Odométrie visuelle à l'environnement intérieur par une caméra binoculaire

Groupe 14 :

Basile GAULIER M2 ISI

Gregoire KUBLER M2 ISI

Imene TARAKLI M2 ISI

Innocent MULAMBA TSHIKOMBA M2 CIMES

Menngmeng SHI M2 ISI

Nicolas CLEMOT M2 ISI

Salim DAHMANI M2 CIMES

Yuancheng ZHANG M2 ISI

Sorbonne Université

Table des matières

I) Présentation du projet d'architecture système :

I.01) Présentation du contexte et des objectifs du projet :

Le SLAM est l'acronyme de Simultaneus Localization And Mapping, cela désigne généralement un robot ou un corps rigide e mouvement, équipé d'un capteur spécifique, estime son propre mouvement et construit un modèle de son environnement, sans informations a priori. Dans le cas où le capteur utilisé est une caméra alors cela est appelé Visual SLAM.

Ce système peut être appliqué dans de nombreux domaines, comme en réalité Virtuelle ou la Réalité Augmentée ou les utilisateurs ne peuvent pas parcourir l'espace sans les informations fournies par le SLAM.

Dans ce projet, nous n'allons pas construite un système Visual SLAM complet, de par sa complexité et les contraintes de temps qui nous sont imposées, nous allons nous concentrer sur l'aspect de la localisation.

Ce que nous allons faire dans ce projet est d'utiliser une caméra binoculaire que nous tiendrons en main puis montée sur un robot mobile afin de collecter les données dans un environnement intérieur (LABORATOIRE DE JUSSIEU), le but étant de collecter les datasets d'images séquentielles pour localiser et récupérer la trajectoire de la camera.

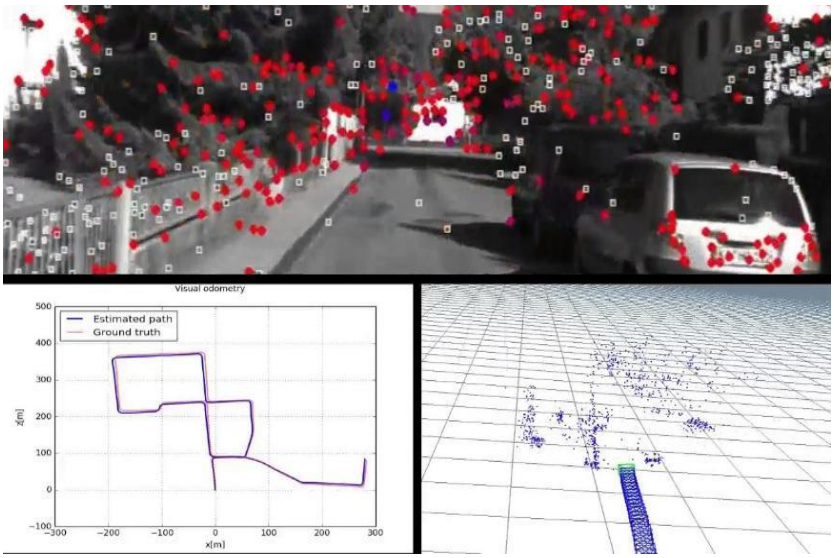


Figure 01 : Démonstration d'un algorithme d'une odométrie par Stéréo Vision par (Aladem et al en 2017)

I.02) Présentation d'éléments de planning et d'organisation du projet :

1	**Préparation du projet**				
1.1	Listage des tâches du projet	Nicolas/Grégoire	07/11/20	10/11/20	3
1.2	Création du calendrier	Nicolas/Grégoire	09/11/20	22/11/20	13
1.3	Documentation sur le Bundle Adjustment	Tous	07/11/20	16/11/20	9
1.4	Documentation sur l'ingénierie Système	Tous	07/11/20	16/11/20	9
1.5	Lister les tests à effectuer sur weedy	Basile / Salim	07/11/20	13/11/20	6
1.6	Effectuer les Tests sur weedy	Basile	13/11/20	17/11/20	4
1.7	Liste des Composants à acheter	Imene/Nicolas	09/11/20	22/11/20	13
1.8	Achat des composants	Imene/Nicolas	17/11/20	22/11/20	5
2	**Cahier des charges et montage**				
2.1	Architecture matérielle	Salim / basile	10/11/20	24/11/20	14
2.2	Architecture logicielle	Imene / Mengmeng	22/11/20	02/12/20	10
2.3	Montage + connexion + acquisition image de la caméra	Basile	22/11/20	07/12/20	15
2.4	Rapport d'ingénierie système	Salim / Innocent	22/11/20	18/12/20	26
3	**Connexion & calibration des caméras**				
3.2	Création d'une base de données stéréoscopique avec trajectoires connues + choix des keyframes	Basile / Innocent	25/11/20	10/12/20	15
4	**Bundle adjustement**				
4.1	Implémentation Oriented FAST + corner matching	Nicolas / Innocent	03/12/20	01/01/21	28
4.2	Triangulation des points (estimation des coord en 3D)	Mengmeng	25/11/20	10/12/20	15
4.3	Application de Ransac afin de garder seulement les meilleurs points	Grégoire / Yuancheng	03/12/20	01/01/21	28
4.4	Estimation de sa matrice de projection (perspective-n-point algorithme	Mengmeng / Imene	03/12/20	01/01/21	28
4.5	Moindres Carrées + Calcul Erreur + Fonction de Cout + Matrice Jacobienne	Yuancheng/Mengmeng/Basile / Nicolas	10/12/20	13/01/21	33
4.8	Faire le main dans lequel lle programme va loop + Cmake	Grégoire / Imène			
5	**Reconstruction du trajet**				
5.1	Création de frame qui contient les points caractéristiques et les matrices de transformations correspondantes	Mengmeng / Nicolas	27/12/20	19/01/21	22
5.2	Reconstruction du trajet à partir de la suite de matrices de transformation Successives	Yuancheng / Imene	12/01/21	22/01/21	10
5.3	Affichage du trajet (en temps réel)	Innocent / Salim	18/01/21	26/01/21	8

6	**Peaufinage et Rendu**				
6.1	Optimisation de code pour faire tourner en temps réel	Basile / Grégoire	13/12/20	30/01/21	47
6.2	Protocole de test (création et application)	Basile / Salim	03/01/21	25/01/21	22
6.3	Mise en forme propre des résultats (graphiques + images pr rapport)	Imene/ Mengmeng	05/01/21	26/01/21	21
6.4	Vidéo de présentation	Tous	01/11/20	05/02/21	94
6.5	Rapports de Gestion	Grégoire	01/11/20	05/02/21	94
6.6	Poster	Tous			

Figure I.2) Etapes et personnes responsables de chaque tâche

I.03) Glossaire :

Odométrie : Technique qui permet d'estimer la position d'un dispositif dans l'espace

Frame : Image générée par un système électronique

Caméra Binoculaire : Caméra disposant de 2 objectifs et revoyant les frames 2 par 2

Data sets : Ensemble de données organisées de sorte à être facilement manipulable

Flux de données : transfert de données d'un système à un autre. Peut se faire via voie filaire ou par onde

FAST : Algorithme de détection permettant d'extraire des features (détails) d'une image

Corner matching : Algorithme de mise en correspondance des features, généralement utilisé parallèlement à un algorithme tel que FAST. Permet de "lier" les 2 images retournées par la caméra

Triangulation : Estimation de la position d'un point dans le repère monde à partir de sa position dans les images de chaque caméra

Matrice de transformation : Matrice qui, appliquée à un point permet de le déplacer dans l'espace d'un angle et d'une position voulue.

Bundle adjustement : Permet d'optimiser les matrices de transformation en minimisant l'erreur de reprojection entre les emplacements des points d'image observés et prédits.

ICP (Iterative Closest Point): Une technique permettant de minimiser la différence entre deux nuages de points 3D. Dans notre cas ICP est utilisé pour estimer les matrices de transformation.

II) Architecture Opérationnelle :

II.1) Analyse de l'environnement :

Le système d'odométrie visuelle que nous allons mettre en place est destiné à fonctionner dans un laboratoire du campus de Jussieu ou l'une de ses salles, ou la variation de lumière n'est pas trop brusque afin d'éviter tout dérèglement du calibrage des caméras, ainsi que la possibilité de l'exploiter dans un environnement ou les trajectoires peuvent varier et peuvent être aléatoires.

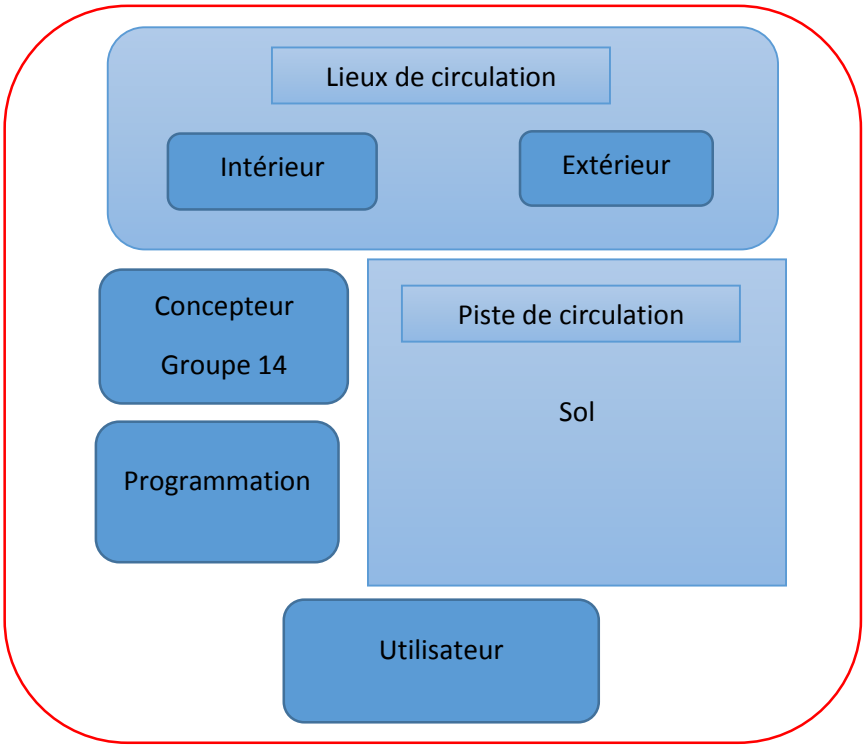


Figure II.1 Environnement du système.

II.2) Interfaces opérationnelles :

Notre système baignera dans un environnement que l'utilisateur voudra explorer, en se déplaçant, les caméras binoculaires récolteront un flux d'images qui devront être injectées dans un algorithme via un ordinateur, notre algorithme codé en C++ traite toutes les séquences et nous indique les trajectoires prises par les caméras binoculaires.

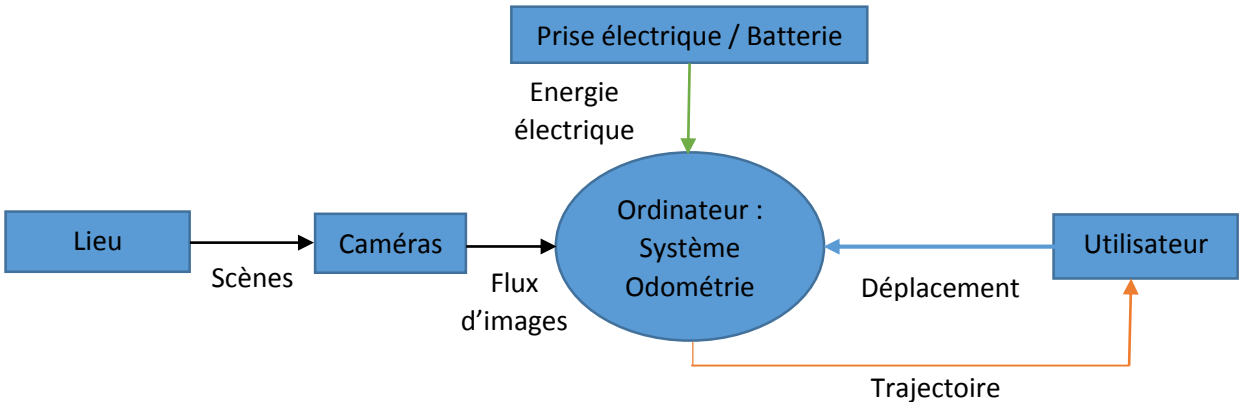


Figure II.2.1 : Interactions du système avec son environnement

Nom du flux	Type de flux fonctionnel	Type de lien physique
Energie électrique	Flux électrique	Câble d'alimentation
Flux d'images	Flux d'informations	Câble USB
Déplacement	Flux de données	Main de l'utilisateur/Robot
Trajectoire	Flux de données	Ecran de l'ordinateur

Figure II.2.2 Tableau contenant les noms des flux, leurs types ainsi que le type de lien physique.

II.3) Analyse des besoins :

Nos besoins concernant l'environnement est la structure permettant d'exploiter notre système sans variations brusque de lumière et avec un minimum de perturbation externes tels que des chocs ou des corps imprévus.

Par contre l'environnement nous expose aussi à des contraintes tel que l'impossibilité de modifier la structure ainsi que la variation de température qui dans certains cas pourrait perturber les composants et fausser les données récoltés.

Environnement	Service ou contraintes	Capacités	Critères	Contexte
Intérieur	Service	Fournir des scènes	Espace contrôlé mesurable	Pièce dont la lumière est facilement gérable
Extérieur	Service	Multitude d'obstacles	Espace large	Pouvoir constater les objets lointains et leurs profondeurs
Extérieur	Contrainte	Impossible de gérer les variations de lumière	Aléas météo	Changements climatiques
Extérieur	Contrainte	Corps étrangers mobiles	Animaux et personnes	Espace extérieur animé
Extérieur	Contrainte	Déplacement instable	Sol irrégulier	Secousses

Figure II.3.1 Tableau contenant les environnements ainsi que tous les services capacités critères et contextes

II.4) Analyse et consolidation des contextes opérationnels

Nous allons dans cette section voir le contexte opérationnel du système ainsi que son cycle de vie.

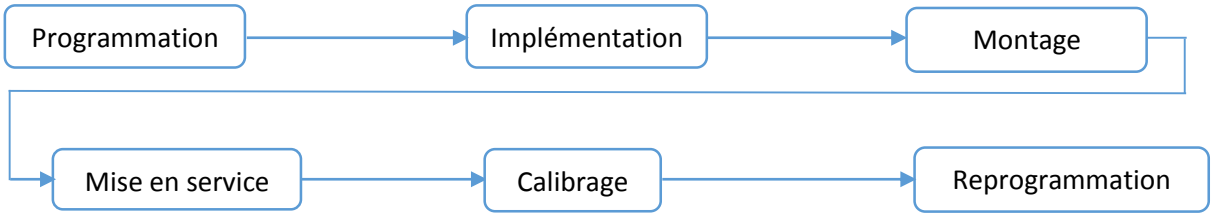
II.4.1) Contextes opérationnels :

Ici, nous allons mettre en évidence le rôle des acteurs qui interagiront avec notre système, et leur rôle :

- Utilisateur/Robot : déplacer le système d'odométrie.
- Caméras : capturer les flux des images et les transmettre via un câble USB.
- Ordinateur : Reçoit le flux d'images émis par les caméras, et transmet la trajectoire à l'utilisateur (ou SLAM)
- Programme : Traite tout le flux, et génère des points qui correspondent à la trajectoire de déplacement.

II.4.2) Cycle de vie :

Ici on voir les différentes étapes du produit final, car notre rôle ne se porte que jusqu'au prototypage, il n'aura donc pas de cycle de vie explicite.



II.5) Analyse des cas d'utilisations

L'analyse des cas d'utilisation nous permet de visualiser les différentes interactions et apport de chaque section du projet par rapport aux autres.

Car chaque bloc a son propre rôle qu'il devra fournir aux autres blocs ou en recevoir pour être fonctionnel.

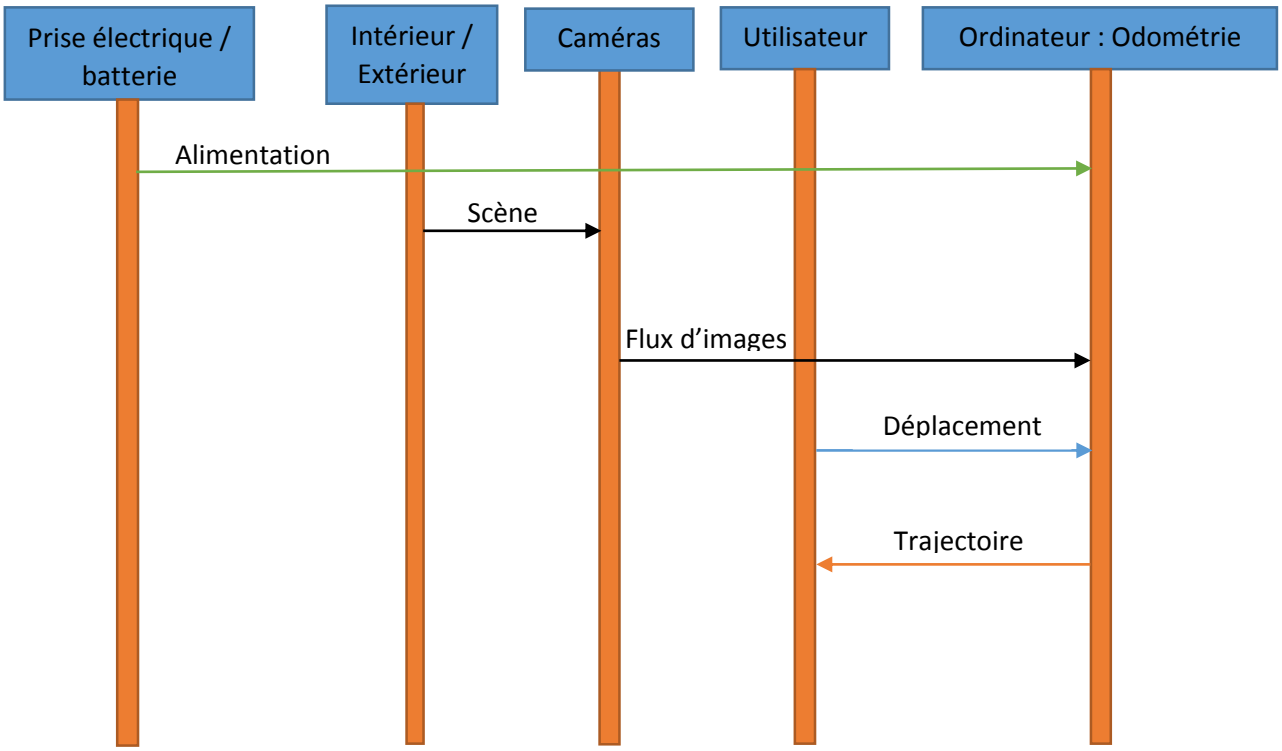


Figure II.5) Analyse des cas d'utilisation de notre système

III) Architecture fonctionnelle

Dans cette section du rapport, nous allons découper notre système en plusieurs blocs afin de remédier aux contraintes de production. Nous allons donc mettre en évidence le rôle de chaque bloc, commençons pas l'analyse des exigences fonctionnelles.

III.1) Analyse des exigences fonctionnelles

Les exigences principales sont d'afficher une trajectoire à partir des images récoltées par deux caméras. Il faut que les calculs effectués soient optimisés, et que les caméras soient calibrées.

Fonction	Critères de performance	Contexte
Détection de points clés	Caméras opérationnelles	Choix pts stables
Création d'une correspondance	Détection pts semblables entre les deux caméras	La correspondance se fait avec les points choisis à l'étape de détection
Triangulation	Les points 3D Doivent être construits à partir d'images 2D des canaux	Utilisation d'une caméra binoculaire pour la stéréo vision
Sélection de points	Points 3D du repère caméra dont la mise en correspondance se fait avec une erreur minimum prédéfinie	La sélection de points se fait avec RANSAC, pour enlever les outliers et garder les meilleurs points seulement.
Calcul de la translation et de la rotation	Le résultat doit matcher le déplacement réel +/- une erreur décidée durant la phase de programmation	Le calcul est fait en comparant la position 3D de points connus entre 2 paires d'images stéréo à (t,t+1)
Optimisation du résultat	Fonctionne en temps réel	Le calcul de la rotation et translation est une approximation qu'il faut minimiser
Génération de la trajectoire	La trajectoire à temps T doit être cohérente avec les positions précédentes	La trajectoire est générée à partir des matrices de rotation et translation successives générées par le bundle adjustment

Figure III.1 : Tableau contenant les fonctions ainsi que les critères de performances et leur s contexte

Nous allons dans le prochain titre mettre en évidence les différentes fonctions et sous fonctions du système et les représenter sous forme de schéma blocks.

III.2) Analyse et architecture fonctionnelle

Sur le schéma bloc suivant, nous décomposons notre fonction en sous fonction.

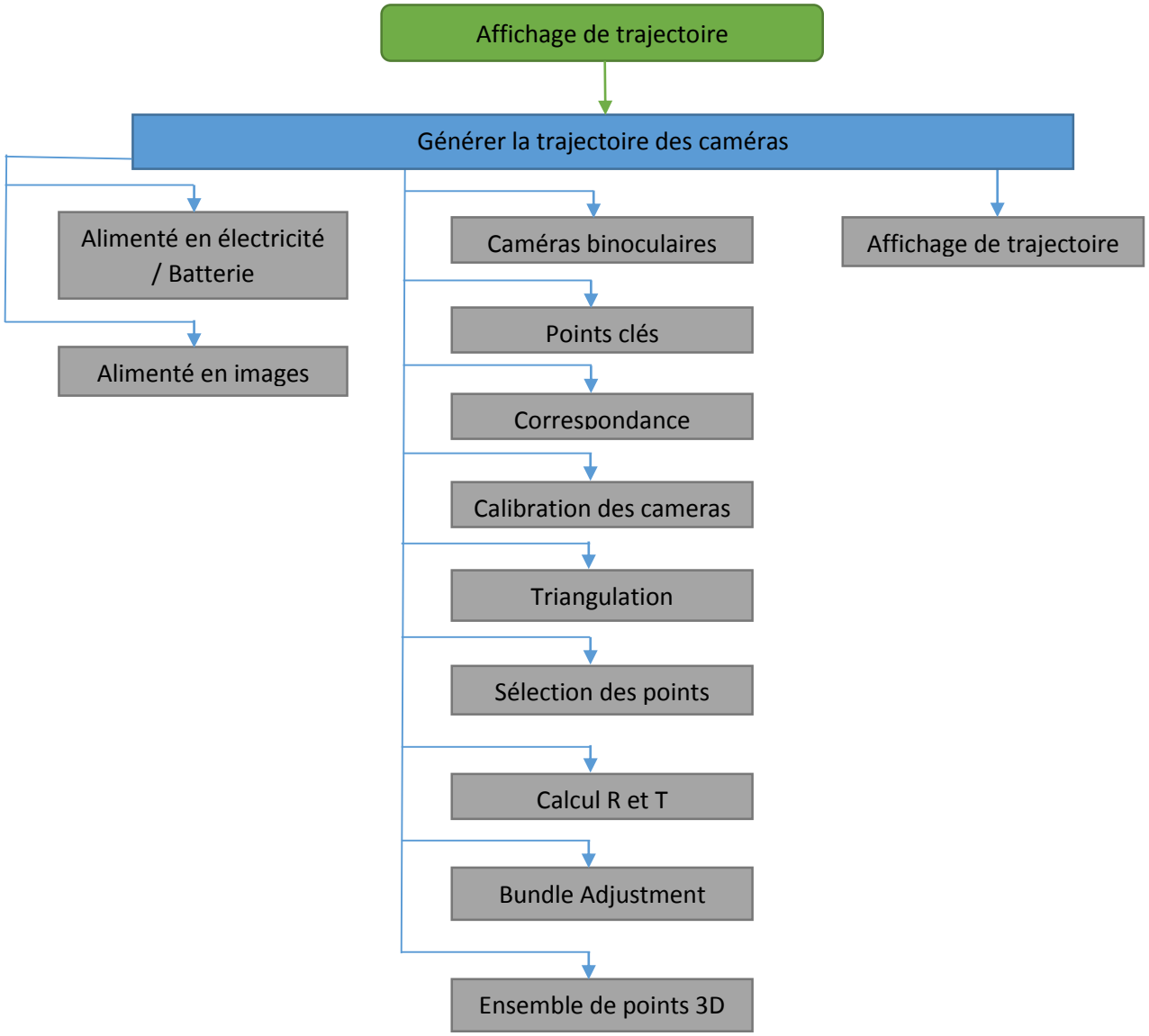


Figure III.2 : Architecture fonctionnelle.

III.3) Architecture fonctionnelle statique

L'architecture fonctionnelle statique nous permet de consolider les différents flux, internes et externes en plus d'identifier les flux fonctionnels du système.

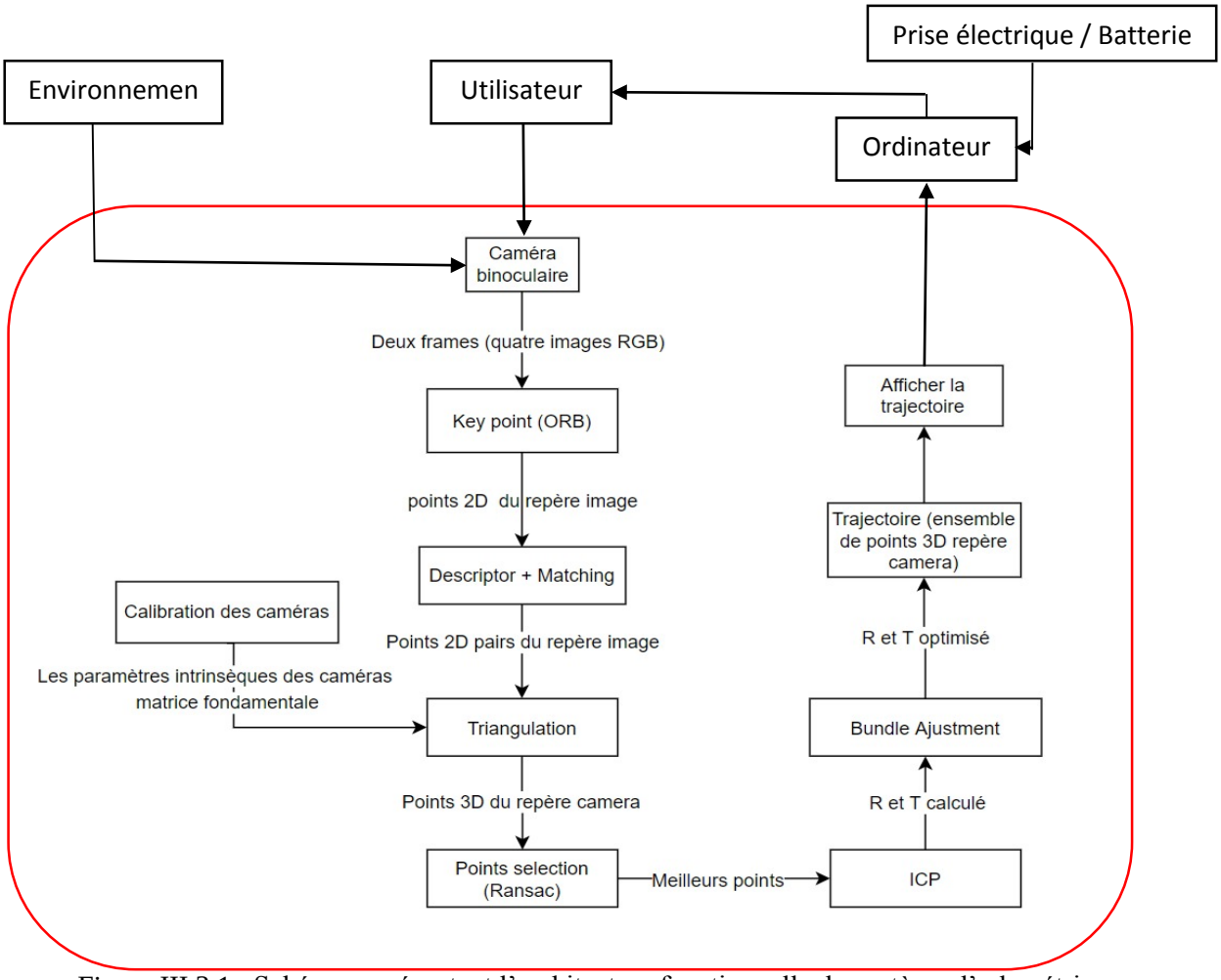


Figure III.3.1 : Schéma représentant l'architecture fonctionnelle du système d'odométrie

Nous verrons maintenant le tableau d'interface fonctionnelle :

	A	**B**	**C**	**D**	**E**	**F**
A	//	//	//	//	//	//
B	//	//	//	//	//	//
C	Matrice intrinsèques des caméras	4 nuages de points 2D (2 canaux et 2 instants)	//	//	//	//
D	//	//	2 Nuages de points 3D	//	//	//
E	//	//	//	2 Nuages de points 3D	//	//
F	//	//	//	//	Rotation et Translation initialisées	//
G	//	//	//	//	//	Rotation et translation optimisées

Figure III.3.2 : Tableau d'interfaces fonctionnelles de notre système

A : Calibration des caméras

B : Points caractéristiques, correspondances

C : Triangulation

D : Sélection des points

E : Calculs des R et T

F : Bundle Adjustment

G : Affichage trajectoire

III.4) Architecture fonctionnelle dynamique

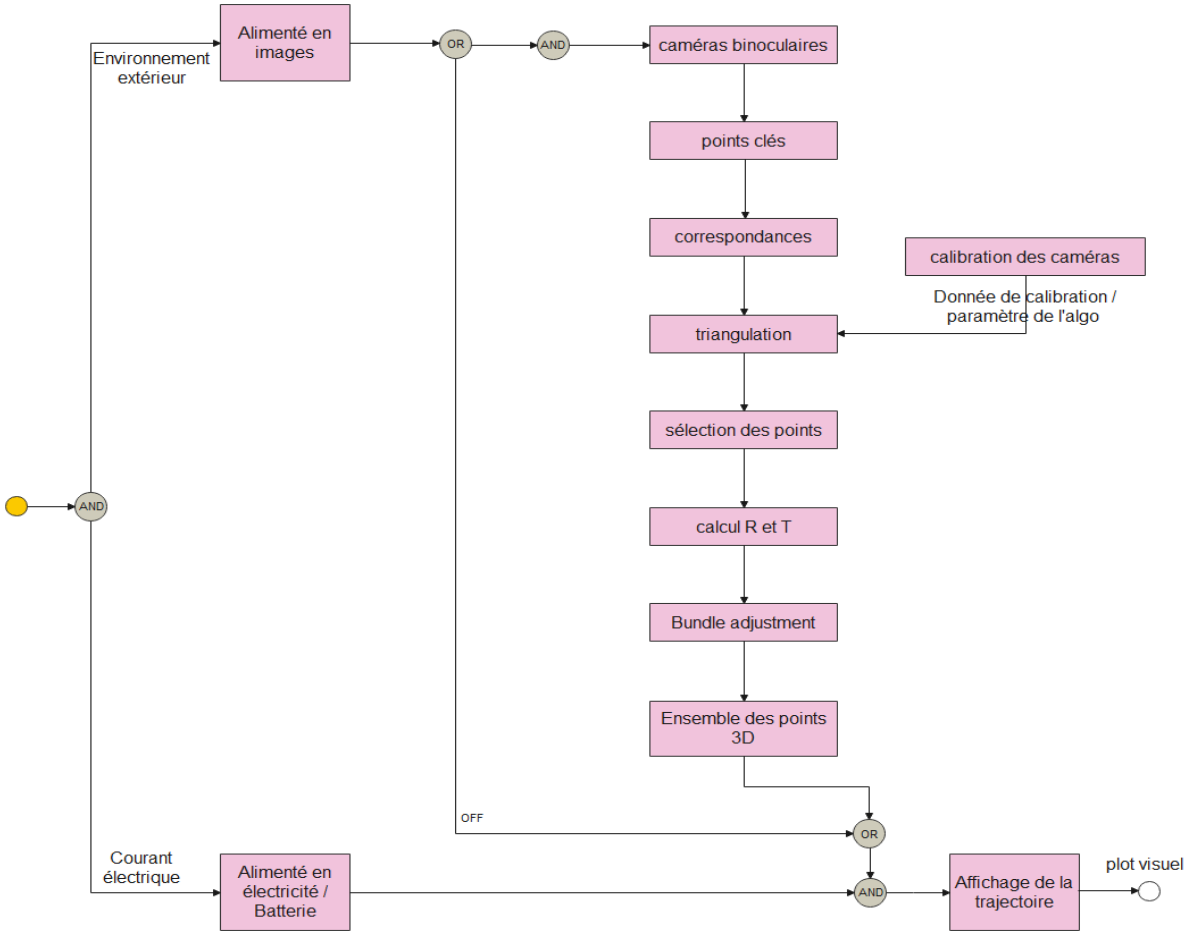


Figure 3.3.3 : Architecture Dynamique du système

III.5) Identification des modes de fonctionnement

Nous compléterons l'architecture fonctionnelle par les modes de fonctionnement du système d'odométrie visuelle.

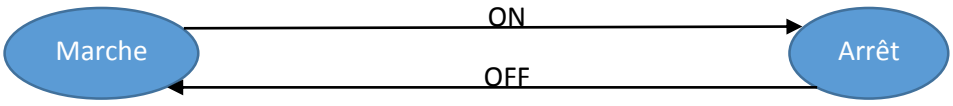


Figure III.5.1 : Identification des modes de fonctionnement

Modes	Fonction disponibles	Contextes couverts
En marche	Toutes des fonctions sont dispo	Alimentation et éclairage
En arrêt	Aucune fonction n'est disponible	teint ou dans le noir
En veille	Sauvegarde de la trajectoire	//

III.6) Interfaces fonctionnelles

Nous allons consolider les interfaces fonctionnelles dans un tableau décrivant les types de données les types de flux et enfin

Nom du flux	Description des données	Flux externes/internes	Recherché / non recherché
2 frames	4 vecteurs de taille (n*n*3)	Interne	Recherché
Points 2D correspondants	2 vecteurs de taille (2*k)	interne	Recherché
Points 3D correspondants	2 vecteurs de taille (3*k)	Interne	Recherché
Translation	1 matrice de taille (3*3)	Interne	Recherché
Rotation	1 matrice de taille (3*3)	Interne	Recherché
Trajectoire	Ensemble de points 3D	Interne	Recherché

IV) Architecture organique

Nous allons dans ce chapitre reboucler les activités avec l'architecture opérationnelle et l'architecture fonctionnelle

IV.1) Analyse des exigences organiques

Après avoir exploré les architectures opérationnelles et fonctionnelles du système, décrivons les caractéristiques physiques.

- Tout d'abord le corps d'épreuve de notre système doit être mobile et malléable.
- La distance entre les caméras doit être fixe.
- Exposer au minimum aux chocs et aux variations brusques de lumière.
- Les caméras soient tout le temps calibrées.

Caractérise physique	Critère de performances	Contexte
Mobilité	Implantation d'un moteur	Tous les contextes
Distance fixe entre les cameras	Distance adapté aux besoins des calculs des profondeurs des points	Utilisation de deux caméras distinctes
Minimisation des chocs	Maintenir les calibrations des caméras	Utilisation du système en extérieur
Calibration des caméras	Trouver la bonne relation entre les coordonnées du monde avec l'image	Déterminer les paramètres intrinsèques

Figure IV.1.1) Tableau stipulant les exigences organiques du système

IV.2) Analyse et architecture organique

Au cœur de cette partie nous identifierons les parties du système et des interactions des composants entre eux et avec l'environnement.

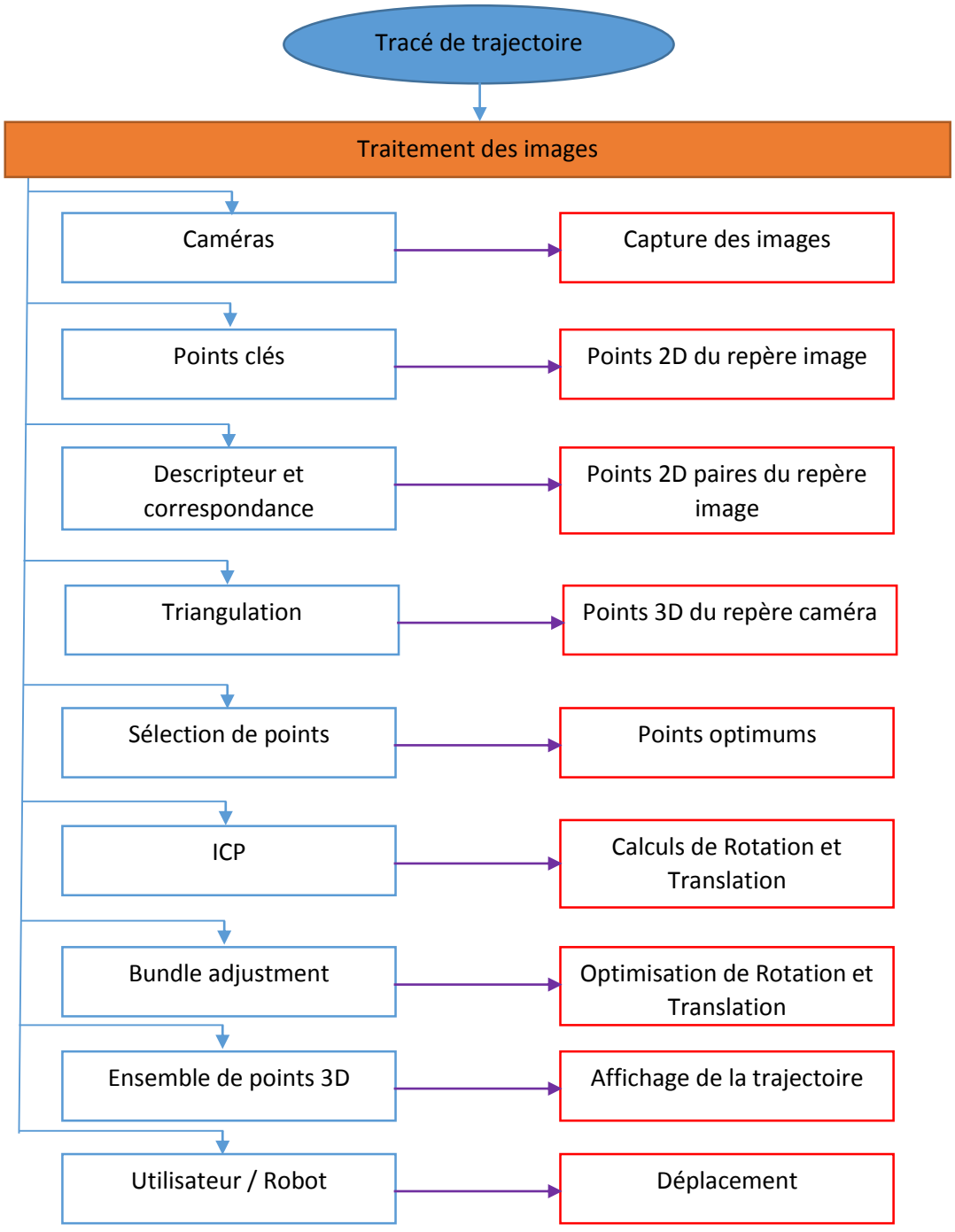


Figure IV.2) Architecture organique du système

IV.3) Architecture physique statique

Identifions maintenant les interactions internes et externes des composants du système et de l'environnement.

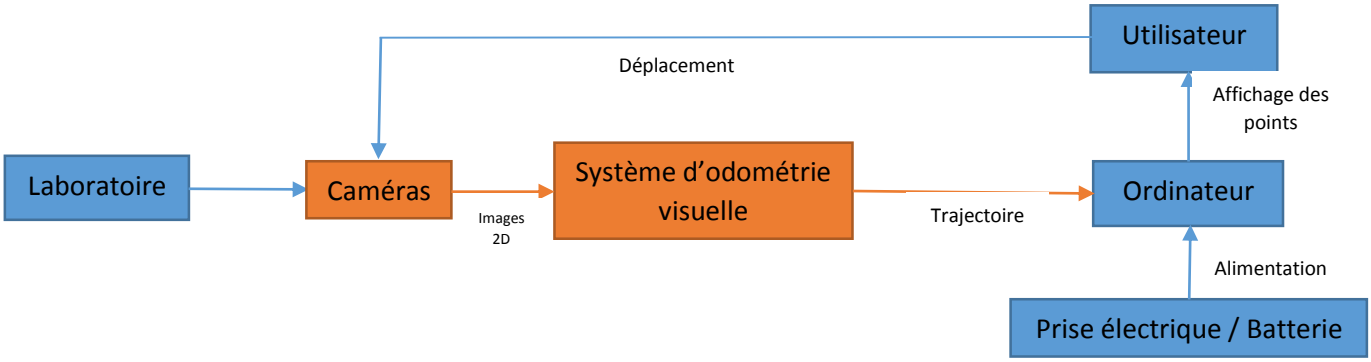


Figure IV.3 : schéma représentant l'architecture physique statique du dispositif

IV.4) Architecture organique dynamique

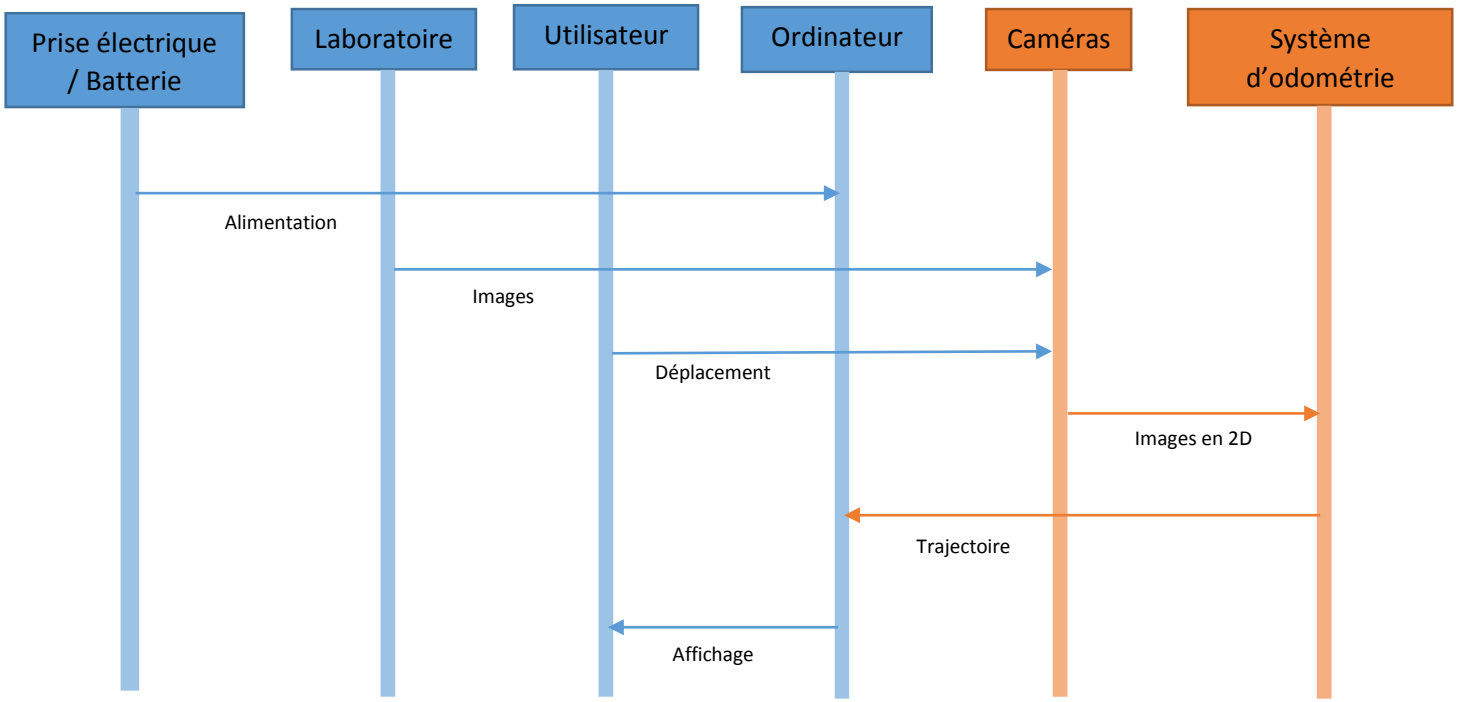


Figure IV.4 Figure de l'architecture organique dynamique du système

IV.5) Interfaces organiques

Ici nous exposons les liens entre qui existent entre les composants, ainsi que les allocations des flux dans les interfaces physiques.

	Alimentation	**Laboratoire**	**Utilisateur**	**Ordinateur**	**Caméras**	**Système d'odométrie**
Alimentation	/	Prise électrique		Câble d'alimentation		
Laboratoire	Prise électrique	/	Espace de travail		Flux d'images	
Utilisateur		Espace de travail	/		Manuelle	
Ordinateur			Affichage à l'écran	/		
Caméras				Câble USB	/	Images en 2D
Système d'odométrie				Trajectoire des caméras		/

Figure IV.5 Tableau d'interfaces entre les composants

Nom de l'interface physique	**Type de lien physique**	**Flux fonctionnelles passant par le lien**	**Interface**
Prise électrique	Mural	Tension électrique	Externe
Espace de travail			
Câble USB	Données	Signal électrique	Interne
Trajectoire des caméras	Affichage de trajectoire		Interne
Images 2D	Données	Signal électrique	Interne
Affichage à l'écran	Visuel		Externe

Figure IV.5.2 Tableau contenant les types de liens ainsi que les flux et les interfaces du système

IV.6) Identification des configurations organiques

Au sein de ce chapitre voyons la principale configuration organique dont sera exposé notre système, cette dernière consiste à définir et tracer la trajectoire des caméras binoculaires en mettant en entrée un flux d'images.

La variation de situation pourrait se manifester par la source de ce flux, qui pourrait venir des caméras en elles-mêmes, ou alors d'un flux vidéo préalablement enregistré. Le processus demeure cependant identique. De plus, selon le domaine d'application ou les conditions d'utilisations, un compromis entre la précision et la vitesse d'exécution est à étudier, car plus la précision est accrue plus le temps d'exécution est important.

IV.7) Dimensionnement du système

Afin d'aboutir à un système optimal, une étude de matériel et de méthode a été effectué :

Caméras : Le choix s'est porté sur l'utilisation de caméras binoculaires car un compromis de coût et de performance a été nécessaire, car une caméra stéréo n'est pas efficace malgré son coût réduit, et un LIDAR, est bien trop coûteux mais dispose de performances bien plus intéressantes. La distance séparant les deux objectifs n'est en rien anodine car pour effectuer de la triangulation à une distance minimum de 10 mètres, un écart de 30 centimètres entre les deux caméras est nécessaire.

La nécessité de munir notre dispositif d'un bloc de triangulation car nos deux caméras n'ont pas de dimensions de profondeur, et la netteté de l'image est primordiale pour le bon fonctionnement de l'algorithme.

RANSAC : Les images collectées pourraient être trop complexes et denses, alors, la l'optimisation de choix des point est primordial pour le bon fonctionnement de l'algorithme, c'est à ce niveau que le bloc RANSAC intervient et optimise l'utilisation de l'algorithme

ICP : Une fois les nuages de points récoltés par les deux caméras, une correspondance entre les deux nuages est primordiale, plusieurs méthodes existent pour effectuer cela, mais la méthode Iterative Closest Points, est la plus commune et dont la documentation est la plus disponible.

Bundle adjustment : L'utilisation de ce bloc est le point phare du projet, car c'est ce qui permet de générer les points de déplacement, c'est une méthode commune et l'approche la plus standard utilisé dans le SLAM, ce qui implique que sa documentation est trés disponible et varié, ce qui nous permets de nous consacrer à la concrétisation de notre projet.

V) Conclusion

Lors de ces dernières semaines, nous avons considérablement amélioré notre travail en équipe et nos organisations, à travers les différentes interactions, et l'obligation d'imaginer différents cas de figures et d'utilisation de notre système. En outre, la responsabilisation de chaque membre en lui consacrant des tâches bien particulières, ce qui nous a permis d'aboutir à un schéma qui correspond aux besoins du projet.

Pour répondre au problème principal, l'aboutissement d'une lettre de mission et d'un diagramme de distribution des tâches à été effectué par différents membres du groupe.

Ces dernières nous ont permis de soulever plusieurs problèmes tels que la chronologie des événements dans le diagramme des taches, ou des anomalies au niveau des premiers schémas ou des dimensionnements ont été résolus et décortiqués au fur et à mesure de nos réunions hebdomadaires, ainsi que la résolution de ces derniers par l'intervention d'un membre qualifié du groupe.

La caractérisation d'une étape primordiale dans la bonne marche du projet est l'étude des différentes méthodes et la fourniture de fiches concernant les principaux sujets et méthodes utilisés de la part de certains membres pour le reste du groupe.

Certains points et paramètres restent cependant en suspens, de par l'impossibilité de nous rencontrer au sein de l'université à cause de la crise sanitaire, et le manque de moyens matériels pour entamer la concrétisation du projet.