

Réunion semestrielle n°1 Projet DS3DLive

Maxime TEIL
maxime.teil@lgp2.grenoble-inp.fr

Laboratoire LGP2 - Grenoble INP

26 avril 2021



Partenaires :



Ordre du jour

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet

Rappel sur l'objectif du projet ...



Objectifs et acteurs du projet

Objectif principal

Mise en place et développement d'un dispositif capable de caractériser des défauts, sur une ligne de production et en continu, par l'application de techniques d'imagerie, à l'échelle microscopique, 2D et 3D en vu d'un pilotage automatique de la ligne de production.

Sont concernés :

- Industrie papetière produisant par enduction (Ahlstrom-Munksjö, Gerex)
- Industrie de l'informatique spécialisée en traitement des images (Reactiv'IP)
- Centres de recherche concernés par la caractérisation des matériaux, notamment fibreux, et par les procédés papetiers (CMTC, LGP2, Agefpi)

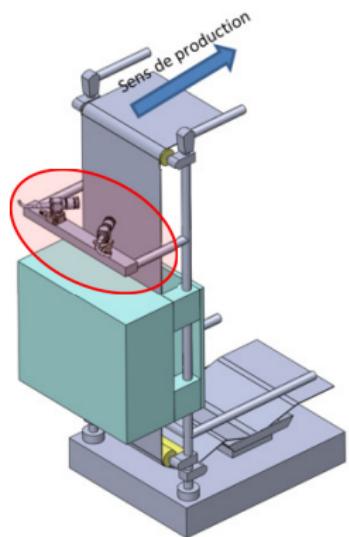
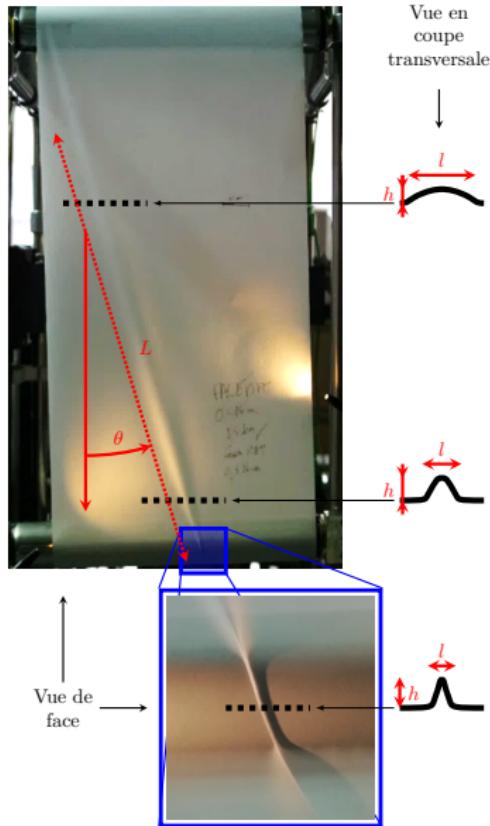


Illustration d'un dispositif imaginé pour la caractérisation sur la machine pilote (CAO produite par Maxime TERRIEN).

Caractérisation des défauts



Caractérisation des plis / goulettes

- Angle par rapport au sens de production
 $\theta \approx [0, 30]^\circ$;
- Longueur $L \approx [10, 100]\text{cm}$;
- Largeur $l < 1\text{ mm}$;
- Hauteur $h < 1\text{ mm}$);
- Répartition des défauts (nombre, orientation moyenne, fréquence d'apparition, etc.).

Images prises par Florian LE GALLIC (Thèse en cours).

Moyen de caractérisation des défauts

Techniques de stéréovision - méthode de dimensionnement bien connue

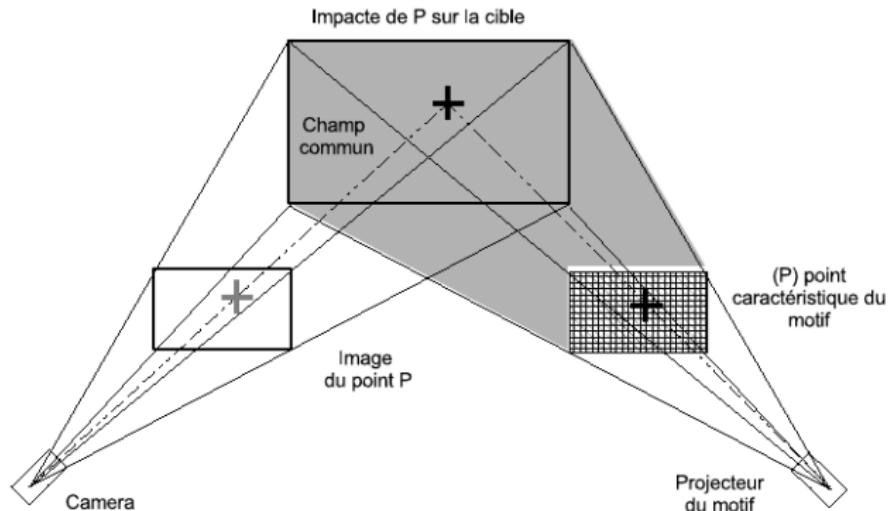
- Métrologie depuis les années 1990 : mesure des déplacements et/ou déformations d'un matériau [Faugeras (1993), Goulette (1999)].
- Métrologie en sollicitations dynamiques ces dernières années : augmentation de la puissance de calcul pour la corrélation temporelle et la stéréocorrélation [Abdullah (2019)].
- Possibilité d'obtenir des champs denses du profil altimétrique sous certaines conditions [Orteu (2002)] :
 - Résolution de la caméra ;
 - Géométrie du dispositif (angles et distances entre caméras / cible) ;
 - Technique de mise en stéréocorrespondance (grille $\approx 1/30$ pixel ou corrélation $\approx 1/100$ pixel).

Système de stéréovision

Au vu de l'encombrement des machines de production industrielles / pilote et de la contrainte de vitesse pour mener les calculs en continu, il est envisagé un système rigide à deux caméras avec vidéoprojecteur pour ajouter de la texture au matériau papier et permettre la recherche de stéréocorrespondants.

Principe de la stéréovision

[Graba (2005)]



- ① Calibration du système optique et définition de la géométrie épipolaire ;
- ② Acquisition des images et recherche des zones stéréocorrespondantes ;
- ③ Calcul de la géométrie 3D par triangulation.

▶ Sommaire

Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

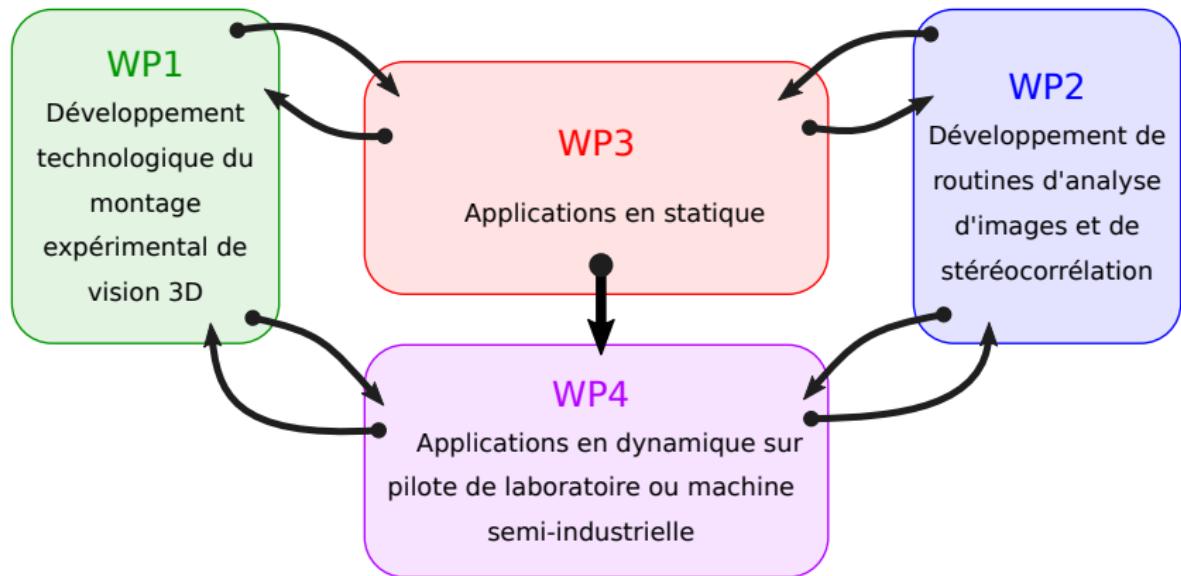
- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet

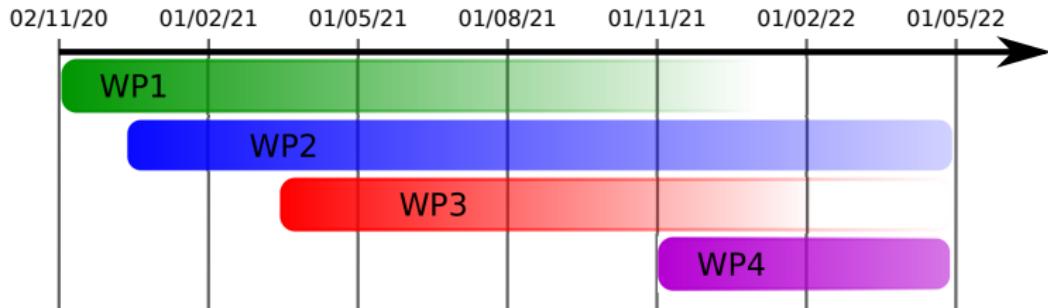
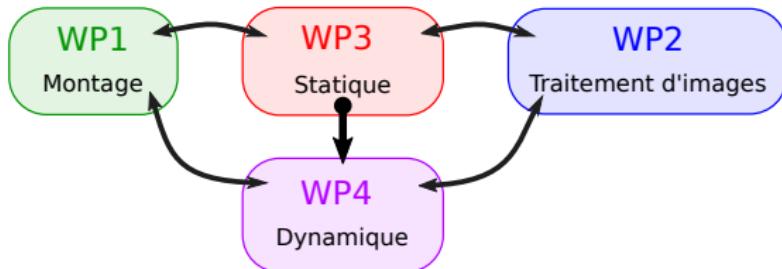
▶ Sommaire

Tâches principales

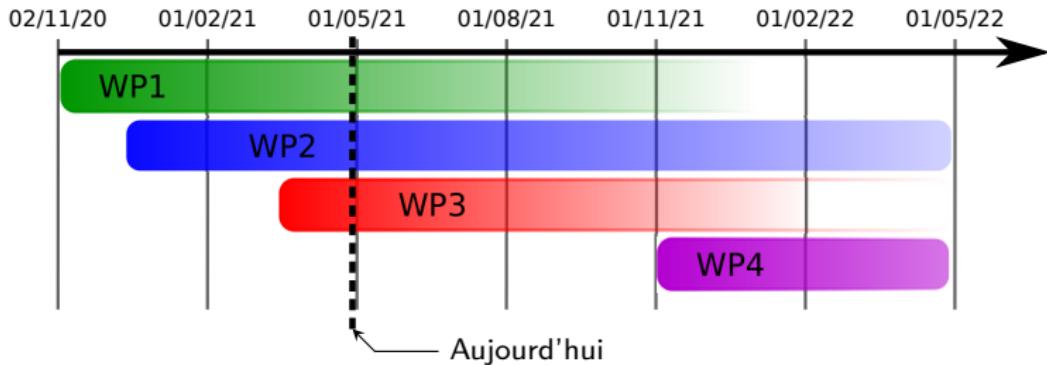
En ne considérant que les aspects techniques (hors gestion du projet et valorisation des résultats), 4 tâches caractérisent l'organisation du travail :



Planning prévisionnel



Planning réel



Planning respecté

Le planning est assez bien respecté. Une modification des tâches et cependant nécessaire : la calibration des systèmes optiques est intégrée dans WP2.

- ⇒ WP2 : Développement de routines de *calibration des systèmes optiques*, d'analyse d'images et de stéréocorrélation.

Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

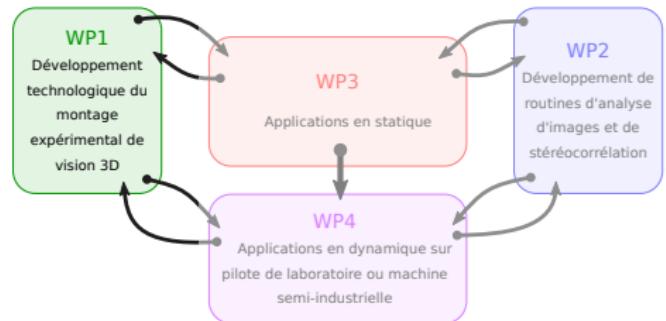
2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet



Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

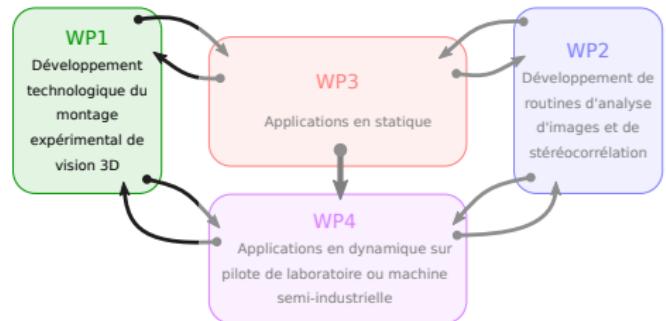
2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet



Constitution du dispositif



Constitution du dispositif

- Structure similaire à la machine pilote



Constitution du dispositif

- Structure similaire à la machine pilote
- Support des systèmes optiques



Constitution du dispositif

- Structure similaire à la machine pilote
- Support des systèmes optiques
- Systèmes optiques

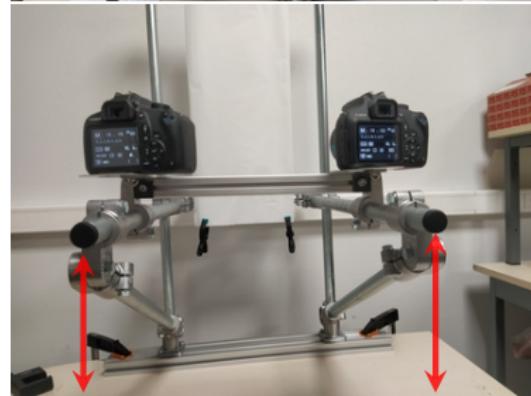
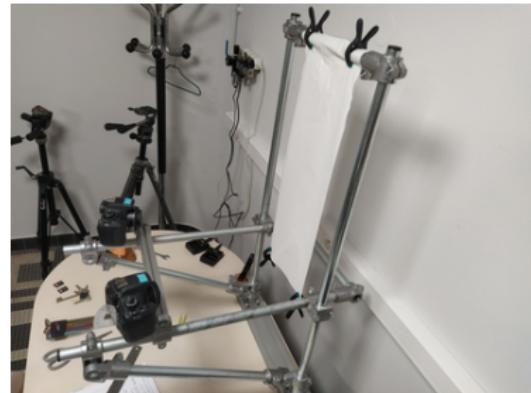


Réglages possibles



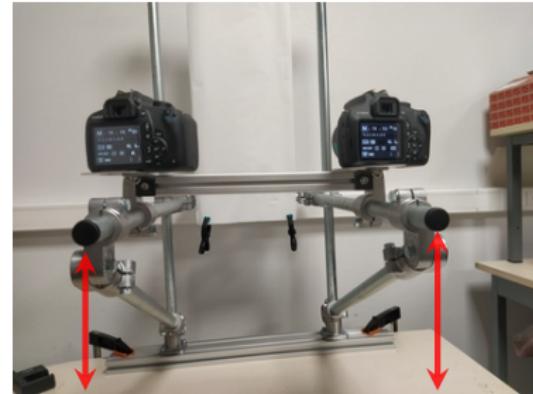
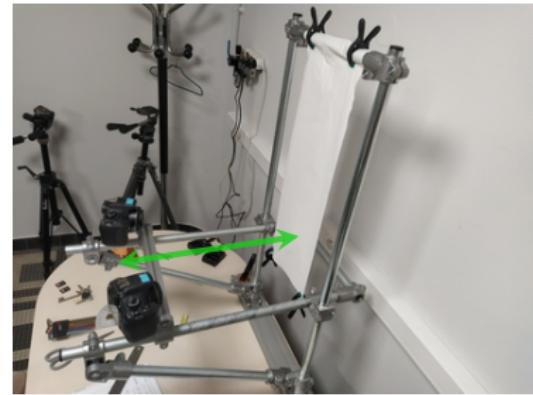
Réglages possibles

- Hauteur des systèmes optiques



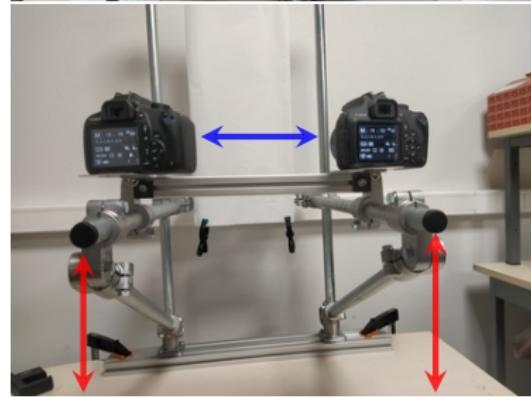
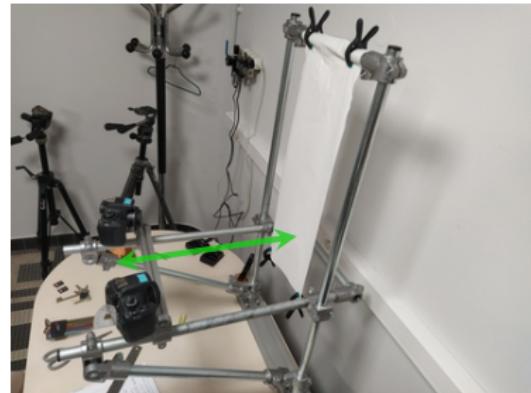
Réglages possibles

- Hauteur des systèmes optiques
- Distance papier / caméras



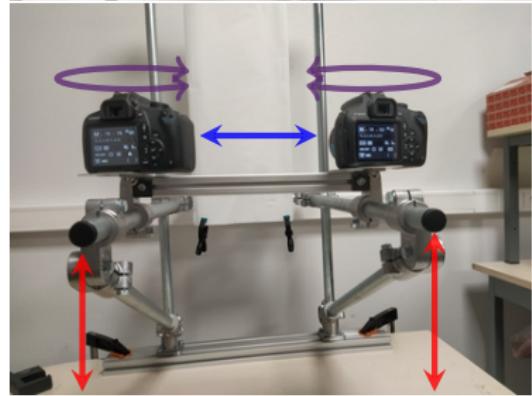
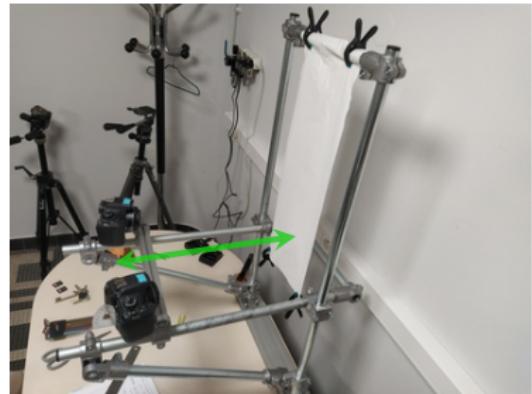
Réglages possibles

- Hauteur des systèmes optiques
- Distance papier / caméras
- Distance entre caméra



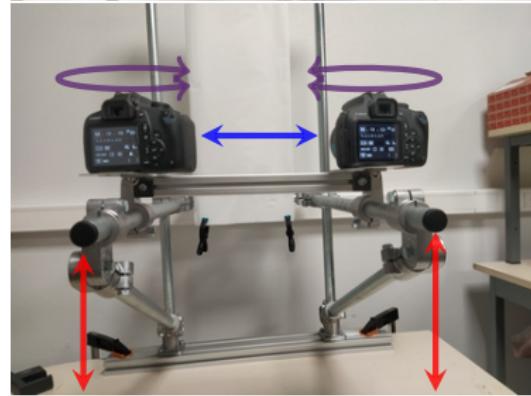
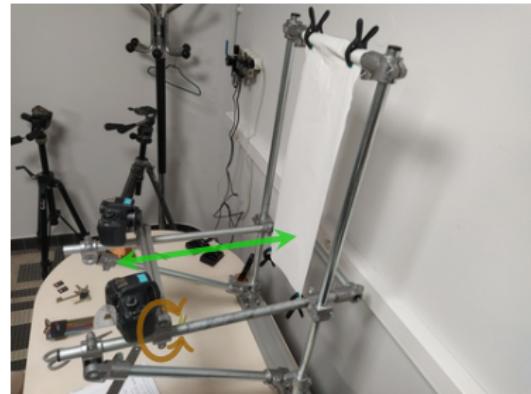
Réglages possibles

- Hauteur des systèmes optiques
- Distance papier / caméras
- Distance entre caméra
- Angle "de lacet" de chacune des caméras



Réglages possibles

- Hauteur des systèmes optiques
- Distance papier / caméras
- Distance entre caméra
- Angle "de lacet" de chacune des caméras
- Angle "de tangage" des deux caméras



Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

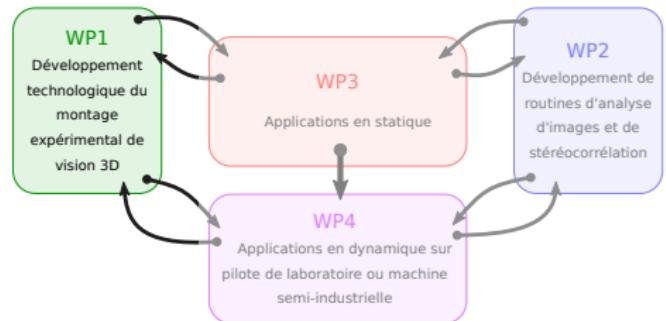
2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet



Appareil photographique numérique (APN)

Propriétés du capteur associé au Canon EOS 1200D

Format	APS-C (22.3 mm × 14.9 mm)
Pixels effectifs	18 MPx
Ratio de format	3 : 2
Filtre passe bas	intégré



Objectif à focale fixe adaptable à l'APN

Propriétés de l'objectif Canon EF-S 24 mm f/2.8 STM

Focale équivalente (36 mm × 24 mm)	38 mm
Distance minimale mise au point	160 mm
Autofocus motorisé	oui
Angle de champ horizontal	59°10'
Angle de champ vertical	50°35'



Utilisation des systèmes optiques

Résolution

Dans ce projet, les APN auront un champ de vision horizontal d'environ 35 cm. Les images obtenues présentent l'axe horizontal selon approximativement 5000 pixels.

$$\Rightarrow \text{Taille du pixel : } \approx 70 \mu\text{m}$$

Méthode d'utilisation

- 1 APN = 1 objectif : pas de changement grâce à l'étiquetage
- Calibration de l'ensemble (APN + objectif) si changement
- Calibration du système stéréoscopique après chaque changement de réglage du montage
- Prise de vue du sujet et enregistrement sur carte SD au format JPG (18 MPx)
- Post-traitement sur ordinateur. Mise en veille de l'expérimentation et protection des équipements

Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

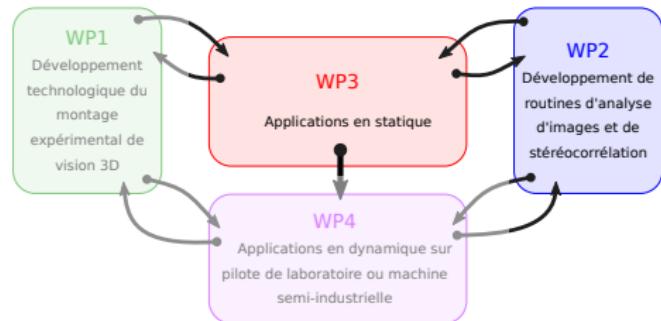
2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet



Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

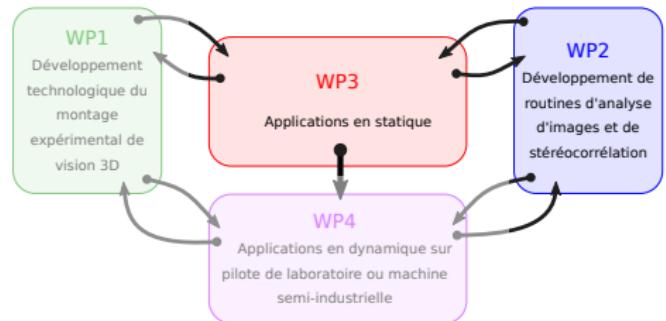
2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet



Quelques règles d'un travail bien fait en développement numérique

Afin d'assurer une bonne distribution, une utilisation facile pour l'utilisateur et une longue durée de vie à un programme, il est nécessaire de respecter les règles qui suivent :

- Fonctionnaliser les tâches ;

Quelques règles d'un travail bien fait en développement numérique

Afin d'assurer une bonne distribution, une utilisation facile pour l'utilisateur et une longue durée de vie à un programme, il est nécessaire de respecter les règles qui suivent :

- Fonctionnaliser les tâches ;
- Donner de la modularité aux algorithmes ;

Quelques règles d'un travail bien fait en développement numérique

Afin d'assurer une bonne distribution, une utilisation facile pour l'utilisateur et une longue durée de vie à un programme, il est nécessaire de respecter les règles qui suivent :

- Fonctionnaliser les tâches ;
- Donner de la modularité aux algorithmes ;
- User des avantages de la programmation orientée objet ;

Quelques règles d'un travail bien fait en développement numérique

Afin d'assurer une bonne distribution, une utilisation facile pour l'utilisateur et une longue durée de vie à un programme, il est nécessaire de respecter les règles qui suivent :

- Fonctionnaliser les tâches ;
- Donner de la modularité aux algorithmes ;
- User des avantages de la programmation orientée objet ;
- Rendre distribuables les librairies, dans la mesure du possible ;

Quelques règles d'un travail bien fait en développement numérique

Afin d'assurer une bonne distribution, une utilisation facile pour l'utilisateur et une longue durée de vie à un programme, il est nécessaire de respecter les règles qui suivent :

- Fonctionnaliser les tâches ;
- Donner de la modularité aux algorithmes ;
- User des avantages de la programmation orientée objet ;
- Rendre distribuables les librairies, dans la mesure du possible ;
- Tester l'ensemble des modules ;

Quelques règles d'un travail bien fait en développement numérique

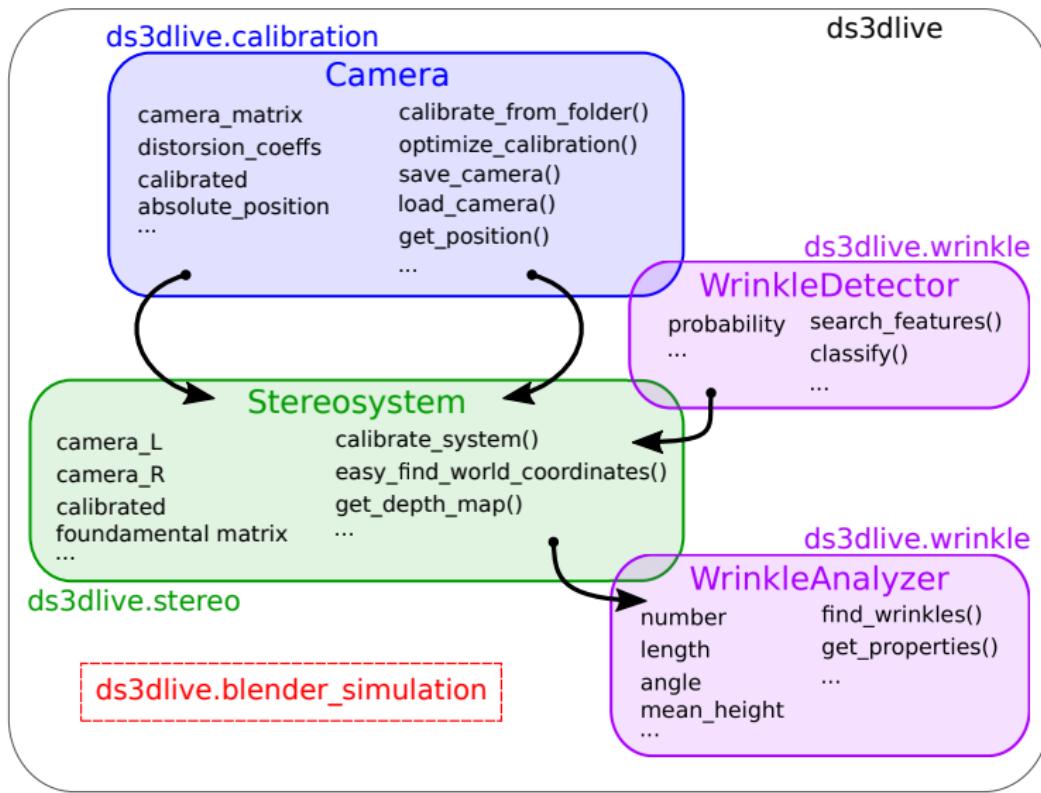
Afin d'assurer une bonne distribution, une utilisation facile pour l'utilisateur et une longue durée de vie à un programme, il est nécessaire de respecter les règles qui suivent :

- Fonctionnaliser les tâches ;
- Donner de la modularité aux algorithmes ;
- User des avantages de la programmation orientée objet ;
- Rendre distribuables les librairies, dans la mesure du possible ;
- Tester l'ensemble des modules ;
- Utiliser des programmes de versionnage qui enregistre toutes les modifications et permettent de retrouver un état antérieur et de collaborer avec d'autres développeurs.

Pour le projet DS3DLive

- Création d'une librairie "ds3dlive" ;
- Création de modules (calibration, simulation, stereo, wrinkle, ...) ;
- Création d'objets adéquats (Camera, Stereosystem, WrinkleDetector, ...) ;
- Projet ouvert sur GitHub (en privé) ;
- Tests unitaires sur l'ensemble des fonctions ;
- Versionnage par l'intermédiaire de Git en utilisant un modèle GitFlow.

Constitution de la librairie



Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

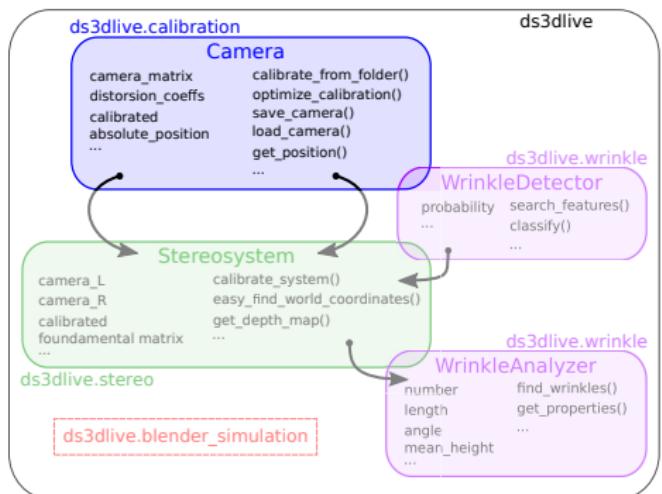
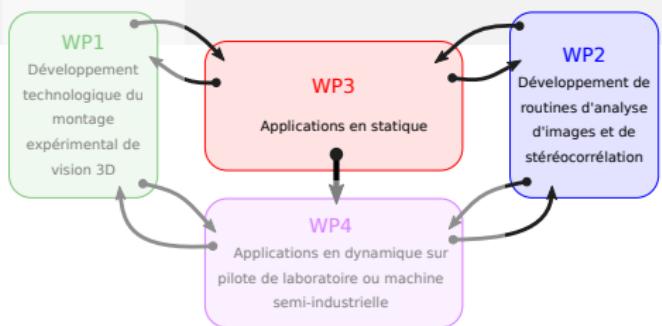
2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques**
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

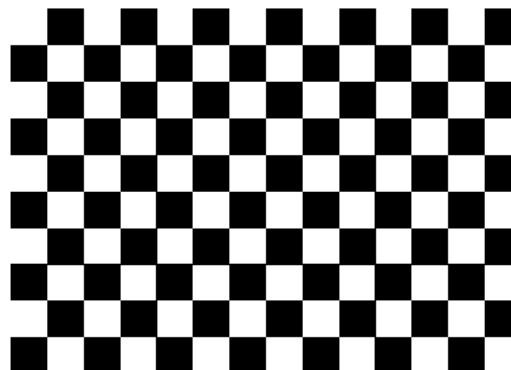
4 Gestion du projet



Méthode par la correspondance points 3D - points image

Mire de calibration

Une mire de calibration est nécessaire. Il s'agit d'un support physique sur lequel sont placés des éléments faciles à segmenter à des positions très précises. La réalisation d'une telle mire permet de déterminer une liste de points 3D (un point par élément à segmenter).



Mise en correspondance

La segmentation des images de la mire permet de générer une liste de points images, associés aux points 3D (il est donc important de les ranger dans le même ordre).

La librairie OpenCV est utilisée ici car elle possède un module de calibration 3D.

Résolution par la minimisation des moindres carrés

On cherche à résoudre pour les n points correspondants

$$\forall p = 1, \dots, n \quad q_p = \begin{pmatrix} u_p \\ v_p \\ 1 \end{pmatrix} \sim P \begin{pmatrix} X_p \\ Y_p \\ Z_p \end{pmatrix} \sim PQ_p \quad \Rightarrow \quad \begin{pmatrix} u_p \\ v_p \end{pmatrix} = \begin{pmatrix} (PQ_p)_1 / (PQ_p)_3 \\ (PQ_p)_2 / (PQ_p)_3 \end{pmatrix}$$

Les équations sont linéaires en les éléments de P et les composantes de P peuvent donc être retrouvées.

Minimisation par les moindres carrés

À cause du bruit dans les images et des erreurs typiques des expérimentations, il serait plus que hasardeux de tomber sur l'unique solution à partir d'un minimum de 12 points. L'ensemble des points sont alors utilisés et on cherche finalement à résoudre l'équation par une méthode des moindres carrés.

Recherche de K , R et t à partir de P

Connaissant les composantes de P il est possible de définir les composantes de K (donc les paramètres intrinsèques, ce qui est souvent recherché) mais aussi les paramètres extrinsèques.

Édition automatique d'une mire de type damier

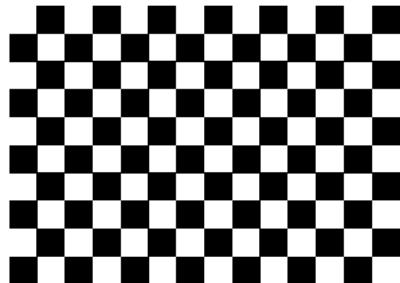
Script \LaTeX

Un script a été écrit pour être compilé en \LaTeX . Il y a trois paramètres :

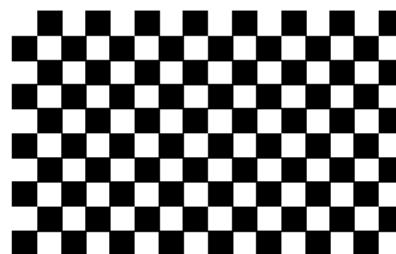
- Taille des carrés en millimètre ;
- Nombre de lignes souhaitées ;
- Nombre de colonnes souhaitées.

La sortie est un fichier PDF (donc image vectorielle) contenant uniquement le damier (aucun espace inutile). Une impression en taille réelle permet d'obtenir une mire très précise.

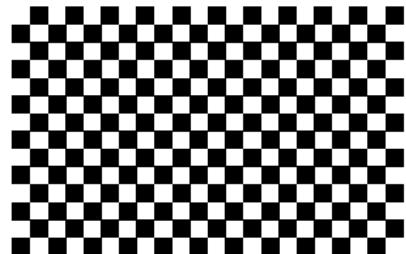
Exemple



25 mm 14 × 10



20 mm 16 × 10



15 mm 22 × 14

▶ Sommaire

Prise d'images avec mire

Règle

Prendre plusieurs images (plus d'une douzaine) avec différents angles et permettant d'observer l'intégralité de la mire.

Exemple

Caméra Gauche

Caméra Droite

Illustration de l'effet de la distorsion

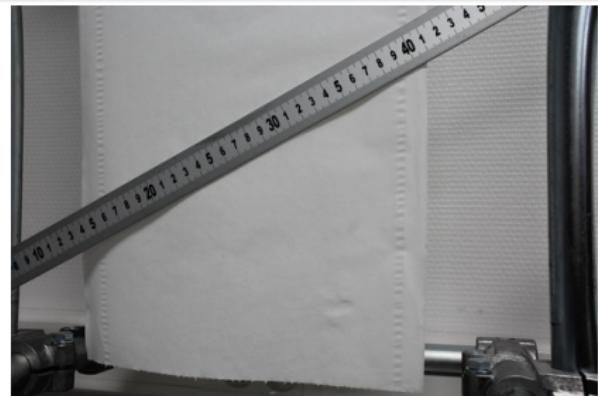
Caméra Droite

- Matrice de caméra :

$$\begin{pmatrix} 5.930\,83 \times 10^3 & 0 & 2.504\,73 \times 10^3 \\ 0 & 5.944\,30 \times 10^3 & 1.769\,74 \times 10^3 \\ 0 & 0 & 1 \end{pmatrix}$$

- Coefficients de distorsion :

$$(-1.47 \times 10^{-1}, 8.82 \times 10^{-1}, 1.42 \times 10^{-3}, -2.39 \times 10^{-3}, -3.59)$$



Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

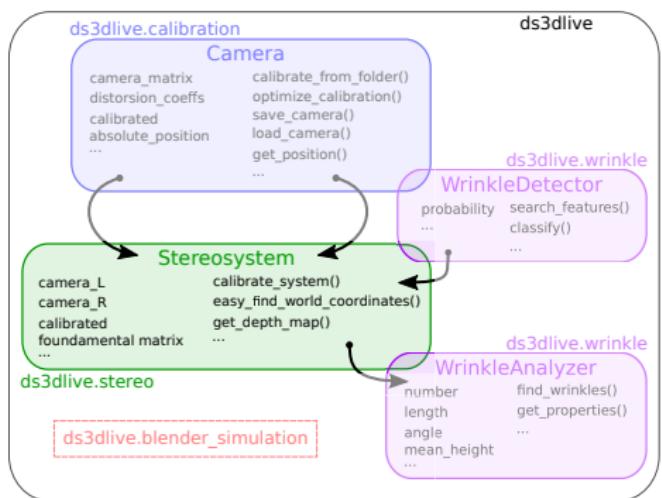
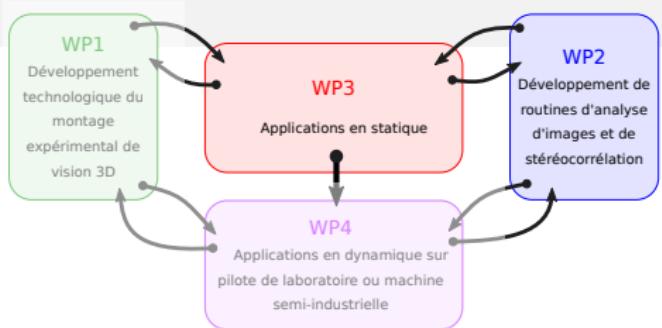
2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique**
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet



Contrainte épipolaire pour la recherche des stéréocorrespondants

Contrainte épipolaire - Matrice fondamentale

Considérant deux points stéréocorrespondants, q_1 et q_2 , la géométrie épipolaire établie une contrainte qui stipule qu'il est possible de trouver, à partir d'un des points stéréocorrespondants, l'autre point en cherchant le long d'une droite.

Mathématiquement, on introduit la matrice fondamentale F de taille 3×3 pour décrire cette contrainte :

$$q_2^T \cdot F \cdot q_1 = 0$$

Contrainte épipolaire calibrée - Matrice essentielle

Contrainte épipolaire en considérant les repères images. On définit maintenant la matrice essentielle E qui correspond à la matrice fondamentale exprimée en longueur métrique (non plus en pixels) :

$$E \sim K_2^T \cdot F \cdot K_1$$

Calibration du système stéréoscopique

Dans le cadre de ce projet, le système stéréoscopique n'est pas voué à changer de position trop souvent. Il y a alors un fort intérêt à calibrer le système en calculant précisément les positions de chacune des caméras. On peut ainsi déterminer directement la matrice essentielle E grâce aux paramètres extrinsèques.

Reconstruction 3D

Globalement, en connaissant les paramètres intrinsèques et extrinsèques

Il est possible de déterminer les coordonnées d'un point Q dans l'espace à partir de la connaissance des ces coordonnées q_i dans i images :

$$\forall i, \quad P_i Q \sim q_i$$

Si l'on ne connaît pas les paramètres extrinsèques

- On détermine la pose d'un objet pour chaque caméra (il s'agit de retrouver la position de la caméra par rapport à l'objet)
- On considère la contrainte épipolaire pour rechercher des points stéréocorrespondants et outrepasser la connaissance des paramètres extrinsèques

Démonstration en direct !

Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

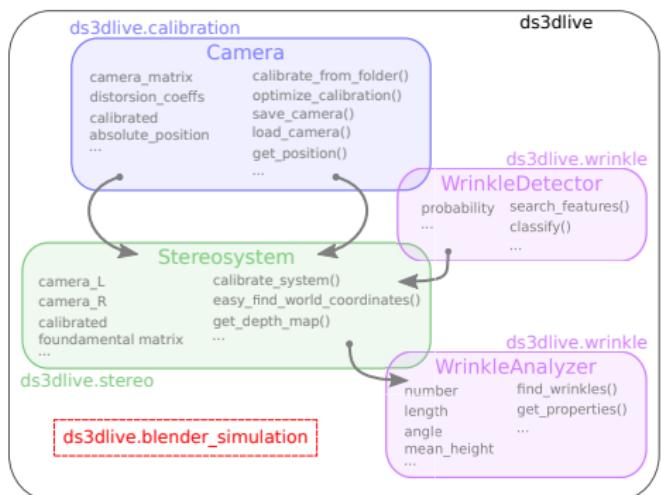
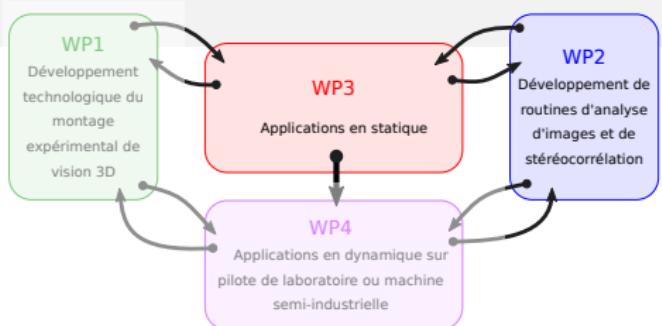
2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D**
- Caractérisation des plis

4 Gestion du projet



Utilisation de Blender

Étapes de réalisation de la simulation

- ➊ Création du projet Blender
- ➋ Ajout de la première caméra
- ➌ Ajout de la seconde caméra
- ➍ Ajout du papier avec pli vertical
- ➎ Ajout d'un "monde" réaliste¹
- ➏ Acquisition des images de rendu

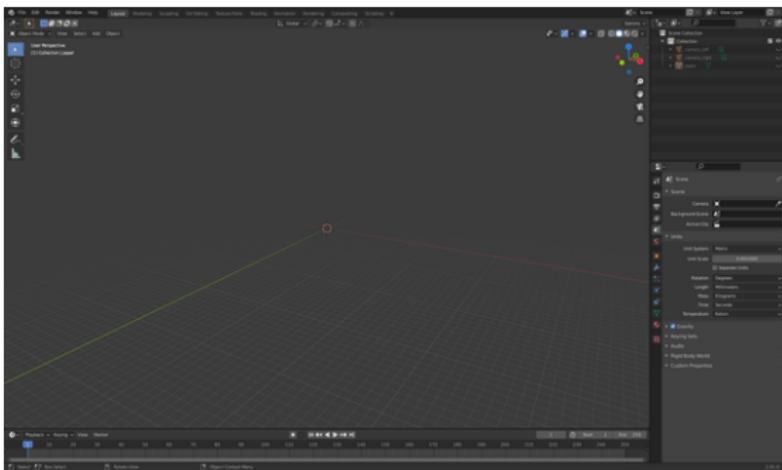


FIGURE – Crédit : Simulation 3D

1. Dont ajout d'un éclairage apporté, qui n'est pas illustré ici.

Utilisation de Blender

Étapes de réalisation de la simulation

- ① Création du projet Blender
- ② Ajout de la première caméra
- ③ Ajout de la seconde caméra
- ④ Ajout du papier avec pli vertical
- ⑤ Ajout d'un "monde" réaliste¹
- ⑥ Acquisition des images de rendu

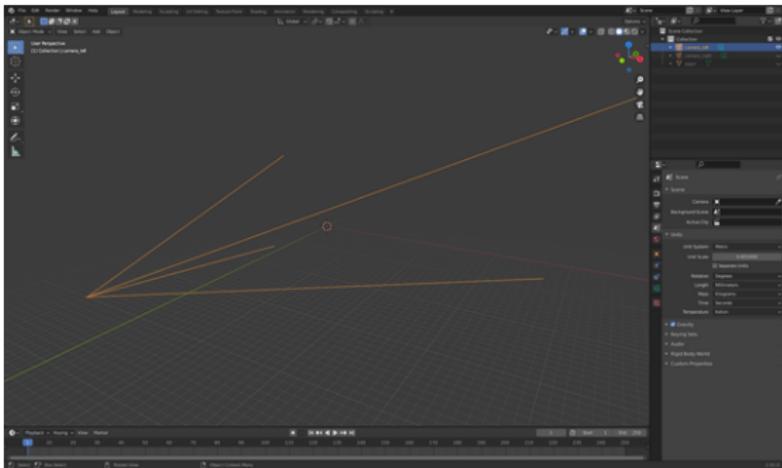


FIGURE – Ajout de la première caméra

1. Dont ajout d'un éclairage apporté, qui n'est pas illustré ici.

Utilisation de Blender

Étapes de réalisation de la simulation

- ① Création du projet Blender
- ② Ajout de la première caméra
- ③ Ajout de la seconde caméra
- ④ Ajout du papier avec pli vertical
- ⑤ Ajout d'un "monde" réaliste¹
- ⑥ Acquisition des images de rendu

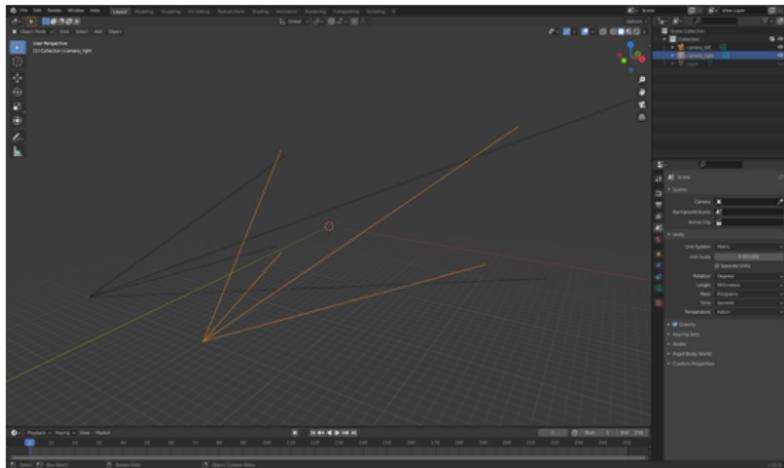


FIGURE – Ajout de la seconde caméra

1. Dont ajout d'un éclairage apporté, qui n'est pas illustré ici.

Utilisation de Blender

Étapes de réalisation de la simulation

- ① Création du projet Blender
- ② Ajout de la première caméra
- ③ Ajout de la seconde caméra
- ④ Ajout du papier avec pli vertical
- ⑤ Ajout d'un "monde" réaliste¹
- ⑥ Acquisition des images de rendu

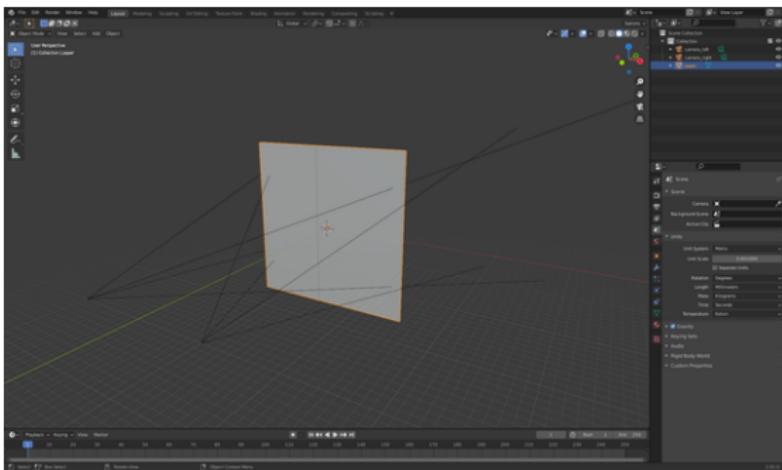


FIGURE – Ajout du papier avec pli vertical

1. Dont ajout d'un éclairage apporté, qui n'est pas illustré ici.

Utilisation de Blender

Étapes de réalisation de la simulation

- ① Création du projet Blender
- ② Ajout de la première caméra
- ③ Ajout de la seconde caméra
- ④ Ajout du papier avec pli vertical
- ⑤ Ajout d'un "monde" réaliste¹
- ⑥ Acquisition des images de rendu

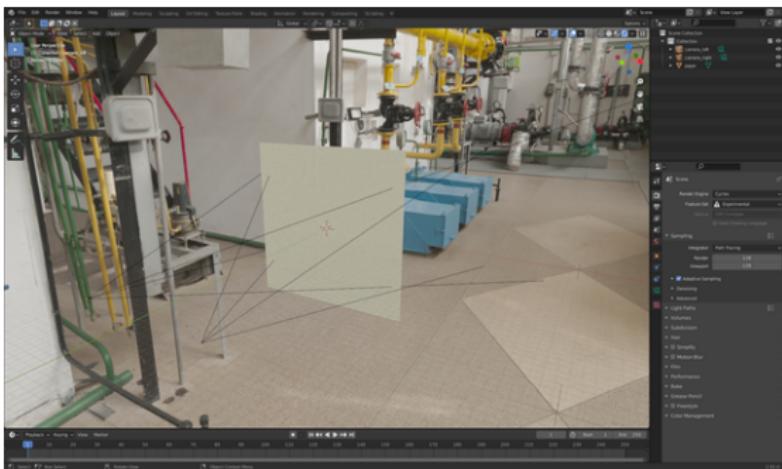


FIGURE – Ajout d'un "monde" réaliste

1. Dont ajout d'un éclairage apporté, qui n'est pas illustré ici.

Utilisation de Blender

Étapes de réalisation de la simulation

- ① Création du projet Blender
- ② Ajout de la première caméra
- ③ Ajout de la seconde caméra
- ④ Ajout du papier avec pli vertical
- ⑤ Ajout d'un "monde" réaliste¹
- ⑥ Acquisition des images de rendu

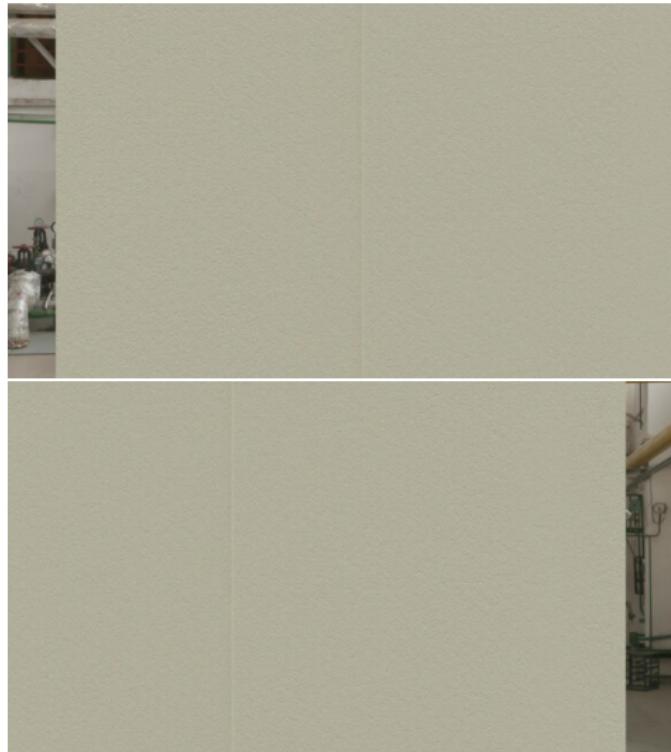


FIGURE – Acquisition des images de rendu

1. Dont ajout d'un éclairage apporté, qui n'est pas illustré ici.

Automatisation de la simulation à partir d'un fichier de paramètres

Utilisation d'un script Python pour l'automatisation des tâches

L'interface de Blender étant essentiellement développée en Python, il est possible d'écrire un script Python qui interagit avec le logiciel. Il est nécessaire pour cela d'utiliser les outils de l'API (Application Programming Interface) de Blender.

Création d'un module pour le projet

Un module permettant l'importation de fonctions et classes appropriées au projet a été créé. Ce module peut être réutilisable pour d'autres simulations du même type.

Simulations paramétriques et classification des résultats

Grâce à la modularité du programme, il est possible de mener des campagnes de simulation en faisant varier les paramètres à notre guise. Par exemple, il est possible de mener, de manière automatique, des simulations pour différents angles des caméras avec des pas de 0.5° .

Les résultats (exemple : les images acquises par les caméras), sont enregistrés dans un dossier avec un fichier rappelant les paramètres utilisés. Un dictionnaire parent est également présent pour retrouvé facilement une simulation à partir de fonctions de tri / filtres.

Paramètres à considérés

Paramètres propre au dispositif

- Distance caméra 1 / caméra 2
- Distance caméras / papier
- Angles des caméras
- Largeur du papier
- Largeur, position et résolution du pli
- Longueurs focales des caméras
- Éclairages
- Autres à venir...

Paramètres physiques - Méthode procédurale pour créer le matériau "papier"

- Texture prédéfinie (pas obligatoire)
- Rugosité
- Autres à venir...
- Effet "subsurface" (transmission partielle de l'éclairage sur l'épaisseur du matériau avec potentielle modification des couleurs)

Paramètres de rendu des images

- Moteur de rendu
- Insertion du monde dans la scène
- Type et méthodes de calcul
- Taille et format des images de sortie
- Nombre d'images moyennées
- Gestion des sauvegardes
- Autres à venir...

Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

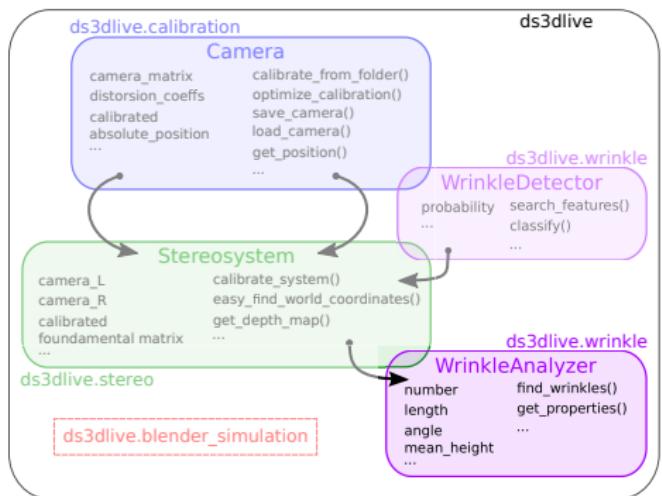
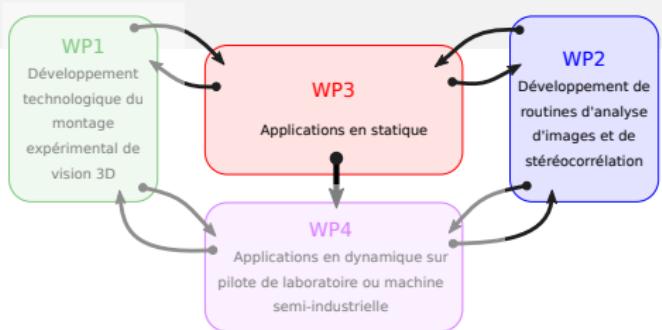
2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet



Caractérisation géométrique d'un pli

Ordre de grandeur des défauts

Une première caractérisation d'un pli a été mené sur Alicona Infinite Focus afin de déterminer un ordre de grandeur des dimensions d'un pli type. Dix-huit points de mesures ont été analysés sur une longueur de 35 mm. La hauteur du pli est en moyenne de 707(74) µm et son épaisseur à mi-hauteur de 680(167) µm. Une résolution permettant d'avoir une précision en Z de l'ordre de 50 µm semble bien adaptée.

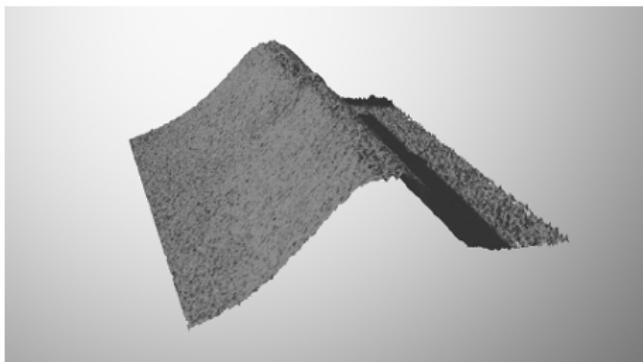
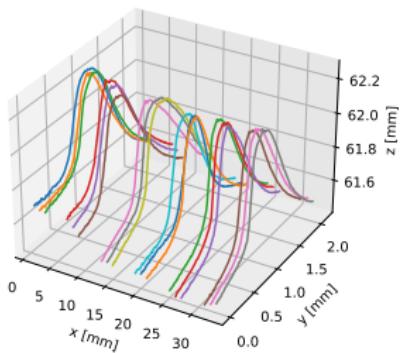


FIGURE – Vue 3D d'une portion mesurée

FIGURE – Profils linéaires d'un pli en 18 mesures sur la longueur

Dans cette partie...

1 État d'avancement

- Rappel sur l'organisation de travail
- Planning

2 Montage du dispositif en statique

- Assemblage structurel et fonctionnel
- Systèmes optiques

3 Travaux numériques

- Librairie Python ds3dlive
- Calibration des systèmes optiques
- Système stéréoscopique
- Simulation 3D
- Caractérisation des plis

4 Gestion du projet

Perspectives

Certaines tâches n'ont pas encore été traitées et vont l'être dans les prochains mois :

- Traitement des images avec la librairie IPSDK.
- Développement d'un détecteur de plis : la méthode SIFT pourrait être utilisée avec la librairie IPSDK ?
- Optimisation de la mesure 3D avec la triangulation [Hartley, 1997].

Gestion du projet

- Communication régulière sur les réseaux LinkedIn et ResearchGate à partir de maintenant.
- Dépenses réalisées en matériel : 1612.85 €.
- Collaborations sur le projet GitHub ? Privé / public ?

Merci de votre attention

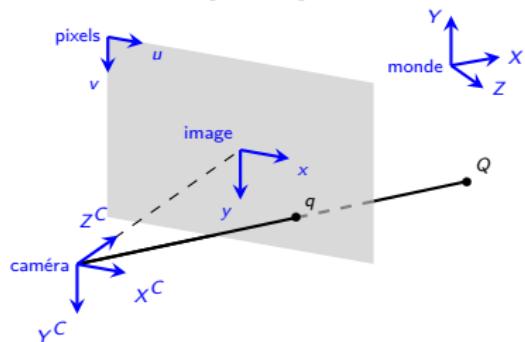
Avez-vous des questions ? Des attentes particulières ?

Références :

- Faugeras, O., "Three-dimensional Computer Vision : A Geometric Viewpoint", MIT Press, 1993.
- Goulette, F., "Modélisation 3D automatique : outils de géométrie différentielle", Presse des Mines, 1999.
- Abdullah, S. C., Amari, M. D. and Ohka, M. "Stereo Vision for Visual Object Tracking and Distance Measurement Assessment", Journal of Mechanical Engineering, Vol 16(1), 121-134, 2019.
- Orteu, J.-J., "Mesure 3D de formes et de déformations par stéréovision", Les Techniques de l'Ingénieur, 2002.
- Graba, T., Granado, B., Romain, O., Ea, T., Pinna, A., Garda, P., "Reconstruction 3D temps réel dans un VSIP", Groupe d'Etudes du Traitement du Signal et des Images, 2005.
- Hartley, R. I. and Sturm, P., "Triangulation", Computer Vision and Image Understanding 68, 1997.

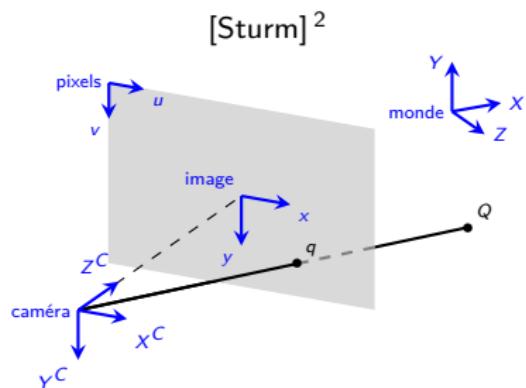
Modèle de sténopé (pinhole) : illustration et notation des repères

[Sturm]²



- Repère monde : $Q(X, Y, Z)$
- Repère caméra : $Q^C(X^C, Y^C, Z^C)$, centre de projection comme origine et axe optique qui oriente l'axe selon Z^C . X^C et Y^C sont orientés selon les pixels.
- Repère image : $q_i(x, y)$, origine au point principal
- Repère des pixels : $q(u, v)$, origine dans un coin de l'image.

Modèle de sténopé (pinhole) : illustration et notation des repères



- Repère monde : $Q(X, Y, Z)$
- Repère caméra : $Q^C(X^C, Y^C, Z^C)$, centre de projection comme origine et axe optique qui oriente l'axe selon Z^C . X^C et Y^C sont orientés selon les pixels.
- Repère image : $q_i(x, y)$, origine au point principal
- Repère des pixels : $q(u, v)$, origine dans un coin de l'image.

L'étude de la projection de points de l'espace à 3 dimensions dans un espace à 2 dimension, nous amène à travailler avec les coordonnées homogènes. Le symbole \sim correspond à une égalité vectorielle définie à un facteur scalaire près : $\exists s \in \mathbb{R}, \bar{q} \sim \bar{Q} \Leftrightarrow s\bar{q} = \bar{Q}$

Modèle algébrique : passage du repère caméra au repère image

Distance focale

On appelle distance focale, notée f , la distance qui relie le centre de projection au point principal. C'est-à-dire la distance entre l'origine du repère caméra et l'origine du repère image.

On montre par la géométrie :

$$\begin{cases} x = f \frac{X^C}{Z^C} \\ y = f \frac{Y^C}{Z^C} \end{cases} \Leftrightarrow q_i = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \sim \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X^C \\ Y^C \\ Z^C \\ 1 \end{pmatrix}$$

$$q_i \sim \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} Q^C$$

Modèle algébrique : passage du repère image au repère des pixels

Densité de pixels

On appelle densité de pixel la grandeur qui détermine le nombre de pixel présents pour une unité de longueur selon une direction. Elle est généralement exprimée en px/mm. Pour une image, on appelle k_u et k_v les densités de pixel suivant les axes des coordonnées u et v , respectivement.

Translation de repère

On appelle x_0 et y_0 les coordonnées du coin origine du repère des pixels dans le repère image. Alors la transformation reliant les deux repères subit une translation de vecteur (x_0, y_0) .

Changement d'unité et translation

$$q = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} k_u & 0 & 0 \\ 0 & k_v & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} k_u & 0 & k_u x_0 \\ 0 & k_v & k_v y_0 \\ 0 & 0 & 1 \end{pmatrix} q_i$$

$$q \sim \begin{pmatrix} fk_u & 0 & k_u x_0 & 0 \\ 0 & fk_v & k_v y_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} Q^c$$

Modèle algébrique : passage du repère monde au repère caméra

Translation et rotation caractérisent la transformation

Soit \bar{t} les coordonnées du centre de projection dans le repère monde. Alors le vecteur \bar{t} représente également le vecteur translation du repère monde vers le repère caméra.

Soit \bar{R} la matrice rotation du repère monde vers le repère caméra.

$$Q^C = \begin{pmatrix} X^C \\ Y^C \\ Z^C \end{pmatrix} = \bar{R} (Q - \bar{t}) = \bar{R} Q - \bar{R} \bar{t} = \bar{R} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - \bar{R} \bar{t}$$

En coordonnées généralisées :

$$\begin{pmatrix} X^C \\ Y^C \\ Z^C \\ 1 \end{pmatrix} = \begin{pmatrix} \bar{R} & -\bar{t} \\ \bar{0}^T & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$q = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \begin{pmatrix} fk_u & 0 & k_u x_0 & 0 \\ 0 & fk_v & k_v y_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \bar{R} & -\bar{t} \\ \bar{0}^T & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Modèle algébrique complet

Matrice de projection P

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \begin{pmatrix} fk_u & 0 & k_u x_0 & 0 \\ 0 & fk_v & k_v y_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \bar{\bar{R}} & -\bar{t} \\ \bar{0}^T & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \sim \bar{\bar{P}} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Matrice de caméra K

On pose $\bar{K} = \begin{pmatrix} fk_u & 0 & k_u x_0 \\ 0 & fk_v & k_v y_0 \\ 0 & 0 & 1 \end{pmatrix}$, alors le modèle devient :

$$P \sim (\bar{K} \quad \bar{0}) \begin{pmatrix} \bar{\bar{R}} & -\bar{t} \\ \bar{0}^T & 1 \end{pmatrix} \sim (\bar{K}\bar{R} \quad -\bar{K}\bar{R}\bar{t}) \sim \bar{K}\bar{R} (\bar{I} \quad -\bar{t})$$

Paramètres intrinsèques et extrinsèques

- Les paramètres intrinsèques sont définis par la matrice de caméra K : ils sont propres à la caméra (densité de pixels, distance focale, alignement des rangées de pixels)
- Les paramètres extrinsèques sont définis par la matrice de rotation R et le vecteur translation t : ils sont propres à la position de la caméra.

Détection automatique des plis

Intérêt de la détection automatique

L'objectif du projet est d'être capable de détecter et caractériser les plis qui se forment lors de la phase de production. Cela veut dire que le temps de traitement doit être **rapide**. De plus, il est inutile de chercher à traiter les images si aucun pli n'est présent : la détection du pli sert de **condition nécessaire** pour entamer une démarche plus lourde qui consiste à analyser la scène par stéréovision.

Vision par ordinateur et classification d'images

La détection d'un pli ne correspond à rien d'autre qu'un problème de classification binaire ("Pas de pli" / "Présence de pli"). Une stratégie existante pour ce genre de travail consiste à réaliser successivement les trois phases qui suivent :

- Détection des features (parties de l'image qui contiennent des informations propres à l'image et qui permettent de renseigner sur les éléments en présence) ;
- Description des features : trouver un moyen mathématique de décrire efficacement les features pour pouvoir les comparer à d'autres ;
- Classification des images à partir des données issues de la description des features.

Détection et description des features

Algorithme SIFT

Détection et description automatique des features à partir de l'information propre de l'image. Cet algorithme est largement utilisé dans le domaine de la vision par ordinateur car il possède les propriétés suivantes :

- Automatisation de la tâche ;
- Invariance au changement d'échelle et à la rotation ;
- Très faible sensibilité au variations de contraste et au changement de luminosité.

Aujourd'hui l'algorithme existe sous différentes formes dont certaines plus adaptées à nos objectifs. Il existe notamment une variante rapide permettant le "video tracking".

Quelques références

- David G Lowe. Object recognition from local scale-invariant features. *Seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.
- J Liu, et al. Image matching based on improved sift algorithm. *Chinese Journal of Scientific Instrument*, 34(5) :1107–1112, 2013.
- Jian Wu, et al. A comparative study of sift and its variants. *Measurement science review*, 13(3) :122–131, 2013.
- Xuelong Hu, et al. Video object matching based on sift algorithm. In *2008 International Conference on Neural Networks and Signal Processing*, pages 412–415. IEEE, 2008.
- Wikipédia. Scale-invariant feature transform — wikipédia, l'encyclopédie libre, 2020.

Détection et description des features

Algorithme SIFT - Détection des features

Plusieurs étapes sont nécessaires pour détecter les features :

- ① Création de l'espace des échelles ;
- ② Calculs des différences des gaussiennes (DoG) ;
- ③ Recherche des maxima locaux et sélection des points d'intérêt par l'utilisation de masques et de méthodes de seuillage.

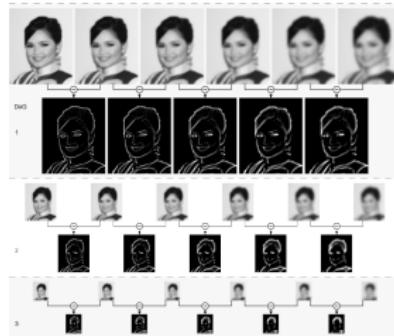


Détection et description des features

Algorithme SIFT - Détection des features

Plusieurs étapes sont nécessaires pour détecter les features :

- ① Création de l'espace des échelles ;
- ② Calculs des différences des gaussiennes (DoG) ;
- ③ Recherche des maxima locaux et sélection des points d'intérêt par l'utilisation de masques et de méthodes de seuillage.

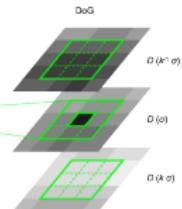
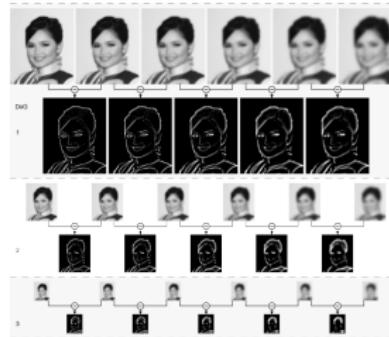


Détection et description des features

Algorithme SIFT - Détection des features

Plusieurs étapes sont nécessaires pour détecter les features :

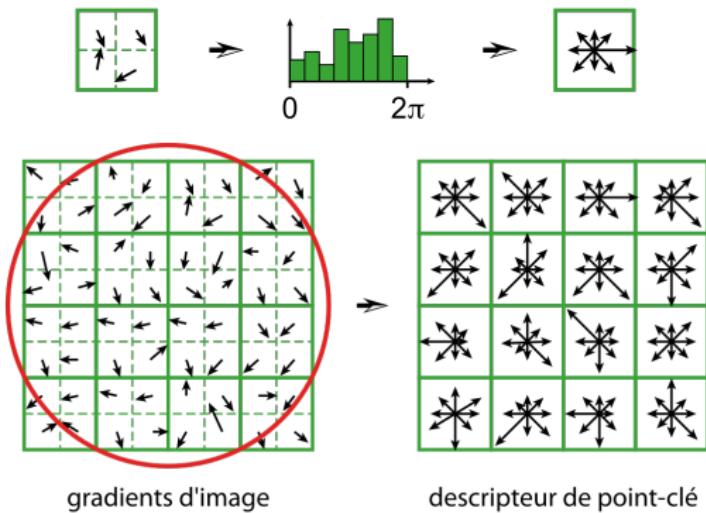
- ① Création de l'espace des échelles ;
- ② Calculs des différences des gaussiennes (DoG) ;
- ③ Recherche des maxima locaux et sélection des points d'intérêt par l'utilisation de masques et de méthodes de seuillage.



Détection et description des features

Algorithme SIFT - Description des features

- ① Calcul de l'orientation au voisinage des points d'intérêt ;
- ② Soustraction de l'orientation principale et concaténation des histogrammes avec post-traitement.



Classification à partir des descripteurs

Méthodes de l'IA

Sont présentées ici les différentes méthodes de l'intelligence artificielle pour classifier des images. Chaque méthode consiste à trouver les bonnes relations entre les données d'entrée. L'évaluation de ces relations se fait généralement par une méthode de pondération lors d'une phase d'apprentissage. Durant cette phase, des images sont données en entrée et le résultat est connu.

Méthodes classiques de classification par Machine learning

Les descripteurs peuvent être utilisés comme données d'entrée pour les algorithmes suivants :

- k Plus proches voisins
- Régression logistique
- Support vector machine (SVM)
- Forêt d'arbres aléatoires
- Réseaux de neurones

Méthode d'apprentissage par Deep learning

Les réseaux de neurones convolutifs sont réputés en reconnaissance d'images et très souvent utilisés car le choix des features est fait automatiquement en fonction de la classification attendue.

▶ Sommaire

Réalisation de la calibration des caméras dans le projet

Réalisation

L'ensemble des codes nécessaires au calculs de calibration des caméras a été écrit. Il subsiste néanmoins quelques interrogations...

Effet de l'ouverture et de la mise au point

En ce qui concerne l'ouverture :

