

République Algérienne Démocratique et Populaire

Ministère de l'enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE KASDI MERBAH OUARGLA

Faculté des Nouvelles Technologies d'information et de la
communication



Département d'électronique et de télécommunication

MÉMOIRE MASTER ACADEMIQUE

Domaine : Electronique et télécommunication

Filière : Télécommunication

Spécialité : Signal et Télécommunication

Présenté par :

BOUKHALFA Sara NAIMI Fatima Zohra

Thème

*La reconstruction 3D d'un cube
par la technique PSO*

Soutenu publiquement

Le : ... /05/2017

Devant le jury :

Mme	S. BENKRINAH	MAA	Présidente	UKM OUARGLA
Mme	S. AOURAR	MAA	Encadreur	UKM OUARGLA
Mme	F. CHARIF	MCB	Examinatrice	UKM OUARGLA

Année universitaire : 2016/2017

REMERCIEMENTS

Nous remercions Dieu le tout puissant et miséricordieux que nous louons pour avoir guidé nos pas dans l'accomplissement de ce modeste travail. Merci à ALLAH

Nous remercions nos encadreurs, Mme CHARIF Fella et Mr MOHAMMED SEDIK CHEBET

pour leurs aides très appréciables caractérisées par leur disponibilité, leur rigoureux encadrement et sans oublier leurs précieux conseils.

Nos remerciements les plus vifs s'adressent aussi membres de jury d'avoir accepté d'examiner et d'évaluer notre travail.

Nous exprimons également notre gratitude à tous les professeurs et enseignants qui ont collaboré à notre formation

Depuis notre premier cycle d'étude jusqu'à la fin de notre cycle universitaire.

Résumé

Ce mémoire étudier le problème de la reconstruction d'un cube en trois dimension, qui consiste à trouver les paramètres intrinsèques et extrinsèques des caméras utilisées. Nous posons le problème en termes d'optimisation, par la technique PSO. Pour la reconstruction 3D d'une forme d'un cube par la méthode de la triangulation.

Mots clés: caméra, reconstruction 3D, optimisation, optimisation par essaim particulaire, triangulation.

Abstract

This work examines the problem of reconstructing a three-dimensional cube, which consists in finding the intrinsic and extrinsic parameters of the cameras used. We propose the problem in terms of optimization, using the PSO technique. For the 3D reconstruction of a form of a cube by the method of triangulation..

Key-words: camera, 3D reconstruction, optimization, particle swarm optimization, triangulation.

ملخص

هذه المذكرة تدرس مشكل إعادة بناء مكعب بثلاثة أبعاد من أجل إيجاد العناصر الجوهرية و الخارجية للكاميرا. نعالج هذا الإشكال بتقنية التحسين، ونقوم باختيار تقنية سرب الجزيئات لحل. النتيجة المتحصل عليها في النهاية هي إعادة بناء المكعب بتطبيق عملية التثليث.

الكلمات الدلالية: التقاط التحركات، الكاميرا، إعادة بناء ثلاثة أبعاد، تقنية التحسين الاستكشافية، التحسين المستمر، التحسين بسرب الجسيمات، التثليث.

Liste des abréviations

MoCap : Motion Capture

FoV : Field of View

CCD: Charge Coupled Device

CMOS: Complementary Metal Oxide Semi-conductor.

PSO: Particule Swarm Optimization

OEP : Optimisation par Essaim Particulaire

G_{best} : Global best

P_{best} : Personal best

Liste des figures

Figure I.1: Le système d'acquisition optique	04
Figure I.2: Résultats d'acquisitions	04
Figure I.3: Objectif de la camera	05
Figure I.4: Distance focal	05
Figure I.5: La structure de capteur CCD et capteur CMOS	09
Figure I.6: Le principe de la reconstruction en 3D	10
Figure I.7: La projection perspective	11
Figure I.8: La transformation entre le repère $R_w \Rightarrow R_c$	11
Figure I.9: La transformation entre le repère $R_c \Rightarrow R_r$	12
Figure I.10: La projection orthographique	12
Figure I.11: La transformation entre le repère $R_c \Rightarrow R_r$	13
Figure II.1: Classification des méthodes de resolution de problèmes d'optimisation.	18
Figure II.2: Volée d'Anser en formation en V	19
Figure II.3: Déplacement d'une particule	20
Figure II.4: Le modèle d'optimum global de la PSO	21
Figure II.5: Le modèle d'optimum local de la PSO	21
Figure II.6: L'organigramme de la PSO	23
Figure II.7: Voisinage social, (a) anneau (avec $n = 2$), (b) rayon, (c) étoile	25
Figure II.8: Un voisinage géographique	26
Figure III.1: l'objet (cube)	31
Figure III.2: Cube en 3D	32
Figure III.3: Vue par la camera N°1	32
Figure III.4: Vue par la camera N°2	32

Figure III.5: Vue par la camera N°3	32
Figure III.6: Vue par la camera N°4	33
Figure III.7: Image prise par la camera N°1	33
Figure III.8: Image prise par la camera N°2	33
Figure III.9: Image prise par la camera N°3	33
Figure III.10: Image prise par la camera N°4	33
Figure III.11: Transformation des images en niveau de gris	35
Figure III.12 : Organigramme de notre application	38
Figure III.13 : Vitesse de déplacement d'une particule	38
Figure III.14 : La fonction de fitness	40
Figure III.15 : Reconstruction de l'objet	41

Liste des tableaux

Tableau II.1: Les avantages et les inconvénients de PSO	28
Tableau III.1: Coordonnées 3D des marqueurs	31
Tableau III.2: Intervalles des paramètres de la matrice de projection	33
Tableau III.3: Les paramètres d'optimisation	34
Tableau III.4: Coordonnées de la reconstruction	38

Sommaire

Remerciements.....	i
Résumé.....	ii
Abstract.....	iii
ملخص	iv
Abréviation.....	v
Liste des figures.....	vi
Liste des tableaux.....	ix
Sommaire.....	x
Introduction générale	01
Chapitre I: Systèmes de capture de mouvement MoCap	
I.1. Introduction.....	4
I.2. La capture de mouvement.....	4
I.2.1.Type de systèmes de capture	4
1) Les systèmes d’acquisitions mécaniques	4
2) Les systèmes d’acquisitions magnétiques	5
3) Les systèmes d’acquisitions optiques.	5
I.3. Technologies des caméras	6
I.3.1. Les composantes de la caméra	6
I.3.2. Différents types de caméras	8
I.4. Principe de la reconstruction en 3D	10
I.4.1. Types de projection	11
a) La projection perspective	11
b) La projection orthographique	11
I.4.2. Les paramètres de la camera	12

I.5 Conclusion	15
Chapitre II La technique d'optimisation PSO	
II.1. Introduction	17
II.2. Problème d'optimisation	17
II.3. Techniques d'optimisation	18
a) Les méthodes exactes	18
b) Les méthodes approchées (heuristiques)	18
II.4. Principe de la PSO	20
II.5. Formalisation	21
II.6. Algorithme de base	22
II.7. Initialisation de l'essaim	25
II.8. Topologie du voisinage	25
a) Les voisinages sociaux	25
Topologie en anneau	25
Topologie en rayon	25
Topologie en étoile	25
b) Le voisinage géographique	26
II.9. Paramètres de réglage	26
II.10. Les avantages et les inconvénients de PSO	28
Conclusion	28
Chapitre III Reconstruction 3D d'un cube (Technique PSO)	
III.1. Introduction	30
III.2. Présentation de notre travail	31
III.3. Description de l'algorithme	34
III.3.1. Initialisation de la PSO	36
III.3.1.1. Déclaration des paramètres	36
III.3.1.2. Initialisation de l'essaim	36
III.3.1.3. Initialisation des vitesses	36

III.1.4. Fonction de fitness	36
III.3.2. Modèle de voisinage	37
III.3.3. Mise à jour de la vitesse et le mouvement de la particule	37
III.3.4. Critère d'arrêt	37
III.4. Résultats de simulation	37
III.5. Reconstruction 3D d'un cube	39
III.6. Conclusion	41
Conclusion générale	43

Chapitre I

Systemes de capture de mouvement

MoCap

Chapitre 1

Systèmes de capture de mouvement

I.1 Introduction

La capture de mouvement MOCAP (*Motion Capture en anglais*), est une technologie très utilisée dans différents domaines, et désigne toutes les techniques d'acquisition de mouvement par des caméras. Ces techniques se regroupent en trois catégories; optique, magnétique et mécanique, mais la technique optique est la plus populaire [1]

Ce chapitre est structuré comme suit: dans la section I.1 nous donne un aperçu général sur les différentes techniques de capture de mouvement, ensuite la section I.2 détail la description de la caméra (Définition, identifiant sa structure, ses différents types, ses caractéristiques, Enfin dans la section I.3 nous donne le principe de la reconstruction en 3D par la camera et ses paramètres).

I.2. La capture de mouvement

La capture de mouvement est une technique permettant de capturer et d'enregistrer les mouvements d'un élément réel, pour déterminer les positions en 3D et la rotation de l'objet, et générer des mouvements réalistes (humain, animal, objet). Ces systèmes sont couramment utilisés dans plusieurs domaines: le cinéma, les jeux vidéo et la publicité [2] [3].

I.2.1.Type de systèmes de capture

Il existe divers systèmes MOCAP, on distingue trois grandes familles les plus commercialisées: Mécanique, magnétique et optique.

1) Les systèmes d'acquisitions mécaniques

Cette méthode fonctionne grâce à un exosquelette construit autour de l'élément à capter. L'exosquelette est une structure dont les angles sont des capteurs angulaires. Ces angles sont reliés par des fils à un ordinateur. Chaque capteur connaît sa position par rapport aux autres, ce qui permet à l'ordinateur de savoir les mouvements de l'ensemble de la structure [4].

Inconvénients

- l'exosquelette est souvent une armature en métal reliée par de nombreux fils Il est plus difficile de se mouvoir avec cette armature. Ce procédé n'est pas adapté pour faire des mouvements trop rapides-Interaction environnement.

2) Les systèmes d'acquisitions magnétiques

Cette méthode utilise des capteurs placés sur le corps de l'acteur, pour mesurer le champ magnétique de basse fréquence généré par un émetteur. Les capteurs ainsi que l'émetteur sont branchés à un ordinateur qui va reporter les positions des capteurs dans le champ magnétique. Un logiciel va ensuite retranscrire la position et les mouvements des capteurs dans un espace en 3D [3].

Inconvénient

- Il y a nombreuses interférences électromagnétiques. Les fréquences d'échantillonnage étant faibles, les mouvements sont obligatoirement assez lent. De plus les cascades sont rendues impossibles à causes de la fragilité des capteurs.

3) Les systèmes d'acquisitions optiques

Le procédé consiste à placer de petites sphères réfléchissantes (markers) sur les personnages réels à réaliser. Des caméras infrarouges sont disposées autour du modèle à représenter numériquement. On peut utiliser 6 à 8 caméras pour les mouvements du corps et 1 à 2 caméras pour les expressions du visage. Les sphères reçoivent un faisceau infrarouge qu'elles réfléchissent sur les caméras. La succession des centres d'inertie de chaque sphère détermine la position du mobile. L'orientation des parties mobiles du corps se calculent à partir de trois sphères attachées sur une même partie mobile formant ainsi un triangle. Plus la vitesse d'échantillonnage est élevée, plus les mouvements rapides sont réalistes. La vitesse d'échantillonnage qui offre le meilleur rendu graphique est de 200Hz. Les calculs s'effectuent à l'aide d'un ordinateur. La motion capture optique est la plus utilisée car c'est la plus rentable et la plus pratique a utilisée (absence de fils). Il n'y pas de contrainte physique sur le modèle; possibilité de faire de nombreuses cascades [5].

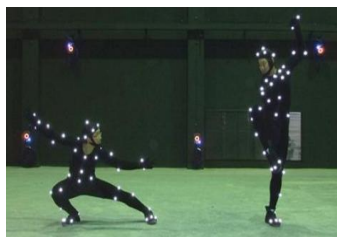


Figure I.1: Le système d'acquisition optique

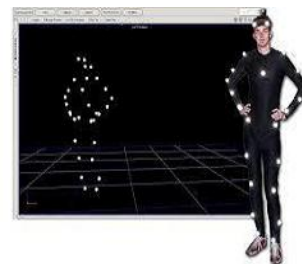


Figure I.2: Résultats d'acquisitions

Les avantages

- ❖ L'absence totale de câble ou de structure métallique
- ❖ Le champ d'acquisition est relativement grand (dépendant du nombre de caméras).
- ❖ La grande précision des données acquises dans la plupart des cas.

- ❖ Un grand nombre de marqueurs peuvent être utilisés.

Les inconvénients

- ❖ Les disparitions de marqueurs: Dans certains cas un marqueur n'est visible par aucune des caméras. Son mouvement est perdu jusqu'à sa réapparition sur les images.
- ❖ Les occlusions de marqueurs: Lorsque deux marqueurs passent l'un devant l'autre, leur identification peut être confondue. Des algorithmes de prédiction de mouvement servent à réduire ce type d'erreurs [6].

I.3. Technologies des caméras

Une caméra (*en français «chambre noire»*) est un dispositif optique de prise de vues destiné à enregistrer ou à transmettre des images photographiques successives, afin de restituer l'impression de mouvement pour le cinéma, la télévision, la recherche, la télésurveillance, l'imagerie industrielle et médicale, ou bien pour d'autres applications, professionnelles ou domestiques.[5] [6]

I.3.1. Les composantes de la caméra

La structure de la camera contient les composantes suivantes:

Capteur

Constitué d'éléments semi-conducteurs photosensibles (Silicium) qui convertissent l'énergie lumineuse (photon) en une énergie électrique (électron) et qui sont disposés sous forme matricielle ou linéaire [7].

Objectif

Constitué d'un assemblage de lentilles de verres, permettent de projeter une scène réelle sur le capteur en faisant focaliser les rayons lumineux. Chaque objectif comprend deux parties mobiles:

- *A l'avant*: une bague rotative permet de régler le point focal, c'est à dire la distance.
- *A l'arrière*: se trouve une bague crantée permettant de régler le diaphragme, c'est à dire de doser le flux de lumière qui entre.

A ces éléments s'ajoutent pour la vidéo:

- Une bague de zoom située entre la bague de point focal et celle du diaphragme, dont la fonction est de faire varier la focale, sans changer l'objectif.
- Une bague macro, placée près de la base de l'objectif, qui permet d'ajuster la netteté à de très courtes distances, généralement cette bague est associée à la bague de tirage optique qui sert au réglage technique de l'objectif en laboratoire technique [8].



Figure I.3: Objectif de la camera

Focale

La longueur focale représente la distance en millimètres du centre optique de la lentille au point où l'image se trouve dans le foyer le plus net et où le capteur de la caméra est localisé. La focale détermine l'angle de couverture d'une scène à visualiser de la caméra. Plus la focale est grande, plus l'angle de vue est étroit et plus la focale est petite, plus l'angle de vue est large. [9].

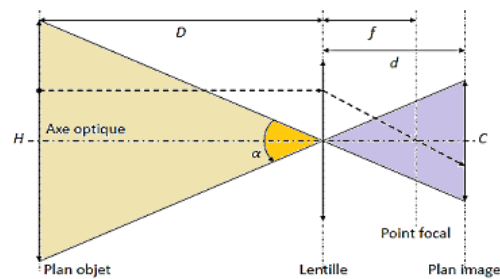


Figure I.4: Distance focale

Avec:

D : La distance entre l'objet et l'objectif (distance de travail).

d : La distance entre l'objectif et le capteur.

f : La distance entre l'objectif et le point focal (la distance focale).

H : La taille du champ de vision.

c : La taille de capteur.

α : L'angle de vue.

- D'après la loi de « Descartes » la distance f est donnée par:

$$\frac{1}{d} + \frac{1}{D} = \frac{1}{f} \Rightarrow f = \frac{d \times D}{d + D} \quad (\text{I.1})$$

- D'après la loi de « Thalès » la distance d est donnée par:

$$\frac{c}{H} + \frac{d}{D} \Rightarrow d = \frac{c \times D}{H} \quad (\text{I.2})$$

Dans le cas où l'objet est suffisamment éloigné du capteur ($d \ll D$):

$$D + d \approx D \Rightarrow f = \frac{D \times d}{d} \approx d \quad (\text{I.3})$$

Donc l'angle de vue:

$$\tan\left(\frac{\alpha}{2}\right) = \frac{c/2}{d} \Rightarrow \alpha = 2\arctan\frac{c}{2f} \quad (\text{I.4})$$

La lentille

La lentille permet de converger les rayons lumineux vers le capteur.

Mise en point

C'est le réglage de la distance entre le plan d'objet et celui du capteur pour obtenir une image nette [8].

Diaphragme

Permet de doser la quantité de lumière qui arrive sur le capteur [8].

Champ vision

Champ de vision (*FoV : Field of View*) est la surface couverte par la caméra qui est égale à celle de la matrice du capteur.

$$FoV \text{ horizontal} = \text{Résolution voulue} \times \text{largeur de la matrice} \quad (\text{I.5})$$

$$FoV \text{ vertical} = \text{Résolution voulue} \times \text{hauteur de la matrice} \quad (\text{I.6})$$

I.3.2. Différents types de caméras

Il existe aujourd'hui un grand nombre de types de caméras, en grande majorité numériques, dont les caractéristiques (taille, qualité de l'image, sensibilité, support d'enregistrement, ... etc.), dépendent du domaine et des besoins d'utilisation. [6]

Les caméras les plus utilisées dans les différentes applications sont:

- **Caméra analogique:** Ces caméras sont reliées par un câble coaxial à un téléviseur/un moniteur, où les images s'affichent; elles ont une sortie de type BNC. Elles envoient des flux continus de données (balayage) à un dispositif de stockage (enregistreur numérique).
- **Caméra numérique (caméra IP):** Une caméra numérique contrairement à sa consœur analogique, ne dispose pas de sortie coaxiale. En revanche elle dispose d'une liaison via une prise RJ 45 qui permet le raccordement au réseau informatique. Généralement c'est une unité intégrée qui contient une caméra [11].
- **Caméra infrarouge:** L'infrarouge est une lumière invisible. Elles utilisent les diodes électroluminescentes placées autour de leur lentille pour repérer les rayonnements infrarouges (ondes de chaleur) et retransmettre les images en noir et blanc.

- **Caméra CCD (*Charge Coupled Device*):** Le capteur CCD basé sur une technologie à semi-conducteur, constitué d'un ensemble de photosites (cellule photosensible de 5 à 30 μm) de silicium. Le CCD ne fait pas de distinction entre les couleurs. L'analyse des couleurs se fait via des filtres qui permettent de récupérer les signaux RVB avant traitement [13].
- **Caméra CMOS (*Complementary Metal Semiconducteur*):** les capteurs CMOS ont les mêmes principes physiques que les capteurs CCD, mais grâce à des transistors MOS placés sur chaque photosites, fonctionnant comme des interrupteurs, la conversion « électron-tension » est effectuée au niveau de chaque photosites, en plus l'intégration supplémentaire d'une logique a permis la conversion numérique immédiatement de l'image, mais cette électronique intégrée sur le capteur génère un bruit supplémentaire [Le capteur CCD et le capteur CMOS, assure la transformation des signaux lumineux en signaux électriques. En vidéo surveillance.

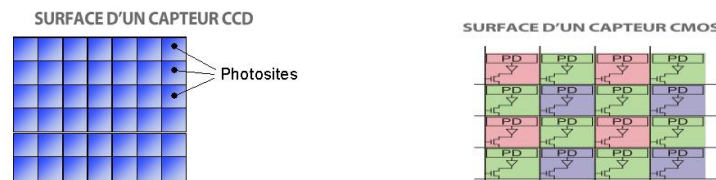


Figure I.5: La structure de capteur CCD et capteur CMOS

Les caractéristiques

- **La résolution:** La résolution est la mesure de la qualité et de la clarté de l'image obtenue, elle dépend de deux facteurs qui sont la résolution du signal vidéo provenant de la caméra et la résolution de l'écran permettant l'affichage des signaux transmis. Pour les caméras CCD, cette résolution horizontale est liée au nombre de pixels qu'il y a sur le plan horizontal. La résolution de l'image est définie en nombre de lignes [12].
- **La sensibilité:** C'est la capacité d'obtenir une image utile dans une lumière faible. Les caméras monochromes ont une sensibilité plus élevée que les modèles couleur. Elles sont également sensibles au rayonnement infrarouge de diverses sources lumineuses.
- **Signal vidéo utile:** La lumière après avoir traversé l'objectif atteint le capteur CCD qui génère un signal proportionnel au flux lumineux reçu. Ce flux est converti en signal électrique (vidéo). Le signal vidéo utile est le signal minimal indiqué dans la notice technique de la caméra pour générer une image acceptable sur un écran. Il est généralement mesuré en pourcentage du signal vidéo. [11]
- **Éclairage et positionnement:** La lumière éclaire l'objet. Une certaine quantité est absorbée et une partie est réfléchiée et se dirige vers la lentille de la caméra. La quantité

de lumière atteignant le capteur CCD dépend de l'ouverture de l'iris. Cette lumière composée de photons, frappe le capteur CCD qui transforme la lumière en charge électrique. [13]

- **Fréquence d'acquisition:** C'est le nombre d'images par seconde que peut acquérir la caméra.
- **Gain:** C'est un facteur multiplicatif permettant d'amplifier le signal image avant qu'il soit numérisé, l'augmentation du gain permet de compenser le manque de lumière mais amplifier également le bruit présent dans le signal. Il peut être fixe ou automatique. [9].

I.4. Principe de la reconstruction en 3D

La reconstruction 3D à partir d'images, (*image-based 3D reconstruction* en anglais), désigne la technique qui permet d'obtenir une représentation en trois dimensions d'un objet ou d'une scène à partir d'un ensemble d'images prises sous différents points de vue de l'objet.

Il est le processus inverse d'obtenir des images 2D à partir de scènes 3D. Permet de trouver les coordonnées des points d'un objet en 3D à partir des images obtenues par des caméras (au moins deux images ou bien deux caméras pour un seul point).

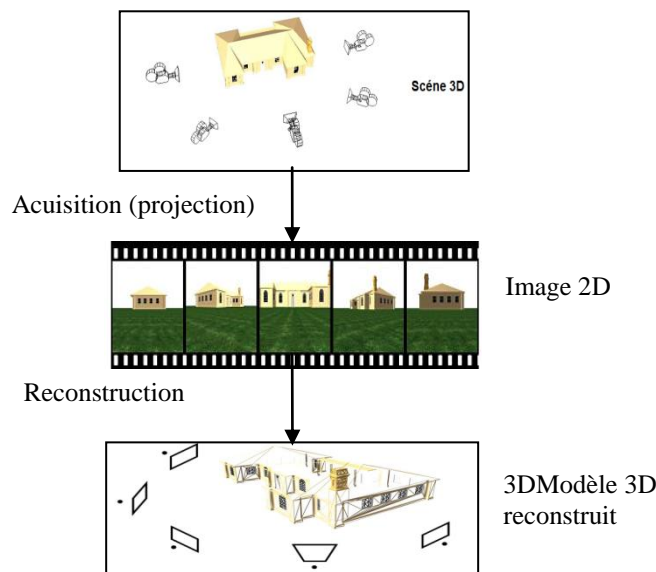


Figure I.6:Principe de la reconstruction en 3D

I.4.1. Types de projection

- a) **La projection perspective :** La projection *perspective* modélisant le modèle sténopé de la caméra Ce modèle transforme un point 3D de l'espace P en un point image p (en 2D) [15].

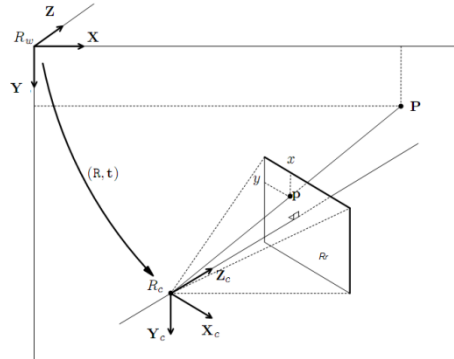


Figure I.7: La projection perspective

On peut décomposer la projection de P en trois transformations élémentaires successives:

- La transformation entre le repère du monde R_w et celui de la caméra R_c
- La transformation entre le repère caméra R_c et le repère (plan rétinien) R_r
- La transformation entre le repère capteur et le repère image.

En vision par ordinateur, on utilise souvent les coordonnées homogènes.

En 2D:

$$(p) = \begin{pmatrix} x \\ y \end{pmatrix} \quad \Rightarrow \quad \tilde{p} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (\text{I.7})$$

Coordonnées euclidiennes Coordonnées homogènes

En 3D:

$$(P) = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad \Rightarrow \quad \tilde{P} = \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (\text{I.8})$$

Coordonnées euclidiennes Coordonnées homogènes

b) La projection orthographique:

La projection *orthographique* fournit une approximation linéaire à la projection en perspective, qui est applicable pour un petit objet loin de l'observateur et à proximité de l'axe optique [9].

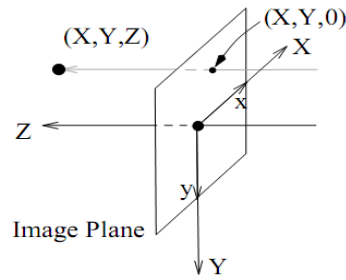


Figure I.8: La projection orthographique

L'emplacement de l'image $P(X, Y, Z)$ en 3D vers l'image $p(x, y)$ en 2D correspondante sous la projection orthographique.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \alpha \begin{pmatrix} X \\ Y \end{pmatrix} \tag{II.9}$$

α : représente le facteur d'échelle elle est constant dans la projection orthographique. est égale à : $\alpha = 1$.

I.4.2. Les paramètres de camera

Les paramètres extrinsèques

Définir deux différents repères tridimensionnels: le repère du monde (R_W) qui est le repère global contenant la scène, et le repère lié à la caméra R_c . Le changement de repère (R_W) Vers R_c . Correspond à la position et la pose de la caméra dans le repère du monde au moment de la prise de vue. Cette (pose, position) est modélisée par les paramètres extrinsèques de la caméra (une rotation (R) et une translation. (t)) [15]

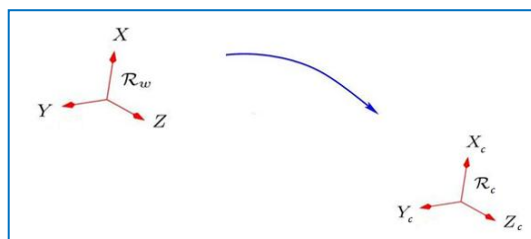


Figure I.9: La transformation entre le repère $R_w \Rightarrow R_c$

La transformation entre le repère du monde R_w et celui de la caméra R_c par l'équation:

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} = [R] \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} + t = [K] \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{I.10}$$

Avec:

$$t \Rightarrow \text{Vecteur de translation de taille } 3 \times 1. \quad t = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \quad (\text{I.11})$$

$R \Rightarrow$ Matrice de rotation de taille 3×3 représente les trois rotations des axes caméras par rapport au repère de l'environnement. Il ya deus type de rotation l'une par l'angle d'Euler en termes de trois angles de rotation autour des axes (x, y et z) par $(\theta_x, \theta_y, \theta_z)$ et l'autre par une matrice rotative au moyen d'un quaternion.

Rotation par Euler :

$$R = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_x & -\sin \theta_x \\ 0 & \sin \theta_x & \cos \theta_x \end{pmatrix} \begin{pmatrix} \cos \theta_y & 0 & \sin \theta_y \\ 0 & 1 & 0 \\ -\sin \theta_y & 0 & \cos \theta_y \end{pmatrix} \begin{pmatrix} \cos \theta_z & -\sin \theta_z & 0 \\ \sin \theta_z & \cos \theta_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (\text{I.12})$$

Rotation par quaternion :

$$R = \begin{pmatrix} 1 - q_y^2 - 2q_z^2 & 2q_xq_y - 2q_sq_z & 2q_xq_z + 2q_sq_y \\ 2q_yq_x + 2q_sq_z & 1 - q_x^2 - 2q_z^2 & 2q_yq_z - 2q_sq_x \\ 2q_zq_x - 2q_sq_y & 2q_zq_y + 2q_sq_x & 1 - q_x^2 - 2q_y^2 \end{pmatrix} \quad (\text{I.13})$$

Avec:

Une rotation de θ radian environ $[u_x \ u_y \ u_z]^T$ peut être représentée par un quaternion

$$q = [q_s \ q_x \ q_y \ q_z]^T \quad \text{où} \quad q_s = \cos(\theta/2)$$

$$\text{Et} \quad [q_x \ q_y \ q_z]^T = (\sin \theta/2) [u_x \ u_y \ u_z]^T \quad (\text{I.14})$$

K Représente une matrice 4×4 .

Les paramètres intrinsèques

Une caméra caractérisée par ses paramètres internes qui sont :

- La distance focale f (distances orthogonales du centre optique au plan image).
- Les coordonnées (x, y) du capteur est l'intersection de l'axe optique et du plan image.
- Des facteurs d'échelle (α) .

La deuxième transformation: relie le repère caméra R_c au repère capteur R_r (plan rétinien)

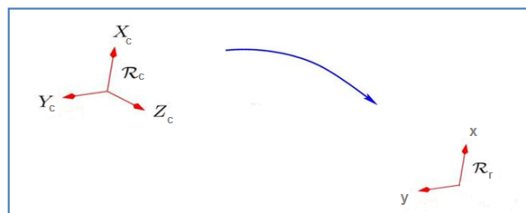


Figure I.10:La transformation entre le repère $R_c \Rightarrow R_r$.

C'est une projection perspective (matrice 3×4 , notée T) qui transforme un point 3D $(X_c, Y_c, Z_c)^T$ en un point-image $(x, y)^T$ (en unité métrique).

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} = [T] \begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} \quad (\text{I.15})$$

Où f désigne la focale de l'objectif utilisé.

L'équation (I.14) qui traduit la projection perspective s'écrit:

$$x = f \frac{X_c}{Z_c} \quad (\text{I.16})$$

$$y = f \frac{Y_c}{Z_c} \quad (\text{I.17})$$

La transformation entre le repère capteur et le repère image est la troisième et dernière transformation, décrit l'opération de conversion des coordonnées images (x, y) (en unité métrique) en coordonnées images discrètes (u, v) pixels.

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} k_x & \alpha & C_x \\ 0 & k_y & C_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = (A) \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (\text{I.18})$$

Où:

- C_x et C_y (en pixels) désignent les coordonnées de l'intersection de l'axe optique avec le plan image (centre de l'image)
- k_x et k_y désignent le nombre de pixels par unité de longueur suivant les directions x et y du capteur respectivement (dans le cas de pixels carrés)
- α : facteur de distorsion. Ce paramètre est désigné par «*skew factor en anglais*». Traduit la non orthogonalité éventuelle des lignes et colonnes de l'image. On considère souvent que le « skew factor » est négligeable et l'équation (I.8) se simplifie alors de la façon suivante :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} k_x & 0 & C_x \\ 0 & k_y & C_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = (A) \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (\text{I.19})$$

Donc le modèle sténopé complet:

$$(X \ Y \ Z) \xrightarrow{K} (X_c \ Y_c \ Z_c) \xrightarrow{T} (x \ y) \xrightarrow{A} (u \ v) \quad (\text{I.20})$$

Cela conduit à l'équation du modèle sténopé:

$$\tilde{p} = \underbrace{A T}_M K \tilde{P} \quad (\text{I.21})$$

Avec:

$$M = A T = \begin{pmatrix} k_x & 0 & C_x \\ 0 & k_y & C_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} f_x & 0 & C_x \\ 0 & f_y & C_y \\ 0 & 0 & 1 \end{pmatrix} \quad (\text{I.22})$$

Où $f_x = f k_x$ et $f_y = f k_y$ désignent la focale de la caméra en nombre de pixels suivant les directions x et y respectivement. Les paramètres $(c_x \ c_y \ f_x \ f_y)$ de la matrice K sont appelés *paramètres intrinsèques* de la caméra.

Finalement le modèle sténopé est décrit par les paramètres intrinsèques et les paramètres extrinsèques.

Conclusion

Dans ce chapitre nous avons présenté des généralités sur la capture des mouvements (MOCAP) avec et ses différents types et techniques d'acquisition. Ensuite nous avons donné la structure d'une caméra (ces types, caractéristiques, paramètres extrinsèques et intrinsèques), puis nous avons présenté une généralité sur les techniques de reconstruction 3D à partir de plusieurs vues.

Chapitre II

La technique d'optimisation *PSO*

Chapitre II

La technique d'optimisation PSO

II.1. Introduction

Les problèmes d'optimisation occupent actuellement une place importante dans la communauté des ingénieurs, des scientifiques et des décideurs. En effet, ce genre de problèmes intervient dans leurs domaines d'activité qui sont très divers, comme la conception de systèmes mécaniques, le traitement d'images et l'électronique ou la recherche opérationnelle. Ce problème est formulé en sélectionnant une ou plusieurs variables et en définissant une fonction objective consiste à trouver la meilleure solution (en minimisant ou maximisant les fonctions objectifs du problème posé) par rapport à tous les paramètres du problème. Le choix d'une technique d'optimisation efficace capable de produire une solution optimale en un temps de calcul raisonnable est la principale difficulté à laquelle est confronté un décideur [12].

L'optimisation par essaim particulaire (*OEP*, ou *PSO en anglais*) est une technique d'optimisation stochastique, pour les fonctions non-linéaires, basée sur la reproduction d'un comportement social, dérive de la famille d'algorithmes métaheuristiques pour explorer l'espace de recherche d'un problème quelconque est apparue en 1995 par Russel Eberhart (ingénieur en électricité) et James Kennedy (socio-psychologue). Ils cherchent à simuler la capacité des oiseaux à voler de façon synchrone et leur aptitude à changer brusquement de direction tout en restant en une formation optimale [15].

Dans ce chapitre, nous présentons des définitions liées aux problèmes d'optimisation, tout en mettant l'accent sur la technique PSO. Nous exposons aussi le principe et les formules mathématiques de cette technique, puis nous décrivons ces paramètres de réglage et quelques avantages et inconvénient de la PSO.

II.2. Problème d'optimisation

Un problème d'optimisation au sens général, est défini par un ensemble de variables. L'ensemble des solutions possibles du problème forme l'espace de recherche X , où chaque dimension correspond à une variable. L'espace de recherche X est fini, puisque le décideur précise exactement le domaine de définition de chaque variable, entre autres pour des raisons de temps de calcul. Suivant le problème posé, nous cherchons à minimiser ou maximiser la fonction objective F . [2][3]

Trouver $x \in X = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix}$ qui sont appelées "variables d'optimisation" qui minimise (ou

maximise) la fonction objective $F(x)$.

Sous les contraintes $g_j(x)$, $j = 1 \dots m$

m :est le nombre des contraintes.

Résoudre ce problème revient à chercher des points de minimum (ou maximum) local (ou global) de la fonction F .

II.3. Techniques d'optimisation

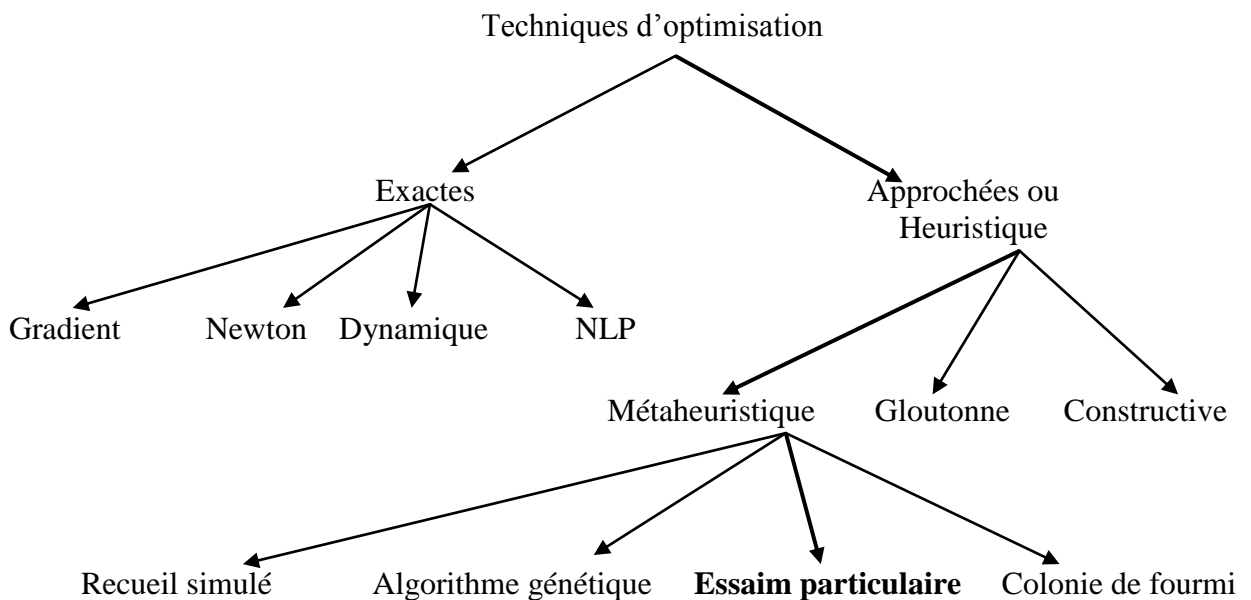


Figure II.1: Classification des méthodes de résolution de problèmes d'optimisation.

Les méthodes de résolution de problèmes ont été classées en deux catégories: les méthodes exactes et les méthodes approchées.

- 1) **Les méthodes exactes:** sont connues par le fait qu'elles garantissent l'optimalité de la solution mais elles demandent des coûts de recherche (temps de calcul et espace mémoire).[17]
- 2) **Les méthodes approchées (heuristiques):** sont connues par le fait qu'elles ne garantissent pas l'optimalité de la solution (elles fournissent des solutions de bonne qualité et des fois optimales), mais elles nécessitent des coûts de recherche raisonnables. Les méthodes approchées sont plus pratiques dans le cas où on cherche une solution de

bonne qualité dans un bref délai. Elles permettent la résolution des problèmes d'optimisation de différentes tailles en un temps de calcul raisonnable [4]

On retrouve parmi ces heuristiques, certains algorithmes qui possèdent un principe générique adaptable et qui s'applique donc à plusieurs problèmes d'optimisation. On les appelle des métaheuristiques:

Une métaheuristique: est un processus de génération itérative, qui permet de guider une heuristique subordonnée par la combinaison de divers concepts tels que l'exploitation et l'exploration de l'espace de recherche, ainsi que des stratégies d'apprentissage qui sont utilisées pour structurer efficacement l'information afin de déterminer des solutions d'excellente qualité. Les métaheuristiques sont en général non déterministes et ne donnent aucune garantie d'optimalité mais ils peuvent faire usage de l'expérience accumulée durant la recherche de l'optimum, pour mieux guider sa suite de la procédure de recherche [8].

Les métaheuristiques classées en deux catégories: les méthodes à base de solution unique et les méthodes à base de population de solutions.

- **Les méthodes à base de solution unique:** se basent généralement sur une recherche locale (recherche par voisinage).
- **Les méthodes à base de population de solutions:** se basent sur une recherche globale sur tout l'espace de recherche.

L'optimisation par essaim particulière est un algorithme évolutionnaire qui utilise une population de solutions candidates pour développer une solution optimale au problème, qui dérive de la descente stochastique, entre dans cette famille d'algorithmes. Elle s'inspire fortement des relations grégaires des oiseaux migrateurs qui doivent parcourir des longues distances et qui doivent donc optimiser leurs déplacements en termes d'énergie dépensée, comme par exemple la formation en V [5]



Figure II.2: Volée d'Anser en formation en V

II.4. Principe de la P.S.O

L'optimisation par essaim de particules se base sur un ensemble d'éléments modélisés en vecteurs, qui sont en mouvement dans tout l'espace de recherche; on les appelle *particules*. Les particules qui constituent un essaim, sont réparties dans l'espace de recherche aléatoirement et de façon homogène, elles se déplacent dans le but de se positionner sur des solutions optimales. Donc, pour appliquer la PSO il faut définir un espace de recherche constitué de particules et une fonction objective à optimiser selon le problème traité, où chaque particule représente une solution potentielle à cette fonction.[16].

Chacune de ces particules est dotée:

- D'une position, c'est-à-dire ses coordonnées dans l'ensemble de définition.
- D'une vitesse qui permet à la particule de se déplacer
- D'un voisinage, c'est-à-dire un ensemble de particules qui interagissent directement sur la particule.

A tout instant, chaque particule connaît:

- Sa meilleure position visitée. On retient essentiellement la valeur du critère calculée ainsi que ses coordonnées.
- La position du meilleur voisin de l'essaim qui correspond à l'ordonnancement optimal.
- La valeur qu'elle donne à la fonction objectif car à chaque itération il faut une comparaison entre la valeur du critère donnée par la particule courante et la valeur optimale [5]

Le déplacement d'une particule est influencé par les trois composantes suivantes:

- Une composante d'inertie: la particule tend à suivre sa direction courante de déplacement.
- Une composante cognitive: la particule tend à se diriger vers le meilleur site par lequel elle est déjà passée.
- Une composante sociale: la particule tend à se fier à l'expérience de ses congénères et, ainsi, à se diriger vers le meilleur site déjà atteint par ses voisins [6].

Le schéma de principe du déplacement d'une particule est le suivant:

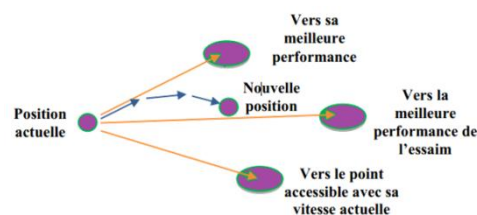


Figure II.3: Déplacement d'une particule

Donc il ya deux modèle de PSO; l'optimum global (où la notation est g_{best} et l'optimum local (où la notation est p_{best}) qui diffèrent dans la taille de leurs voisinages.

- L'optimum *global*: C'est où la position de chaque particule est influencée par la meilleure particule dans l'essaim entier, donc la même information sociale obtenue par toutes les particules dans l'essaim.

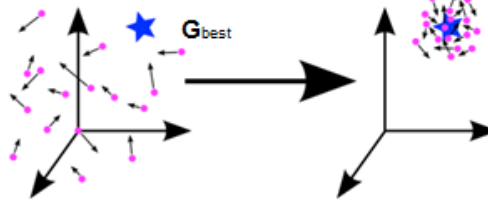


Figure II.4:Le modèle d'optimum global de la PSO

- L'optimum *local* : chaque particule est influencée par la meilleure particule choisie de son voisinage, ici l'information sociale échangée dans le voisinage de la particule.

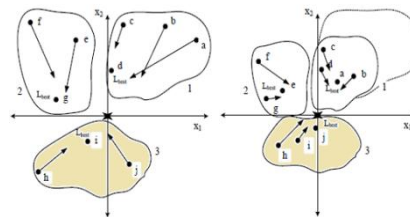


Figure II.5:Le modèle d'optimum local de la PSO

II.5. Formalisation

La version de base de la PSO peut facilement être formalisée et programmée. L'espace de recherche est de dimension D . La position courante d'une particule dans cet espace à l'instant t influencé par:

- Le nombre de particules de l'essaim.
- La vitesse maximale d'une particule, notée (V_{max}).
- La topologie et la taille du voisinage d'une particule qui définissent son réseau social
- L'inertie d'une particule, notée ω
- Les coefficients de confiance, notés $c1$ et $c2$, qui pondèrent le comportement conservateur.

Chaque particule représente une solution potentielle dans l'espace de recherche. La nouvelle position d'une particule est déterminée en fonction de sa propre valeur et celle de ses voisins.

Dans un espace de recherche de dimension D , la particule i de l'essaim est modélisée à:

$$v_{ij}(t+1) = \omega v_{ij}(t) + c_1 rand_1 [p_{besti,j}(t) - x_{ij}(t)] + c_2 rand_2 [g_{besti,j}(t) - x_{ij}(t)] \quad (II.1)$$

Avec:

$v_{ij}(t)$: est la vitesse de particule i à l'instant t .

$x_{ij}(t)$: est la position de particule i à l'instant t .

$p_{besti,j}$: La position de la meilleure solution par laquelle elle est passée.

$g_{besti,j}$: La position de la meilleure solution connue de son voisinage.

Le paramètre ω est une constante, appelée *coefficient d'inertie*, c_1 et c_2 sont deux constantes appelées *coefficients d'accélération*. Sont des coefficients constants, fixés par l'utilisateur. Les nombres $rand1$ et $rand2$: sont des nombres aléatoires tirés à chaque itération, dans l'intervalle $[0,1]$.

- $g_{best}(t)$: est la meilleure solution trouvée jusqu'à l'instant t
- $p_{best}(t)$: est la meilleure solution trouvée par la particule i .
- $c_1 rand_1 [p_{besti,j}(t) - x_{ij}(t)]$ (*Influence personnelle*): Correspond à composante conservatrice du déplacement de la particule.
- $c_2 rand_2 [g_{besti,j}(t) - x_{ij}(t)]$ (*Influence social*): Correspond à la composante sociale du déplacement de la particule.

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \quad (II.2)$$

Correspond la nouvelle position (mise à jour) au temps t de la particule i .

II.6. Algorithme de base

L'algorithme PSO commence par initialiser la taille de l'essaim ainsi que les différents paramètres, affecter à chaque particule une position et une vitesse initiales et initialiser les(p_{best}). Ensuite, calculer les fitness des particules afin de pouvoir calculer la meilleure position trouvée par l'essaim (g_{best}). À chaque itération du processus de la recherche, les particules se déplacent en fonction des équations (II.1) Leurs fitness sont calculées, les p_{best} et la g_{best} sont mises à jour. Le processus est répété jusqu'à la satisfaction du critère d'arrêt.

Nous permettent de construire l'organigramme suivant:

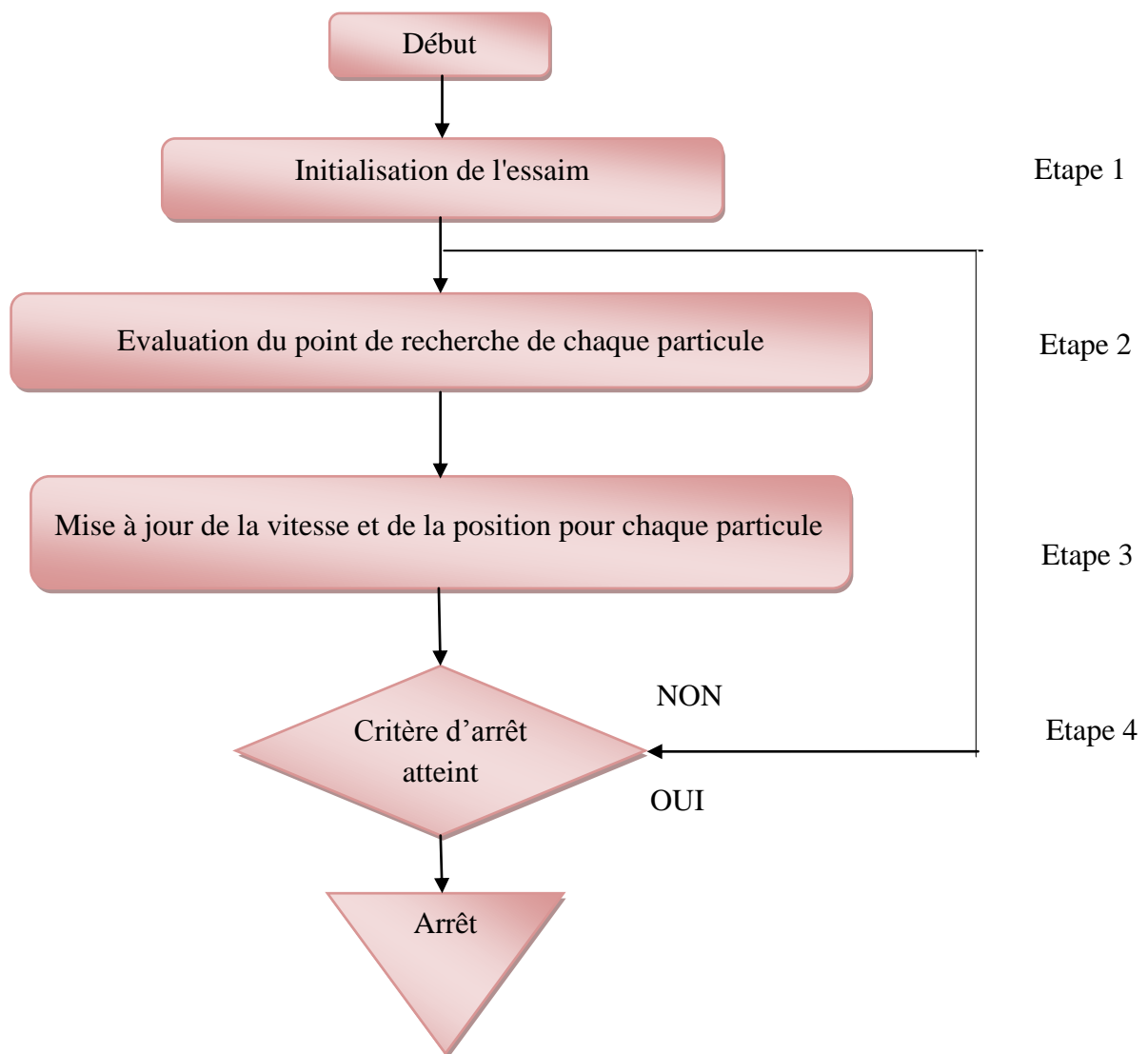


Figure II.6: L'Organigramme de la PSO

L'algorithme de base est:

Déclarations

N: nombre de particule

Nombre d'itération maximale

Les coefficients d'accélération, et d'inertie

Initialisation

Des positions et des vitesses de chaque particule d'une façon aléatoire.

Pour chaque particule i , $\vec{p}_{best_{i,j}} = \vec{x}_{i,j}(t)$

Traitements

Tant que le critère d'arrêt n'est pas atteint **faire**

Pour $i = 1$ à N **faire**

Déplacement de la particule à l'aide de (II.1) et (II.2)

Evaluation des positions

Si $f(\vec{x}_{i,j}) < f(\vec{p}_{best_{i,j}})$

$p_{best_{i,j}} = \vec{x}_{i,j}$

Fin Si

Si $f(p_{best_i}) < f(g_{best_i})$

$g_{best_i} = p_{best_i}$

Fin Si

Fin Pour

Fin Tant que

L'algorithme PSO produit des résultats sensiblement différents selon les paramètres qu'on lui injecte. Un bon compromis entre exploration et exploitation de l'espace de recherche permet d'obtenir une bonne performance de l'algorithme (convergence, temps d'exécution,...etc.).

- *Exploration*: La tendance de l'algorithme à explorer (tester) des régions distinctes de l'espace de recherche.
- *Exploitation*: La tendance de l'algorithme à axer la recherche non loin des solutions considérées comme intéressante.

II.7. Initialisation de l'essaim

Les positions des particules ainsi que leurs vitesses initiales doivent être initialisées aléatoirement selon une loi uniforme sur $[0,1]$. Cependant, en ce qui concerne la position des particules.[5].

II.8. Topologie du voisinage

Le voisinage d'une particule est le sous-ensemble de particules de l'essaim avec lequel il a une communication directe. Ce réseau de rapports entre toutes les particules est connu comme la sociométrie, ou la topologie de l'essaim [7]

Il existe deux principaux types de voisinage:

1) Les voisinages sociaux

Les voisins sont définis à l'initialisation et ne sont pas modifiés ensuite. C'est le voisinage le plus utilisé, pour plusieurs raisons on peut citer:

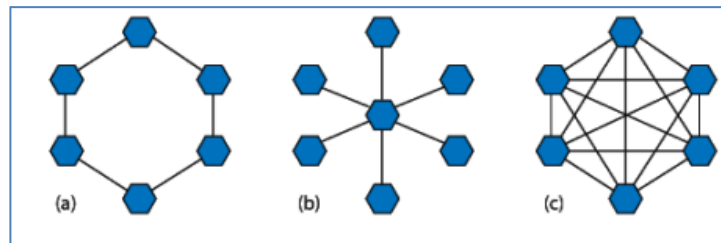


Figure II.7: Voisinage social, (a) anneau (avec $n = 2$), (b) rayon, (c) étoile.

a) Topologie en anneau

Chaque particule est reliée à n autres particules (en général, $n = 2$), et tend à converger vers la meilleure de son voisinage. C'est la topologie la plus utilisée.

b) Topologie en rayon

Les particules du voisinage sont connectées à une particule centrale. Seule cette dernière est en mesure de modifier sa position vers la meilleure; en cas d'amélioration, l'information est propagée.

c) Topologie en étoile

Toutes les particules sont reliées entre elles, on parle de réseau social complet. On considère la meilleure solution du voisinage comme l'optimum g_{best} [3].

2) Le voisinage géographique

Les voisins sont considérées comme les particules les plus proches. Cependant, à chaque itération, les nouvelles voisins doivent être recalculées à partir d'une distance prédéfinie dans l'espace de recherche.

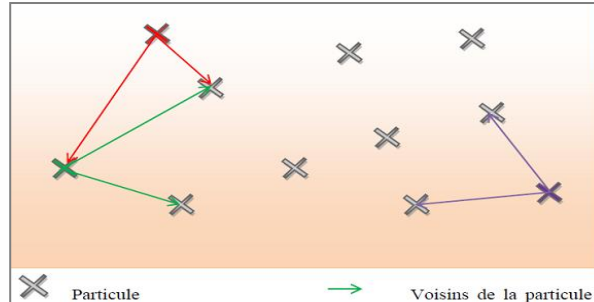


Figure II.8:Un voisinage géographique

II.9. Paramètre de réglage

Les paramètres utilisés sont:

- Le nombre de particule de l'essaim noté N .
- Vitesse maximum d'une particule v_{max} .
- Les coefficients d'accélération, d'inertie et de constriction.
- Le critère d'arrêt.

Nombre de particules

La quantité de particules allouées à la résolution du problème dépend essentiellement de deux paramètres: la taille de l'espace de recherche et le rapport entre les capacités de calcul de la machine et le temps maximum de recherche. Il n'y a pas de règle pour déterminer ce paramètre, faire de nombreux essais permet de se doter de l'expérience nécessaire à l'appréhension de ce paramètre [6].

La vitesse maximale

Certaines particules peuvent sortir de l'espace de recherche si leur vitesse est trop importante, on peut fixer une vitesse maximale afin de pallier ce problème; ainsi la vitesse d'une particule $v_i(t)$ est égale à:

$$v_{\max} \quad \text{si } v_i(t) > v_{\max}$$

$$-v_{\max} \quad \text{si } v_i(t) < -v_{\max}$$

Les coefficients

➤ Coefficient de confiance

Deux autres paramètres importants sont les coefficients de confiance qui sont nommés précédemment. (c_1 et c_2) Ils permettent de pondérer les tendances des particules c_1 exprime pour une particule le niveau de confiance qu'elle a en soi, tandis que c_2 exprime le niveau de confiance qu'elle a en ses voisines. Le choix de ces coefficients est très important pour la convergence de l'algorithme.

La version originale de PSO utilise la valeur ($c_1 = c_2 = 2$) pour ces deux coefficients

➤ Facteur d'inertie

Le facteur d'inertie ω (décroissant au fil du temps) permet de pondérer l'influence de la vitesse actuelle d'une particule sur la valeur de sa vitesse suivante et du coup de contrôler son comportement entre une exploration locale et une exploration globale de l'espace de recherche. Une valeur élevée de ($\omega > 1$) oriente la particule vers une large exploration de l'espace de recherche alors qu'une petite valeur ($\omega < 1$) centre la recherche sur un espace limité. Dans la plus part des cas les chercheurs utilisent la valeur 0,5

➤ Coefficient de constriction

Afin d'éviter que les particules ne se déplacent trop rapidement dans l'espace de recherche, passant éventuellement à côté de l'optimum, il peut être nécessaire de fixer une vitesse maximale (V_{max}) pour améliorer la convergence de l'algorithme. Cependant, on peut s'en passer si on utilise un coefficient de constriction κ introduit par *Maurice CLERC* et qui permet de resserrer l'hyper espace de recherche.

L'équation de la vitesse devient alors:

$$v_{ij}(t + 1) =$$

$$\kappa v_{ij}(t) + \rho_1 rand_1 [p_{best_{i,j}}(t) - x_{ij}(t)] + \rho_2 rand_2 [g_{best_{i,j}}(t) - x_{ij}(t)] \quad (II.3)$$

$$\kappa = \frac{2}{\rho - 2 + \sqrt{\rho^2 - 4\rho}} \quad \text{Et} \quad \rho = \rho_1 + \rho_2$$

Dans la majorité des cas on utilise $\rho = 4$ et $\rho_1 = \rho_2$

$\kappa \Rightarrow$ Approximativement égal à 0.7298.

Critères d'arrêt

La convergence vers la solution optimale globale n'est pas garantie dans tous les cas même si les expériences dénotent la grande performance de la méthode. De ce fait, il est fortement conseillé de doter l'algorithme d'une porte de sortie en définissant un nombre maximum d'itération (Critères d'arrêt). L'algorithme doit alors s'exécuter tant que l'un des critères de convergence suivant n'a pas été atteint [12]:

- Nombre d'itération maximal a été atteint.
- la variation de la vitesse est proche de zéro.
- La fitness de la solution est suffisante.

II.10. Les avantages et les inconvénients de PSO

Tableau II.1: Les avantages et les inconvénients de la PSO

La technique d'optimisation PSO	
Avantages	<ul style="list-style-type: none"> • Technique appliquée dans différents domaines de recherche scientifique par ce que elle est basé sur l'intelligence. • Une simple implémentation. • La vitesse de recherche est très rapide. • Chaque particule garde en mémoire ses anciennes meilleures valeurs ainsi que ceux de ces meilleurs voisins. • Plus efficace en maintenance de la diversité de l'essaim. • Très peu de paramètres. • Algorithme de recherche global très efficace.
Inconvénients	<ul style="list-style-type: none"> • Souffre de la convergence prématurée, qui résulte quand des individus attirent l'essaim vers un optimum local, ou une mauvaise initialisation de l'essaim. • Trouver les meilleures valeurs de certains paramètres n'est pas une tache facile et elle peut différer d'un problème à un autre.

Conclusion

Dans ce chapitre nous avons présenté des généralités sur les problèmes d'optimisation et les techniques de résolution, puis nous avons étudié la technique PSO (principe, la formalisation, l'algorithme de base et enfin quelques avantages et inconvénients de la PSO).

Nous allons utiliser cette étude dans le chapitre suivant pour l'application de la reconstruction 3D d'un cube.

Chapitre III

Reconstruction 3D

d'un cube

(Technique PSO)

Chapitre III

Reconstruction 3D d'un Cube (Technique PSO)

III.1. Introduction

De nos jours le champ d'application du domaine capture de mouvement optique est assez large et dont le but est la détermination de la pose et la position 3D d'une personne ou pour l'animation et la reconstruction 3D des objets réels. Le processus de création de modèles 3D à partir de données capturées par des caméras est une reconstruction en trois dimensions d'un objet, qui permet d'estimer la structure 3D de la scène observée en se basant sur le principe de triangulation géométrique. Elle consiste à calculer les coordonnées (X, Y, Z) à partir des coordonnées images 2D [6]. Dans notre application, qui est une reconstruction 3D d'une forme d'un cube, on se base sur l'algorithme d'optimisation par essaim particulaire, pour résoudre ce problème.

III.2 Présentation de notre travail

Nous avons essayé de résoudre ce problème par une simulation sous Matlab (version V.7.10 (32 bits) installé sur un PC utilisant un système d'exploitation Windows 7), à l'aide des instructions prédéfinies (par exemple : campos, campan), la première qui permet de positionner une caméra virtuelle ou de récupérer sa position et la deuxième de régler la pose de la caméra selon deux angles (Azimut et élévation). Dans cette simulation, l'objectif est de récupérer la matrice de projection (2D-3D pour chaque caméra utilisée), par la technique PSO, qui permet d'optimiser un grand nombre de paramètres intrinsèque et extrinsèques, la distance focale, le coefficient de distorsion...etc. [10]

Pour la partie simulation, nous avons suivi les étapes suivantes :

1. Création d'un cube de dimension $50 \times 50 \times 50$, avec des marqueurs (8 marqueurs) de couleurs différentes, placés aux coins du cube;

2. Placer les caméras par les instructions de Matlab à des différents endroits et capturé des images 2D du cube (sans l'utilisation du zoom).
3. Transformation du cube en niveau de gris et codage des couleurs des marqueurs selon leurs couleurs. Cette étape nous a permis d'identifier le marqueur à partir de sa couleur en niveau de gris.
4. Récupération des coordonnées 2D (u, v) des marqueurs, à l'aide du codage utilisé.
5. Utilisation de la PSO pour créer les matrices de projection et chercher la solution optimale.
6. Récupération des coordonnées 3D à partir des matrices de projection générées par l'optimisation.
7. Affichage du résultat.

Pour la partie pratique, nous avons fabriqué un cube (vide) et nous avons utilisé un calendrier de forme triangulaire pour une reconstruction 3D. Les photos sont prises par une caméra de téléphone portable, avec une résolution de 3.1MP, et une MIRE imprimé sur un papier ordinaire de format A4 pour l'étape de calibrage.

Dans ce qui suit, nous allons détailler les étapes de la partie simulation.

- **Création de l'objet**

Nous avons choisi un objet de forme simple (un cube), où nous avons placé à ces extrémités des marqueurs colorés (Rouge, Bleu, jaune, vert, maron, bleu claire, violet, orange), qui représentent respectivement les marqueurs numérotés (1,2,3,4,5,6,7,8).

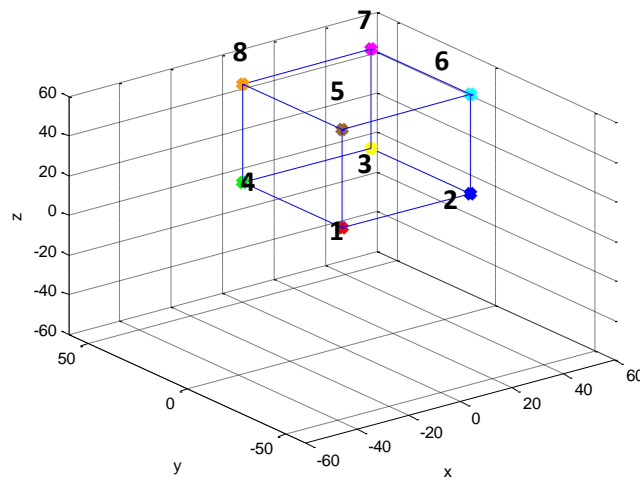


Figure III.1: l'objet (cube)

Puis nous avons dessiné (d'une manière structurée) les lignes pour montrer la forme de l'objet. Les coordonnées des points (marqueurs) du cube, sont illustrées dans le tableau III.1:

Positions	P1	P2	P3	P4	P5	P6	P7	P8
X	0	50	50	0	0	50	50	0
Y	0	0	50	50	0	0	50	50
Z	0	0	0	0	50	50	50	50

Tableau III.1: Coordonnées 3D des marqueurs

- **Types de projection**

La projection est la loi géométrique selon laquelle les points de l'espace réel 3D se projettent sur la surface du capteur. Nous distinguons deux modèles de projection, la projection perspective et la projection orthographique. Nous avons travaillé dès de début avec la projection orthographique, qui est la projection par défaut dans l'espace de travail de Matlab. Quatre caméras ont été placées dans différentes positions et poses, afin de récupérer des images en 2D (figures III.3 à III.6).

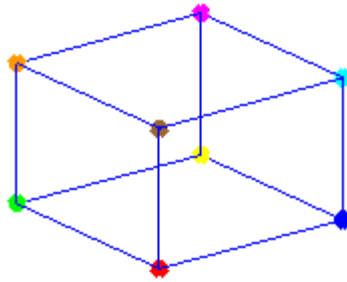


Figure III.2: Cube en 3D

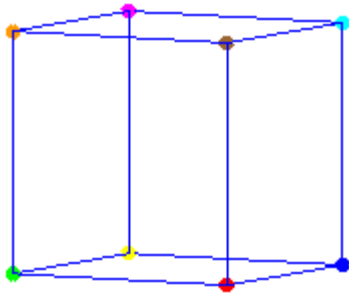


Figure III.3: Vue par la camera N°1

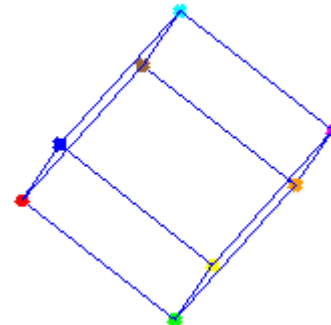


Figure III.4: Vue par la camera N°2

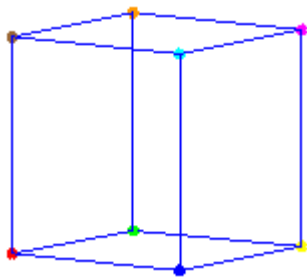


Figure III.5: Vue par la camera N°3

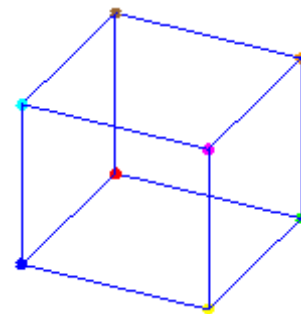


Figure III.6: Vue par la camera N°4

Le choix de convertir les couleurs des marqueurs en niveau de gris, nous a permis aussi de minimiser la quantité de données (compression).

Chaque pixel en niveau de gris est codé par une seule valeur, tandis qu'il est codé par trois niveaux de gris pour un pixel en couleurs (Matrice RGB).

Le résultat de la capture (projection en 2D) par les quatre caméras, est donné par les figures III.7 à III.10.

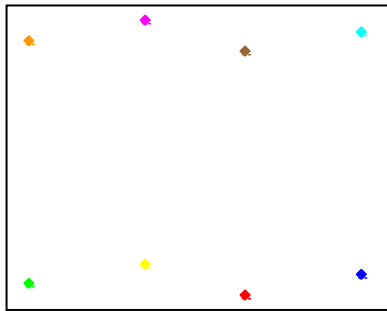


Figure III.7: Image prise par la camera N°1

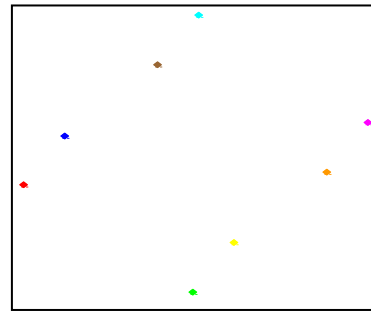


Figure III.8: Image prise par la camera N°2

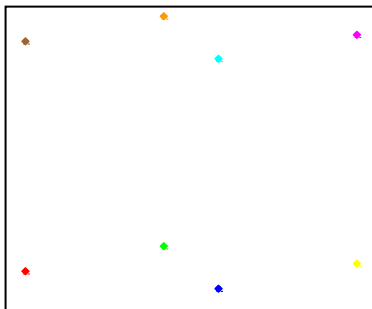


Figure III.9: Image prise par la camera N°3

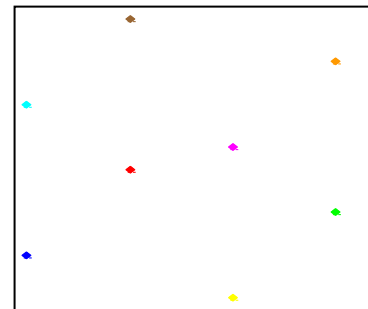
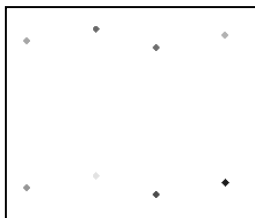
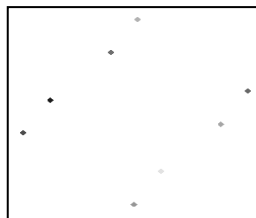


Figure III.10: Image prise par la camera N°4

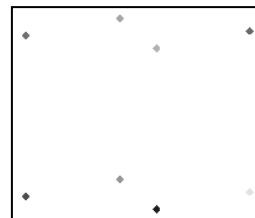
Après la transformation en niveau de gris, nous avons obtenu les images suivantes :



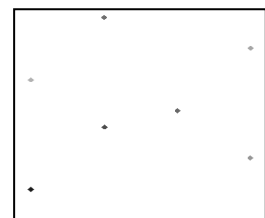
Camera 1



Camera 2



Camera 3



Camera 4

Figure III.11: Transformation des images en niveau de gris

Le choix des paramètres de la matrice de projection est aléatoire. L'utilisation de l'équation II.16, nous a permis de prendre une valeur quelconque, mais dans considéré pour chaque paramètre, suivant le tableau III.2.

$$v(t) = v_{max} + (v_{max} - v_{min}) * \rho \quad \rho \in [0,1] \quad (III.1)$$

Paramétrés	Distance focal		Centre d'image		Facteurs d'échelle	Translation			Rotation			
	f_x	f_y	c_x	c_y		α	t_x	t_y	t_z	q_s	q_x	q_y
v_{min}	85	85	-12	-12	-0.2	-15	-15	100	-1	-1	-1	-1
v_{max}	115	115	12	12	0.2	15	15	430	1	1	1	1

Tableau III.2: L'intervalle des paramètres de matrice de projection

- **Récupération des coordonnées 2D**

Pour trouver les coordonnées des points (x, y) en 2D nous avons lire les codages des pixels dans chaque image (au niveau de gris)

$P_{image(i,j)} = [x_{ij} \ y_{ij} \ 1]$: représente les coordonnées des images en 2D.

$P_i = [X_i \ Y_i \ Z_i]$: $i = 1 \dots n$. et $j = 1 \dots m$ représente les coordonnées des points en 3D.

n représente le nombre des points (8 points) et m le nombre des caméras (4 caméras).

Lorsque nous obtenons les valeurs des paramètres des caméras:

$$(M_j) = \begin{pmatrix} f_{xi} & \alpha_i & c_{xi} \\ 0 & f_{yi} & c_{yi} \\ 0 & 0 & 1 \end{pmatrix} * [R_i \ t_i] \in R^{3*4} \quad (III.2)$$

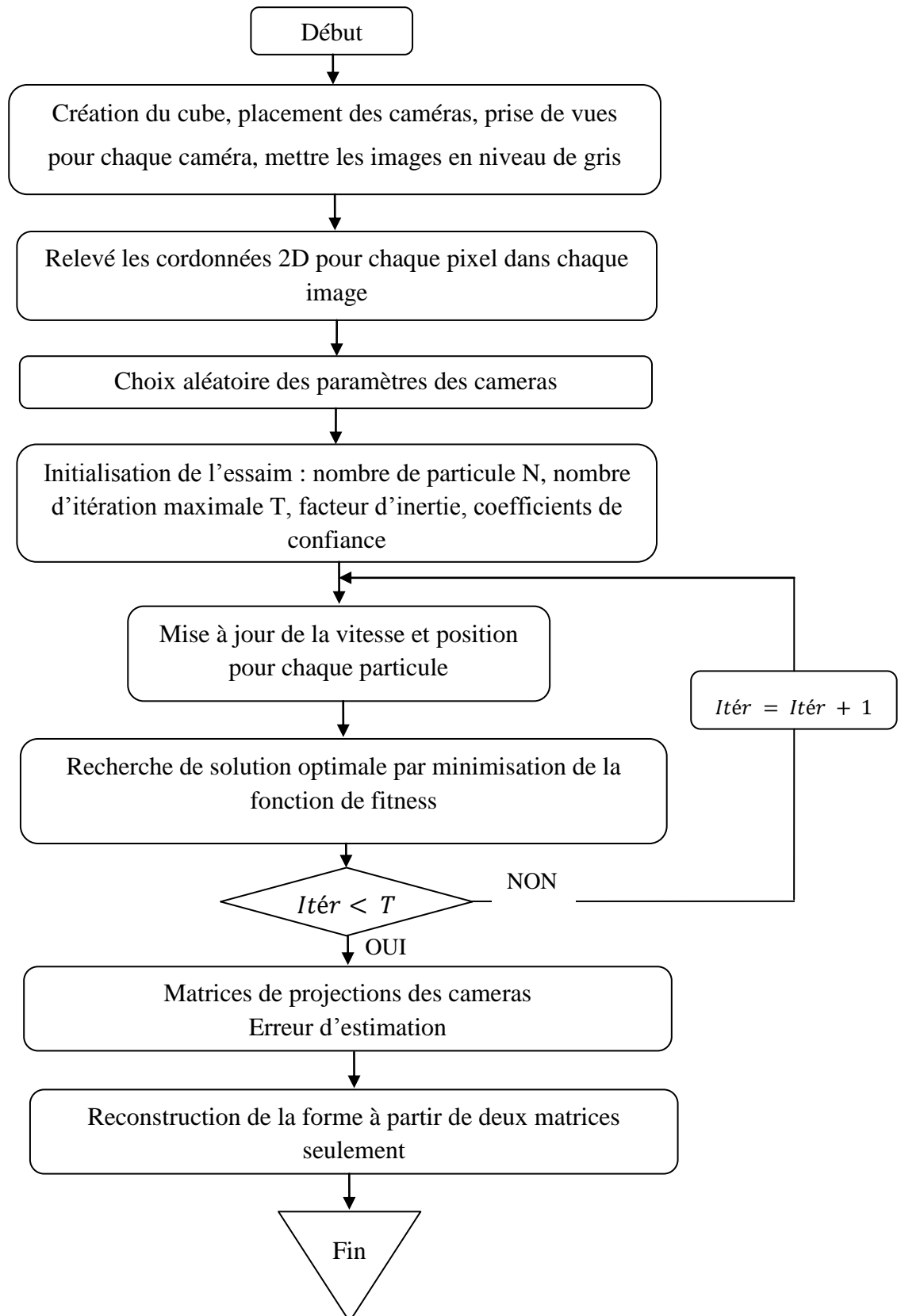
Les points estimés:

$$P_{estimé(i,j)} = (M_{i,j}) P_i \quad (III.3)$$

$$P_{estimé(i,j)} = [\hat{u}_{ij} \ \hat{v}_{ij} \ 1] = \frac{1}{z_{ij}} M_j \quad (III.4)$$

III.3 Description de l'algorithme

Comme nous l'avons vu dans le chapitre précédent, la technique PSO a été utilisée efficacement pour la résolution des problèmes difficile grâce à sa rapidité et sa simplicité d'implémentation. Notre application peut être structurée selon l'organigramme suivant :

**Figure III.12** : Organigramme de notre application

III.3.1. Initialisation de la PSO

La PSO est un algorithme de population, qui commence avec une initialisation aléatoire de l'essaim et une déclaration des paramètres de réglage.

1. Déclaration des paramètres

Le choix des coefficients de confiance reste en grande partie empirique, mais suit une certaine logique, afin d'attirer les particules vers la moyenne entre p_{best} et g_{best} , on peut les fixer à une valeur choisie et égale à 2.05 [9]

Le coefficient d'inertie varie généralement dans l'intervalle [0,1.4], afin de trouver un compromis entre l'exploration global et local, on peut le fixer à la valeur 0,5. La taille d'essaim a été fixée pour 200 particules.

2. Initialisation de l'essaim

Comme il a été exprimé dans le chapitre II, dans l'algorithme PSO chaque particule représente une solution potentielle pour le problème d'optimisation. Ces solutions s'améliorent au cours du temps. Le choix des valeurs initiales de notre essaim est donné par l'équation II.16, sous forme de vecteur de 12 paramètres, comme suit :

$$[f_x \ f_y \ \alpha \ c_x \ c_y \ t_x \ t_y \ t_z \ q_s \ q_x \ q_y \ q_z]$$

La dimension du problème est :

$$m \times n = 12 \times 4 = 48$$

m : Nombre de paramètres

n : Nombre de caméras

3. Initialisation des vitesses

L'initialisation de chaque particule s'accompli par l'initialisation aléatoire de la vitesse, qui peut être nulle.

4. Fonction de fitness

Nous avons choisi un processus d'optimisation par minimisation d'une fonction de fitness. Il s'agit d'une fonction qui calcul l'erreur entre les pixels des points dans les images 2D et les pixels estimés par les matrices de projection constituées à partir des paramètres.

Alors notre fonction de fitness est basée sur la minimisation de la distance suivante :

$$Ftnes = Min\{E\} = Min \left\{ \sum_{i=1}^m \sum_{j=1}^n |dist(P_{estimé(i,j)} - P_{image(i,j)})|^2 \right\} \quad (III.5)$$

III.3.2. Modèle de voisinage

Chaque particule dans l'essaim améliore sa position dans l'espace selon sa propre expérience et celle de son voisinage, le modèle de voisinage g_{best} est utilisé dans notre application, où toutes les particules de l'essaim ont la même information sociale [11].

III.3.3. Mise à jour de la vitesse et le mouvement de la particule

La modification du comportement de chaque particule se base sur la mise à jour de la vitesse et de la position ; à chaque itération la particule combine linéairement par les trois tendances (suivre sa direction courante de déplacement, ou diriger vers le meilleur site par lequel elle est déjà passée, et enfin vers le meilleur site déjà atteint par ses voisins), afin de trouver sa nouvelle vitesse et effectuer son mouvement par conséquence.

III.3.4. Critère d'arrêt

La procédure de recherche prend sa fin, lorsque le nombre courant d'itération devient supérieur ou égal à la valeur maximale d'itérations $T = 1000$, dans ce cas la valeur g_{best} est considérée comme une solution optimale au problème.

III.4. Résultats de simulation

Les paramètres résultant, de la simulation sont données par le tableau III.3.

Les paramètres	Camera 1	Camera 2	Camera 3	Camera 4
f_x	89.13	126.3905	94.251	109.4878
f_y	103.77	99.256	72.9887	81.2248
α	0.0786	0.3649	-0.3278	-0.0256
c_x	5.77	3.214	-7.254	11.6578
c_y	-1.56	11.2898	7.722	5.2547
t_x	2.60	7.56	5.2589	2.1425
t_y	10.2145	4.254	-3.254	-0.2145
t_z	244.454	155.70	267.8570	60.9558
q_s	-0.0995	-0.3875	1.254	1.2356
q_x	1.9478	-0.1499	-1.0516	0.2541
q_y	1.2271	0.0872	-1.0516	1.2541
q_z	0.1817	-0.0512	-0.7827	-1.254

Tableau III.3: Les paramètres d'optimisation

Nous remarquons bien que les paramètres des matrices de projection, ne sortent des intervalles fixés au début.

Nous avons pris l'évaluation de la vitesse d'une des particules, ainsi nous avons tracé la courbe de la figure III.13.

Selon les paramètres de l'essaim choisis, pour une bonne optimisation, parmi plusieurs tests et observations, cette courbe montre bien la phase d'exploration de l'espace de recherche (ici en 10 et -8), dès le début des itérations jusqu'à à peu près l'itération 45, puis une phase d'exploitation où l'oscillation de la vitesse commence à devenir nulle.

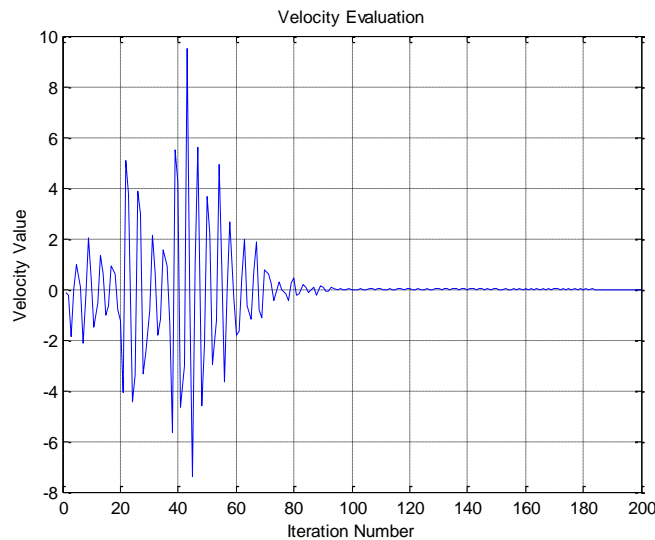


Figure III.13 : Vitesse de déplacement d'une particule

Pour mieux comprendre le comportement de l'essaim, nous avons tracé l'évaluation de la fitness sur la figure III.14. Par comparaison avec la courbe III.13, nous remarquons bien l'évaluation de la solution sous forme d'une descente de la courbe de fitness (puisque'il s'agit d'une minimisation) et cela jusqu'à l'itération 80.

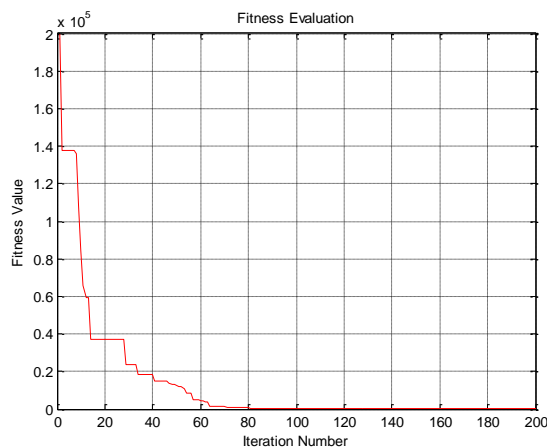


Figure III.14 : La fonction de fitness

D'après la courbe de la figure III.14, l'erreur se stabilise avant l'itération 80, à une valeur de $E = 0.2423$ [pixel]. Nous pouvons juger que cette valeur est la meilleure parmi plusieurs résultats obtenus dans des anciens tests.

III.5. Reconstruction 3D d'un cube

Pour la reconstruction d'un cube nous avons utilisé l'opération inversée de passage (2D-3D). Dont le but est d'essayer de récupérer les coordonnées réelles par la méthode de triangulation.

Le résultat de la phase de calibration, donne une seule matrice des paramètres intrinsèques (Puisqu'il s'agit d'un seul type de caméra):

$$k = \begin{pmatrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 1 & 1 \end{pmatrix} \quad (\text{III.6})$$

Et pour chaque caméra i , nous avons une matrice de rotation R_i et un vecteur de translation T_i [15][16][17][20].

Alors, la matrice de projection pour chaque caméra i est la suivante:

$$M_i = k * [R_i \ T_i] \in \mathcal{R}^{3*4} \quad (\text{III.7})$$

A partir des i images 2D capturées, nous avons les coordonnées en pixel de chaque projection du point j :

$$m_{ij} = [u_{ij} \ v_{ij} \ 1]^T \quad i = 1 \dots m \text{ et } j = 1 \dots n \quad (\text{III.8})$$

Pour la reconstruction des points en 3D, en suit l'algorithme suivant:

$$N_i = M_i^{3*3} * -M_i^4 \quad (\text{III.9})$$

$$L_{ij} = M'_i * m_{ij} \quad (\text{III.10})$$

$$a_{ij}(k) = \frac{L_{ij}(k)}{L_{ij}(4)} - N_i(k) \ , \quad k = 1 \dots 3 \quad (\text{III.11})$$

Pour $i = 2$ images, nous calculons:

$$A_{ij} = \begin{bmatrix} a_{1j}(1) + a_{1j}(2) + a_{1j}(3) & -(a_{1j}(1) * a_{2j}(1) + a_{1j}(2) * a_{2j}(2) + a_{1j}(3) * a_{2j}(3)) \\ -(a_{1j}(1) * a_{2j}(1) + a_{1j}(2) * a_{2j}(2) + a_{1j}(3) * a_{2j}(3)) & a_{2j}(1) + a_{2j}(2) + a_{2j}(3) \end{bmatrix} \quad (\text{III.12})$$

$$V_{ij} = \begin{bmatrix} (N_2(1) - N_1(1)) * a_{1j}(1) + (N_2(2) - N_1(2)) * a_{1j}(2) + (N_2(3) - N_1(3)) * a_{1j}(3) \\ (N_1(1) - N_2(1)) * a_{2j}(1) + (N_1(2) - N_2(2)) * a_{2j}(2) + (N_1(3) - N_2(3)) * a_{2j}(3) \end{bmatrix} \quad (\text{III.13})$$

Le point reconstitué est donné par:

$$P_j = [N_1(1) + a_{1j}(1) * r_{ij}(1) \quad N_1(2) + a_{1j}(2) * r_{ij}(1) \quad N_1(3) + a_{1j}(3) * r_{ij}(1)] \quad (\text{III.14})$$

Nous résumons dans le tableau III.4, les valeurs numériques des coordonnées 3D à partir des coordonnées 2D obtenues après récupération de l'erreur minimale.

Positions	P1	P2	P3	P4	P5	P6	P7	P8
X'	-30	-15	8	59	39	-23	-1	3
Y'	163	37	113	140	155	5	71	80
Z'	54	25	43	52	36	13	27	22

Tableau III.4: coordonnées de la reconstruction

A partir de ces résultats, nous avons tracé la forme de l'objet (Figure III.5) de la même façon qu'au début de notre travail, pour visualiser la forme reconstruite de notre objet (cube).

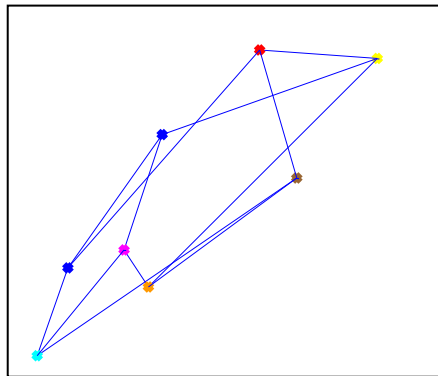


Figure III.15 : Reconstruction de l'objet

Malheureusement, nous avons obtenu un objet déformé. La cause de cette déformation est probablement due aux équations de transformation 3D-2D, qui sont utilisées pour une projection perspective et aussi avec des images réelles (photos), tandis que nos images sont prises pour une projection orthographique, ainsi que les caméras utilisées sont des caméras virtuelles de l'environnement Matlab, dont on ne connaît pas les paramètres intrinsèques.

III.6. Conclusion

Dans ce chapitre nous avons appliqué la PSO à un système MoCap afin de d'extraire la matrice de projection et ensuite la reconstruction 3D d'une forme d'un cube et par la méthode de triangulation. D'après les résultats obtenus on peut constater que la PSO est arrivée à optimiser les paramètres intrinsèques des caméras avec une minimale valeur de fonction objectif présentée en pixel. L'optimisation par essaim particulaire utilise la diversification pour une recherche efficace et elle s'applique facilement aux problèmes de plusieurs dimensions avec une convergence très rapide.