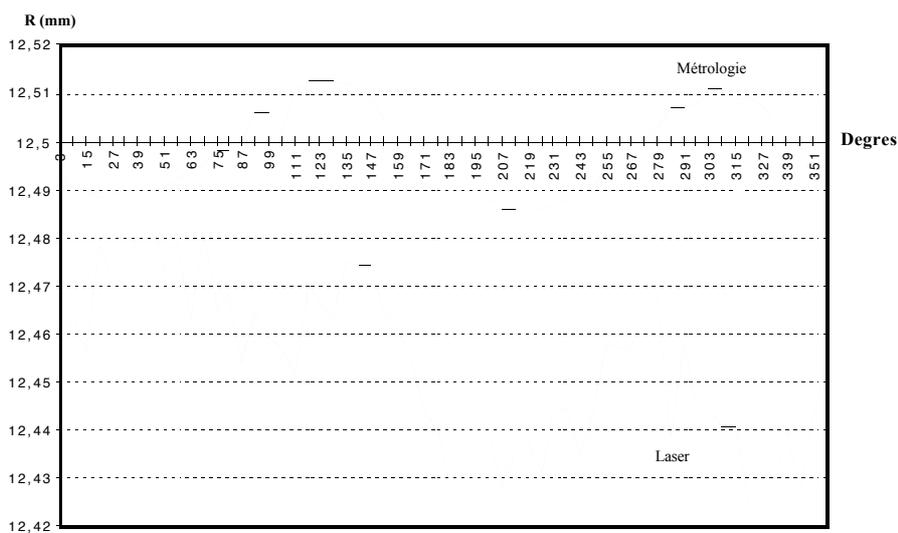
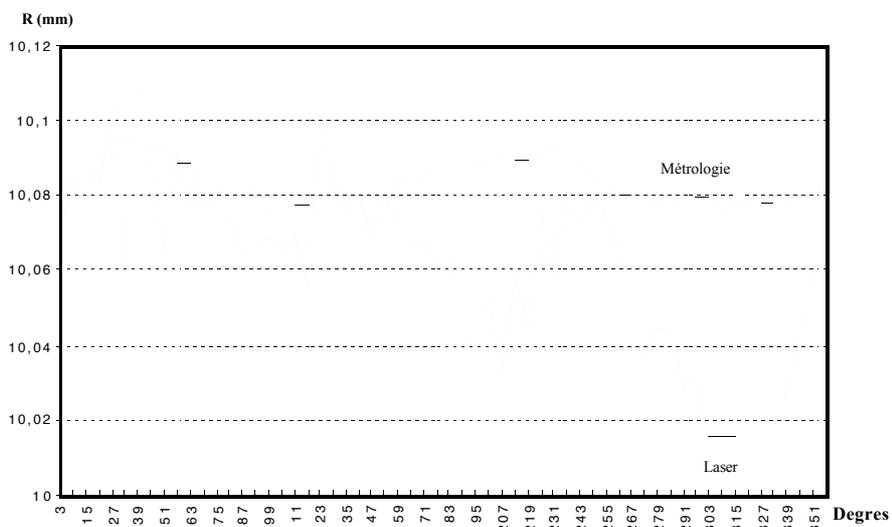
	Lamage 1				Lamage 2			
	Planéité pour x=2,5		Planéité pour x=27		Planéité pour x=71		Planéité pour x=99	
	Laser	Métro.	Laser	Métro.	Laser	Métro.	Laser	Métro.
Moyenne	15,660945	15,653900	15,685309	15,695664	15,718809	15,758718	15,797973	15,778518
Ecart.Moyen	0,022550	0,018855	0,015643	0,018749	0,019374	0,017598	0,023425	0,017417
Ecartype	0,024765	0,021206	0,019950	0,022091	0,022437	0,020169	0,028500	0,019456
Maximum	15,692400	15,683700	15,713900	15,730000	15,756600	15,790000	15,834900	15,804500
Minimum	15,619500	15,615800	15,642000	15,658500	15,684300	15,727700	15,745800	15,746700
Médiane	15,670200	15,661700	15,689300	15,696000	15,709500	15,759500	15,796500	15,784000
Σ Carrés.Ecarts	0,006746	0,004947	0,004378	0,005368	0,005537	0,004475	0,008934	0,004164
Var.P	0,000613	0,000450	0,000398	0,000488	0,000503	0,000407	0,000812	0,000379
Pente	0,003146	0,003063	0,002641	0,003211	0,002634	0,002915	0,003406	0,002800

E.8 : Planéité pour $z=15,5$ suivant x



	y=21				y=49			
	Lamage 1		Lamage 2		Lamage 1		Lamage 2	
	Laser	Métro.	Laser	Métro.	Laser	Métro.	Laser	Métro.
Moyenne	15,610600	15,624223	15,708908	15,725815	15,683138	15,724554	15,783185	15,815454
Ecart.Moyen	0,010969	0,012283	0,008929	0,002289	0,013522	0,011988	0,007278	0,003041
Ecartype	0,012536	0,014066	0,013839	0,002964	0,015174	0,013792	0,010029	0,003408
Maximum	15,624600	15,646300	15,729600	15,729500	15,698900	15,745700	15,795400	15,819500
Minimum	15,584900	15,602500	15,667600	15,719000	15,650200	15,701300	15,755100	15,809200
Médiane	15,610400	15,625500	15,711600	15,726300	15,691800	15,725500	15,784400	15,817000
Σ Carrés.Ecarts	0,002043	0,002572	0,002490	0,000114	0,002993	0,002473	0,001308	0,000151
Var.P	0,000157	0,000198	0,000192	0,000009	0,000230	0,000190	0,000101	0,000012
Pente	0,001263	0,001877	0,001073	0,000368	0,001575	0,001837	0,000403	0,000433

E.9 : Circularité Cercles 1 et 2



	Rayon Cercle 1		Rayon Cercle 2	
	Laser	Métrologie	Laser	Métrologie
Moyenne	10,061708	10,083070	12,453625	12,498150
Ecart.Moyen	0,017987	0,005148	0,014225	0,007532
Ecartype	0,022536	0,006019	0,016445	0,008597
Maximum	10,108800	10,095600	12,485600	12,512500
Minimum	10,012700	10,070800	12,424300	12,485700
Médiane	10,064850	10,082300	12,456400	12,497350
Somme.Carrés.Ecarts	0,030473	0,002173	0,016226	0,004435
Var.P	0,000508	0,000036	0,000270	0,000074
Pente	-0,000155	-0,000019	-0,000109	0,000021

Références

Bibliographiques

- [Arbab 90] **ARBAB, F.** Set models and boolean operations for solids and assemblies. *IEEE Computer Graphics and Applications*, 1990, Vol. 10, N° 6, p. 76-86.
- [Barr 84] **BARR, A.H.** Global and local deformations of solid primitives. *Computer Graphics*, 1984, Vol. 18, N° 3, p. 21-30.
- [Bascle 92] **BASCLE, B. DERICHE, R.** Feature extraction using parametric snakes. *Proceedings 11th IAPR International Conference on Pattern Recognition, Volume III, Conference C : Image, Speech and signal analysis, ICPR 11, The Hague, The Netherlands, August 30 - September 3, 1992, Los Alamitos, California, IEEE Computer Society Press, 1992, p. 659-662.*
- [Bergevin 95] **BERGEVIN, R. LAURENDEAU, D. POUSSART, D.** Registering Range Views of multipart objects. *Computer Vision and Image Understanding*, January 1995, Vol. 61, N° 1, p. 1-16.
- [Bertin 92] **BERTIN, E. CHASSERY, J.M.** 3-D Voronoï diagrams: application to segmentation. *Proceedings 11th IAPR International Conference on Pattern Recognition, Volume III, Conference C : Image, Speech and signal analysis, ICPR 11, The Hague, The Netherlands, August 30 - September 3, 1992, Los Alamitos, California, IEEE Computer Society Press, 1992, p. 197-200.*
- [Besl 88] **BESL, P.J.** Geometric modeling and computer vision. *Proceedings of the IEEE*, August 1988, Vol. 76, N° 8, p. 936-958.
- [Besl 92] **BESL, P.J. MAC KAY, N.D.** A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, February 1992, Vol. 14, N° 2, p. 239-256.
- [Besl 94] **BESL, P.J.** Triangles as a primary representation. *NSF/ARPA Workshop on 3D object representation for computer vision*, New York City, December 5-7, 1994, 64 pages.
- [Bézier 87] **BEZIER, P.** *Mathématiques et CAO, Volume IV : Courbes et surfaces*, Deuxième édition revue et augmentée. Paris Londres Lausanne : Hermès, 1987, 237 pages, ISBN 2-86601-080-9.

- [Bézier 88] **BEZIER, P.** *L'utilisation des courbes et surfaces en CAO*. Paris Londres Lausanne : Hermès, 1988, 64 pages, ISBN 2-86601-139-2.
- [Bhanu 87] **BHANU, B. HO, C.C.** CAD Based 3D object representation for robot vision. *Computer*, August 1987, p. 19-35.
- [Bispo 94] **BISPO, E.M. FISHER, R.B.** Inspection of free-form surfaces using dense range data. *SPIE Automated 3D and 2D Vision*, 1994, Vol. 2249, p. 211-220.
- [Bittar 93] **BITTAR, E. LAVALLEE, S. SZELINSKI, R.** A method for registering range images of arbitrarily shaped surfaces for 3D object reconstruction. *Proceedings of SPIE Sensor fusion VI*, November 1993, Vol. 2059, p. 384-395.
- [Blais 95] **BLAIS, G. LEVINE, M.D.** Registering multi-view range data to create 3D computer objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, August 1995, Vol. 17, N° 8, p. 820-824.
- [Boulanger 93a] **BOULANGER, P. SEKITA, I.** *Automatic extraction of complex surface models from range images using a trimmed-rational bezier surface*. Ottawa : Conseil National de Recherche du Canada, 1993, 11 p., N° 35097.
- [Boulanger 93b] **BOULANGER, P.** *Reverse Engineering of complex surfaces based on a new hierarchical segmentation method*. Ottawa : Conseil National de Recherche du Canada, 1993, 12 p., N° 37051.
- [Boulanger 94a] **BOULANGER, P. COHEN, P.** Viewpoint Invariant Computation of Surface Curvatures in Range Images. *Proc. of Vision Interface 94*, Banff, 1994, p. 145-154.
- [Boulanger 94b] **BOULANGER, P.** *Extraction Multiéchelle d'éléments géométriques*, PhD, Université de MONTREAL, Ecole Polytechnique, Aout 1994, 225 p.
- [Brady 82] **BRADY, M.** Computational approaches to image understanding. *Computing surveys*, March 1982, Vol. 14, N° 1, p. 3-71.
- [Brown 81] **BROWN, C.M.** Some mathematical and representational aspects of solid modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, July 1981, Vol. 3, N° 4, p. 444-453.
- [Brown 92] **BROWN, L.G.** A survey of image registration techniques. *ACM Computing Surveys*, December 1992, Vol 24, N° 4, p. 325-376.
- [Cardew Hall 93] **CARDEW-HALL, M.J. LABANS, T. WEST, G. DENCH, P.A.** Method for representing dimensions and tolerances on solid based freeform surfaces. *Robotics and Computer-Integrated Manufacturing*, June 1993, Vol. 10, N° 3, p. 223-234.
- [Chen 92] **CHEN, Y. MEDIONI, G.** Object modelling by registration of multiple range images. *Image and Vision Computing*, 1992, Vol. 10, N° 3, p. 145-155.
- [Chevalier 95] **CHEVALIER, A.** *Guide du dessinateur industriel*. Paris : Hachette technique, 1995, 320 pages.

- [Chin 86b] CHIN, R.T. DYER, C. Model based recognition in robot vision. *Computing Surveys*, March 1986, Vol. 18, N° 1, p. 67-108.
- [Clarysse 95] CLARYSSE, P. JAOUEN, O. MAGNIN, I.E. MORVAN, J.M. Interpolation 3D des parois cardiaques à partir de données discrètes. *Journées GDR - Traitement du signal et images, Mouvement 3D d'objets non-rigides*, Lille, 4-5 Mai 1995, 64 pages.
- [Cohen 94] COHEN, I. COHEN, L.D. *A hybrid hyperquadric model for 2D and 3D data fitting*. Rocquencourt : INRIA, Janvier 1994, 34 pages, RR-2188.
- [Collignon 94] COLLIGNON, A. VANDERMEULEN, D. SUETEN, P., MARCHAL, G. Registration of 3D multi-modality medical images using surfaces and point landmarks. *Pattern Recognition Letters*, 1994, Vol. 15, N° 5, p. 461-467.
- [Coquillart 90] COQUILLART, S. Extended free-form deformations: A sculpturing tool for 3D geometric modeling. *Computer Graphics*, August 1990, Volume 24, Number 4, p. 187-193.
- [Crocker 91] CROCKER, G.A. REINKE, W.F. An editable nonmanifold boundary representation. *IEEE Computer Graphics and Applications*, March 1991, Vol. 11, N° 2, p. 39-51.
- [Delingette 94] DELINGETTE, H. *Simple meshes : a general representation for 3D shape reconstruction*. Rocquencourt : INRIA, 1994, RR-2214, 13 pages.
- [Dolenc 94] DOLENC, A. *An overview of Rapid Prototyping Technologies in Manufacturing*. Helsinki University of Technology : Institute of Industrial Automation, 24 Juillet, 1994, TKO-B116, 21 pages.
- [Dorai 94] DORAI, C. WENG, J. JAIN, A.K. Optimal registration of multiple range views. *Proceedings, 12th IAPR International Conference on Pattern Recognition*, Conference A (Jerusalem, Israel, October 9-13, 1994), Los Alamitos, California : IEEE Computer Society Press, 1994, p. 569-571.
- [Farin 92a] FARIN, G. From conics to NURBS: a tutorial and survey, *IEEE Computer Graphics and Applications*, September 1992, Vol. 12, N° 5, p. 78-86.
- [Farin 92b] FARIN, G. *Courbes et surfaces pour la CGAO*. Paris : Masson. 1992, 401 pages, ISBN 2-225-82765-6.
- [Faugeras 86] FAUGERAS, O.D. HEBERT, M. The representation, recognition and locating of 3D objects. *The International Journal of Robotic Research*, Fall 1986, Vol 5, No 3, p. 27-52.
- [Faux 81] FAUX, I.D. PRATT, M.J. *Computational geometry for design and manufacture*. New York : Halsted Press, Chichester, Eng : Ellis Harwood Series in Mathematics and its applications, 1981, 329 pages.
- [Ferrie 93] FERRIE, F.P. LAGARDE, J. WHAITE, P. Darboux frames, snakes, and super-quadrics, geometry from the bottom up. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, August 1993, Vol. 15, N° 8, p. 771-784.

- [Filbois 93] **FILBOIS, A. GEMMERLE, D.** Modélisation d'objets polyédriques. *4^{ème} Journées ORASIS*, Mulhouse, 11-14 octobre 1993, p. 128-131.
- [Fleming 88] **FLEMING, A.** Geometric relationships between toleranced features. *Artificial Intelligence*, 1988, Vol. 37, p. 403-412.
- [Flynn 91] **FLYNN, P.J. JAIN, A.K.** CAD-based computer vision: From CAD models to relational graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, February 1991, Vol. 13, N° 2, p. 114-132.
- [Fujimura 92] **FUJIMURA, K. YOKOYA, N. YAMAMOTO, K.** Motion tracking of deformable objects based on energy minimization using multiscale dynamic programming. *Proceedings 11th IAPR International Conference on Pattern Recognition, Volume III, Conference C : Image, Speech and signal analysis, ICPR 11*, The Hague, The Netherlands, August 30 - September 3, 1992, Los Alamitos, California : IEEE Computer Society Press, 1992, p. 83-86.
- [Gigus 90] **GIGUS, Z. MALIK, J.** Computing the aspect graph for line drawings of polyhedral objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, February 1990, Vol 12, N° 2, p. 113-122.
- [Givert 88] **GIVERT, D.** *Modélisation géométrique d'objets en environnement robotique*. Thèse de docteur en sciences : Université de Paris Sud Centre d'Orsay, 1988, 324 pages.
- [Godin 94] **GODIN, G. RIOUX, M. BARIBEAU, R.** Three-dimensional registration using range and intensity information. *Proceedings SPIE Videometrics III*, Novembre 1994, Vol. 2350, Boston, p. 279-290.
- [Godin 95] **GODIN, G. BOULANGER, P.** Range image registration through viewpoint invariant computation of curvature. *IAPRS*, Vol. 30, Part 5W1, ISPRS Intercommission Workshop "From pixels to sequences", Zurich, March 22-24, 1995, p. 170-175.
- [Gossart 88] **GOSSART, D.C. ZUFFANTE, R.P. SAKURAI, H.** Representing dimensions, tolerances, and features in MCAE systems. *IEEE Computer Graphics and Applications*, 1988, Vol. 8, N° 2, p. 51-59.
- [Gros 94] **GROS, P.** Using quasi-invariants for automatic model building and object recognition: an overview. *NSF/ARPA Workshop on 3D object representation for computer vision*, New York City, December 5-7, 1994, 64 pages.
- [Higuchi 94] **HIGUCHI, K. DELINGETTE, H. HEBERT, M. IKEUCHI, K.** Merging multiple views using a spherical representation, *Proceedings of the 2nd workshop on CAD Based vision*, Champaign, Pennsylvania, Feb 94, 12 p.
- [Hoppe 92] **HOPPE, H. DE ROSE, T. DUCHAMP, T. Mc DONALD, STUETZLE, W.** Surface reconstruction from unorganized points. *Computer Graphics*, July 1992, Vol. 26, N° 2, p. 71-78.
- [Hoppe 93] **HOPPE, H. DE ROSE, T. DUCHAMP, T. Mc DONALD, STUETZLE, W.** Mesh optimization. *SIGGRAPH'93, Conference proceedings, Anaheim, CA, August 1-6*, ACM , New York, 1993, p. 19-26.

- [Huang 93] HUANG, W.C. GOLDFOF, D.B. Adaptive-size meshes for rigid and non-rigid shape analysis and synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, June 1993, Vol. 15, N° 6, p. 611-616.
- [Huttenlocher 94] HUTTENLOCHER, D. Using two-dimensional models to interact with the three-dimensional world. *NSF/ARPA Workshop on 3D object representation for computer vision*, New York City, December 5-7, 1994, 64 pages.
- [Ikeuchi 94] IKEUCHI, K. HEBERT, M. Spherical representation: from EGI to SAI. *NSF/ARPA Workshop on 3D object representation for computer vision*, New York City, December 5-7, 1994, 64 pages.
- [Jacq 95] JACQ, J.J. ROUX, C. Registration of 3D images by genetic optimization. *Pattern Recognition Letters*, 1995, Vol. 16, p. 823-841.
- [Juster 92] JUSTER, N.P. Modeling and representation of dimensions and tolerances : A survey. *Computer Aided Design*, 1992, Vol. 24, N° 1, p. 3-17.
- [Kamgar 91] KAMGAR PARSİ, B. JONES, J.L. ROSENFELD, A. Registration of multiple overlapping range images : Scenes without distinctive features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1991, Vol. 13, N° 9, p. 857-871.
- [Kholadi 91] KHOLLADI, M.K. *Représentation, modélisation et manipulation des connaissances spatiales en géomatique "G.R.E.M.A.C.O.S"*. Thèse de doctorat : INSA de Lyon, LISI, 1991, 222 pages.
- [Krishnan 94] KRISHNAN, S. MANOCHA, D. *An efficient surface intersection algorithm based on lower dimensional formulation*. University of North California, November 1994, 13 p., TR94-062.
- [Kurfess 95] KURFESS, T.R. BANKS, D.L. Statistical verification of conformance to geometric tolerance. *Computer Aided Design*, May 1995, Vol. 27, N° 5, p. 353-361.
- [Lamousin 94] LAMOUSIN, H.J. WAGGENSPACK, W.N. NURBS-based free-form deformations. *IEEE Computer Graphics and Applications*, November 1994, Vol. 14, N° 6, p. 59-65.
- [Lavallée 95] LAVALLEE, S. SZELISKI, R. Recovering the position and orientation of free-form objects from image contours using 3D distance maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995, Vol. 17, N° 4, p. 378-390.
- [Marshall 91] MARSHALL, A.D. MARTIN, R.R. HUTBER, D. Automatic inspection of mechanical parts using geometric models and laser range finder data. *Image and Vision Computing*, Vol. 9, N° 6, December 1991, p. 385-405.
- [Masuda 95] MASUDA, T. YOKOYA, N. A Robust Method for Registration and Segmentation of Multiple Range Images. *Computer Vision and Image Understanding*, 1995, Vol. 61, N° 3, p. 295-307.

- [Menq 92] **MENQ, C.H. YAU, H.T. LAI, G.Y.** Automated precision measurement of surface profile in CAD directed inspection. *IEEE Transactions on Robotics and Automation*, April 1992, Vol. 8, N° 2, p. 353-361.
- [Milgram 92] **MILGRAM, M.** *Reconnaissance des formes, méthodes numériques et connexionnistes*. Paris : Armand Colin, 1992, 173 pages.
- [Modayur 92] **MODAYUR, B.R. SHAPIRO, L.G. HARALICK, R.M.** Visual inspection of machined parts. *Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Los Alamitos, CA : IEEE Computer Society Press, USA, 1992, p. 393-398.
- [Nayar 94] **NAYAR, S. MURASE, H.** Parametric appearance representation. *NSF/ARPA Workshop on 3D object representation for computer vision*, New York City, December 5-7, 1994, 64 pages.
- [Newman 94] **NEWMAN, T.S. JAIN, A.K.** CAD-Based inspection of 3D objects using range images. *Proceedings of the 1994 Second CAD-Based Vision Workshop*, Los Alamitos, CA : IEEE Computer Society Press, USA, 1994, p. 236-243.
- [Newman 95] **NEWMAN, T.S. JAIN, A.K.** A survey of automated visual inspection, *Computer Vision and Image Understanding*, March 1995, Vol. 61, N° 2, p. 231-262.
- [Nguen 92] **NGUYEN, Q.L. LEVINE, M.D.** 3-D Object representation in range images using geons. *Proceedings 11th IAPR International Conference on Pattern Recognition, Volume III, Conference C : Image, Speech and signal analysis, ICPR 11*, The Hague, The Netherlands, August 30 - September 3, 1992, Los Alamitos, California : IEEE Computer Society Press, 1992, p. 149-153.
- [Noble 95] **NOBLE, J.A.** From inspection to process understanding and monitoring : A view on computer vision in manufacturing. *Image and Vision Computing*, April 1995, Vol. 13, Number 3, p. 197-214.
- [Pentland 91] **PENTLAND, A. SCLAROFF, S.** Closed-form solutions for physically based shape modeling and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, July 1991, Vol. 13, N° 7, p. 715-729.
- [Peroche 90] **PEROCHE, B. ARGENCE, J. GHAZANFARPOUR, D. MICHELUCCI, D.** *La synthèse d'images*. Paris - Londres - Lausanne : Edition Hermès. 1990, 295 p. ISBN 2-86601-112-0.
- [Ponce 88] **PONCE, J.** *Représentation des objets tridimensionnels*. Thèse de doctorat en sciences : INRIA Rocquencourt, 1988, 153 p., ISBN 2-7261-0564-5.
- [Press 92] **PRESS, W.H. TEULISKY, S.A. VETTERLING, W.T. FLANNERY, B.P.** *Numerical recipes in C : The art of scientific computing*. Second Edition, Cambridge : Cambridge University Press, 1992, 993 pages.
- [Qamhiyah 93] **QAMHIYAH, A. Z. VENTER, R.D. BENHABIB, B.** *Classification of 3D objects models for CAD*, ASME Winter Arrival Meeting, New Orleans, DE-Vol. 66, 1993, p. 119-124.

- [Requicha 86] **REQUICHA, A.A.G. CHAN, S.C.** Representation of geometric features tolerances, and attributes in solid modelers based on constructive geometry. *IEEE Journal of Robotics and Automation*, September 1986, Vol. RA-2, N° 3, p. 156-166.
- [Robert 95] **ROBERT, A. SCHMITT, F.** Modélisation de la forme et du mouvement du coeur à l'aide de superquadriques déformables globalement, *Journées GDR - Traitement du signal et images, Mouvement 3D d'objets non-rigides*, Lille, 4-5 Mai 1995, 64 pages.
- [Rousseuw 87] **ROUSSEUW, P.J. et LEROY, A.M.** *Robust Regression & Outlier Detection*, New York : John Wiley & Sons, Wiley Series in probability and statistics, 1987, ISBN : 0471852333.
- [Sabata 91] **SABATA, B. AGGARWAL, J.K.** Estimation of motion from a pair of range images : A review. *CVGIP: Image Understanding*. 1991, Vol. 54, N° 3, p. 309-324.
- [Sclaroff 95] **SCLAROFF, S.E.** *Model matching : a method for describing, comparing, manipulating digital signals*. Phd : Massachusetts Institute of Technology, 1995, 213 p.
- [Sebahi 95] **SABAHI, A. HERMENT, A. DE CESARE, A. MOUSSEAU, E.** Reconstruction 3D de cavités cardiaques à l'aide d'un modèle de surface déformable. *Journées GDR - Traitement du signal et images, Mouvement 3D d'objets non-rigides*, Lille 4-5 Mai 1995, 64 pages.
- [Sedeborg 86] **SEDEBERG T.W. PARRY S.R.,** *Free-form deformation of solid geometric models*. ACM SIGGRAPH'86, Volume 20, Number 4, 1986, Dallas, August 18-22, p. 151-160
- [Sedeborg 90a] **SEDEBERG, T.** Techniques for cubic algebraic surfaces. *IEEE Computer Graphics and Applications*, Juillet 1990, p. 14-25.
- [Sedeborg 90b] **SEDEBERG, T.** Techniques for cubic algebraic surfaces. *IEEE Computer Graphics and Applications*, Septembre 1990, p. 12-21.
- [Selsis 93] **SELSIS, M. VIEREN, C. POSTAIRE, J.G.** Extraction et modélisation des contours d'objets en mouvement dans une séquence d'images. *4^{ème} Journées ORASIS*, Mulhouse, 11-14 Octobre 1993, p. 82-85.
- [Shapiro 94] **SHAPIRO, L.G. COSTA, M.S.** Appearance based 3D object recognition. *NSF/ARPA Workshop on 3D object representation for computer vision*, New York City, December 5-7, 1994, 64 pages.
- [Shirai 87] **SHIRAI, Y.** *Three dimensional computer vision*, Berlin : Springer-Verlag, 1987, 296 pages, ISBN 0-387-15119-2.
- [Simon 94] **SIMON, D.A. HEBERT, M. KANADE, T.** Real-time 3D pose estimation using a high-speed range sensor. *IEEE International Conference on Robotics and Automation*, San Diego, CA, May 8-13 1994, p. 2235-2241.
- [Solina 90] **SOLINA, F. BAJCSY, R.** Recovery of Parametric Models from Range Images : The case of superquadrics with global deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, February 1990, Vol. 12, N° 2, 13 p.

- [Soucy 92] **SOUCY, M. LAURENDEAU, D.** Surface modeling from dynamic integration of multiple range images. *Proceedings 11th IAPR International Conference on Pattern Recognition, Volume III, Conference C : Image, Speech and signal analysis, ICPR 11, The Hague, The Netherlands, August 30 - September 3, 1992, Los Alamitos, California : IEEE Computer Society Press, 1992, p. 449-452.*
- [Stockman 82] **STOCKMAN, G. KOPSTEIN, S. BENETT, S.** Matching images to models for registration and object detection via clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, May 1982, Vol. 4, N° 3, p. 229-241.
- [Stockman 94] **STOCKMAN, G.** Object representation for recognition-by-alignment. *NSF/ARPA Workshop on 3D object representation for computer vision, New York City, December 5-7, 1994, 64 pages.*
- [Terzopoulos 88] **TERZOPOULOS, D. FLEISCHER, K.** Deformable models. *The Visual Computer*, 1988, Vol. 4, p. 306-331.
- [Terzopoulos 91] **TERZOPOULOS, D. VASILESCU, M.** Sampling and reconstruction with adaptative meshes. *Proceedings CVPR'91, Maui, HI, June 3-6, 1991, p. 70-75.*
- [Thingvold 90] **THINGVOLD, J.A. COHEN, E.** Physical modeling with B-Spline Surfaces for interactive design and animation. *Siggraph Proceedings (1990)*, p. 129-137.
- [Toennies 90] **TOENNIES, K.D. UDUPA, J.K. HERMAN, G.T. WORNOM III, I.L. BUCHMAN, S.R.** Registration of 3D objects and surfaces, *IEEE Computer Graphics and Applications*, May 1990, p. 52-62.
- [Turlan 88] **TURLAN, S.** *Métrologie dimensionnelle*, Villeurbanne : INSA Génie Mécanique Construction, 1988, 285 pages.
- [Turner 87] **TURNER, J.U. WOZNY, M.J.** Tolerances in computer-aided geometric design. *The Visual Computer*, 1987, Vol. 3, p. 214-226.
- [Wirtz 91] **WIRTZ, B. MAGGIONI, C.** 3D Pose estimation by an improved Kohonen Net. *Proceedings of the International Workshop on visual form, Capri, Italy, May 27-30, 1991, New York : Plenum, 1992, p. 593-602.*
- [Wu 94] **WU, K. LEVINE, M.D.** Parametric geons : A discrete set of shapes with parametrized attributes, *International Symposium on Optical Engineering and Photonics in Aerospace Sensing : Visual Information Processing, April 1994, Orlando, FL, Vol. 2239, p. 14-26.*
- [Yokoya 92] **YOKOYA, N. KANETA, M. YAMAMOTO, K.** *Recovery of superquadric primitives from range images by simulated annealing.* Tsukuba, Japan : Electrotechnical Laboratory, April 1992, TR-90-18.
- [Zhang 94] **ZHANG, Z.** Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 1994, Vol. 13, N° 2, p. 119-152.
- [Zhuang 94] **ZHUANG, X. HUANG, Y.** Robust 3D-3D pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, August 1994, Vol. 16, N° 8, p. 818-824.

Bibliographie

de l'auteur

- [Moron 95] MORON, V. BOULANGER, P. MASUDA, T. REDARCE, H.T. Automatic Inspection of Industrial Parts Using 3-D Optical Range Sensor, *Videometrics IV*, Sabry F. El Hakim, Editor, Proc. SPIE 2598, pp. 315-325 (1995).
- [Moron 96a] MORON, V. BOULANGER, REDARCE, H.T. JUTARD, A. Mise en correspondance du modèle CAO d'un objet avec son image 3D : Application à l'inspection, *RFIA'96, Congrès de Reconnaissance de Formes et Intelligence Artificielle*, 16-18 janvier 1996, Rennes, pages 913-922.
- [Moron 96b] MORON, V. BOULANGER, P. REDARCE, H.T. JUTARD, A. 3D Range Data / CAD Model Comparison : Industrial Parts Conformity Verification, *First International Conference on Integrated Design and Manufacturing in Mechanical Engineering, IDMMME'96*, Nantes, 15-17 Avril 1996, p. 1023-1032.
- [Moron 96c] MORON, V. Mise en correspondance d'un modèle CAO avec des données 3D, *Journées ORASIS'96*, Clermont Ferrand, 20-24 Mai 1996, p. 13-18.
- [Moron 96d] MORON, V. BOULANGER, P. MASUDA, T. REDARCE, H.T. High Speed Optical Inspection of Rapid Prototyping Parts. *European Symposium on Lasers, Optics, and Vision for Productivity in Manufacturing I*, 10-14 June 1996, Micropolis, Besançon, France.
- [Moron 96e] MORON, V. BOULANGER, P. MASUDA, T. REDARCE, H.T. High Speed and Non-Contact Validation of Rapid Prototyping Parts, *SPIE's International symposium on Intelligent Systems and Advanced Manufacturing: Rapid Product Development Technologies*, Boston, November 18-19, 1996.

FOLIO ADMINISTRATIF

THESE SOUTENUE DEVANT L'INSTITUT NATIONAL DES SCIENCES APPLIQUEES DE LYON

Nom : MORON (avec précision du nom de jeune fille, le cas échéant)	Date de Soutenance 09 Décembre 1996
Prénoms : Véronique, Sophie, Martine	
TITRE : MISE EN CORRESPONDANCE DE DONNEES 3D AVEC UN MODELE CAO : APPLICATION A L'INSPECTION AUTOMATIQUE	
Nature : Doctorat Formation doctorale : Automatique Industrielle	Numéro d'ordre : 96 ISAL 131
Cote B.I.U. - Lyon : T50/210/19 / et bis	
Classe :	
RESUME : Ce travail porte sur l'inspection automatique d'objets comportant des surfaces complexes à partir de leur modèle CAO et de données 3D provenant, soit d'un capteur télémétrique, soit d'une machine à mesurer. Nous présentons d'abord un état de l'art complet et récent sur la modélisation 3D. Nous sélectionnons deux modèles surfaciques : un modèle triangulé et un modèle exact NURBS découpées, pour lesquels nous développons un calcul de distance point/modèle. Nous détaillons ensuite comment obtenir les données 3D à partir d'images fournies par un capteur télémétrique. Nous proposons une méthode de mise en correspondance générale entre des données 3D et un modèle CAO, qui a l'avantage d'être robuste (jusqu'à 50% de points parasites), automatique, et qui fonctionne quelque soit la position initiale des données et du modèle. Nous avons exploité plusieurs applications à cette méthode, parmi lesquelles la reconnaissance de formes, mais surtout l'inspection automatique d'objets de forme complexe. Nous présentons une méthode d'inspection de ce type d'objet, d'une part quantitative avec une approche globale et locale, et d'autre part qualitative utilisant différents procédés de visualisation que nous avons développés. A partir de ces outils, un opérateur ou un système robotisé peut rapidement identifier des pièces défectueuses, ou une dérive du procédé de fabrication, sur une ligne de production.	
MOTS-CLES : Vision, Mesure sans contact, Conception assistée, Modèle 3 dimensions, Surface libre, Optimisation, Reconnaissance forme, Contrôle dimensionnel	
Laboratoires de recherches : Laboratoire d'Automatique Industrielle de l'INSA de Lyon Institut de Technologie de l'Information, Conseil National de Recherche du Canada (Ottawa - Canada)	
Directeurs de thèse : Hervé Tanneguy Redarce (LAI - INSA de LYON) Pierre Boulanger (CNRC - CANADA)	
Président de jury : A. Jutard	
Composition du jury : MM. Ph Bolon, P. Boulanger (Codirecteur), M. Dhome (Rapporteur), D. Pyzak, H.T. Redarce (Codirecteur), S. Tichkiewitch (Rapporteur)	