SSL Seminars
Five-month internship in partnership with ISAE-Supaero

Calibration and Fusion of Stereoscopic and Time-of-Flight Cameras for Zero Gravity Targets Inspection

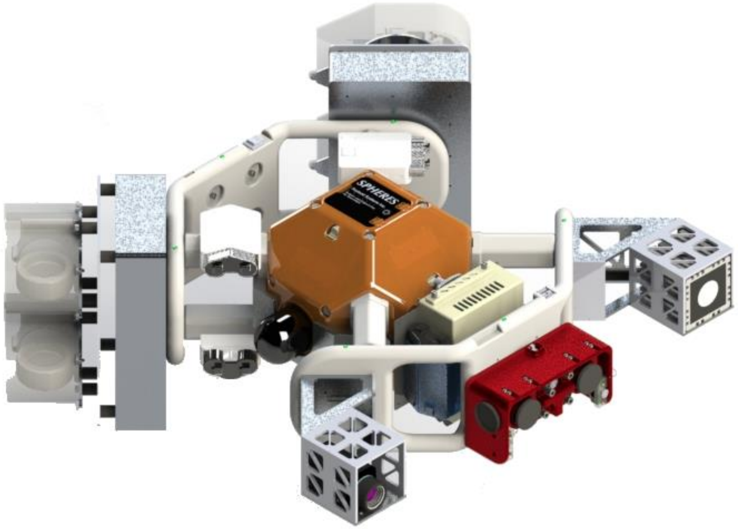
Gabriel Urbain - Graduate student - ISAE Supaero – France

Wednesday, October 29th 2014

Overview

- Introduction
- Sensors Features and Acquisition
  - Time-of-Flight Camera (Optical Range Finder)
  - Stereoscopic Cameras (VERTIGO)
  - Thermocam
- Calibration Algorithm
  - ORF calibration
  - Stereo calibration
  - System calibration
- Fusion Algorithm
- Conclusions and Perspectives

Introduction

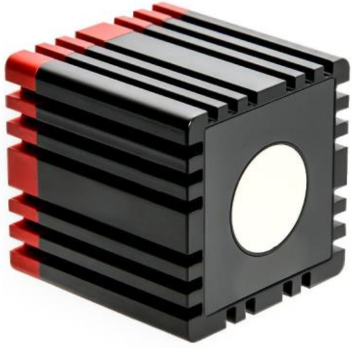


INSPECT is an extension of SPHERES intending to test pieces of hardware inside the ISS prior to moving into the vacuum of space

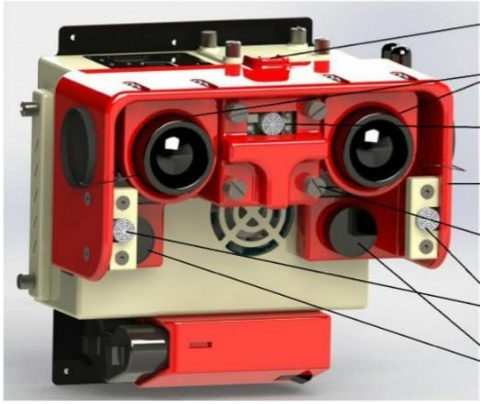
Introduction

INSPECT adds two cameras to VERTIGO: a thermocam and an Optical Range Finder to perform Visual Navigation.

Thermocam



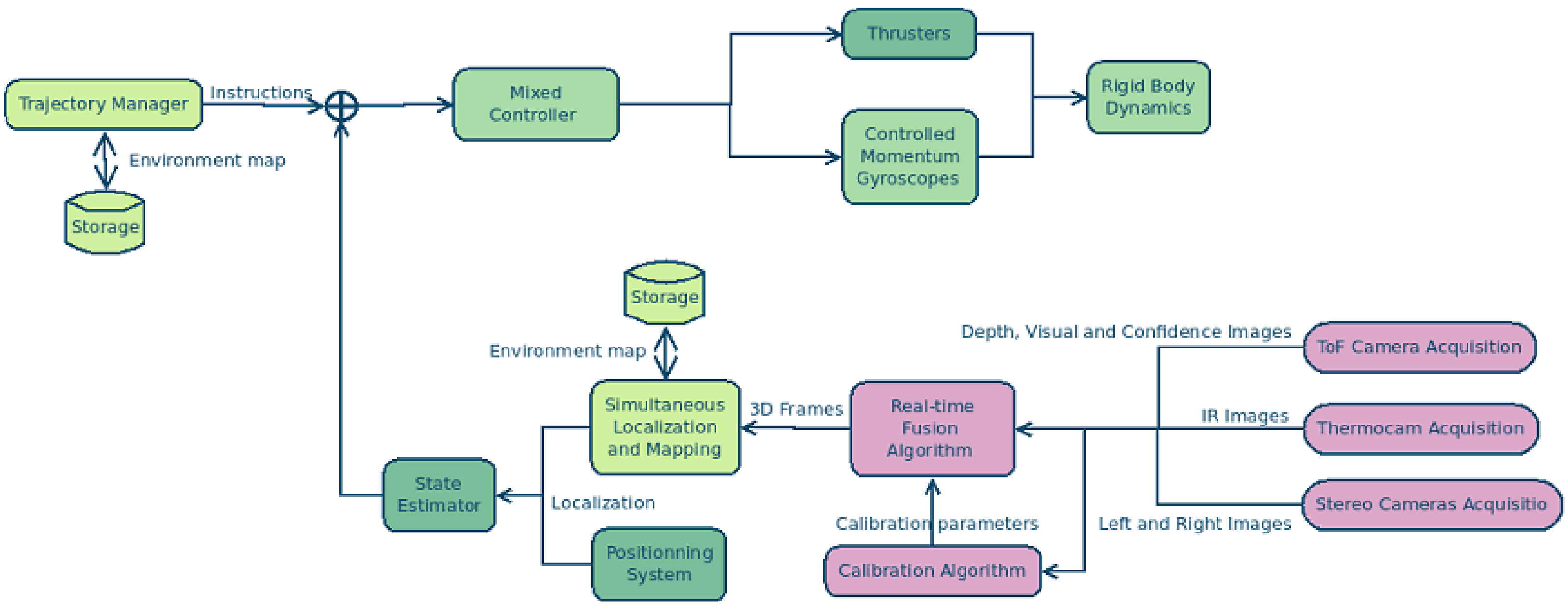
Optical Range Finder or Time-of-Flight Camera



Stereoscopic Cameras of the VERTIGO Goggles

Introduction

Computer Vision enables Simultaneous Localization and Mapping.



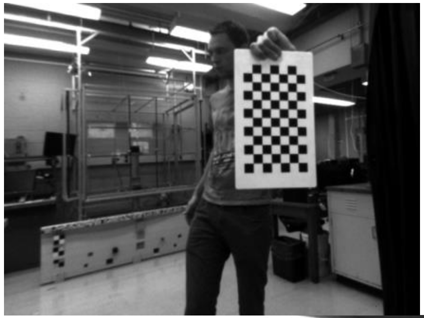
Overview

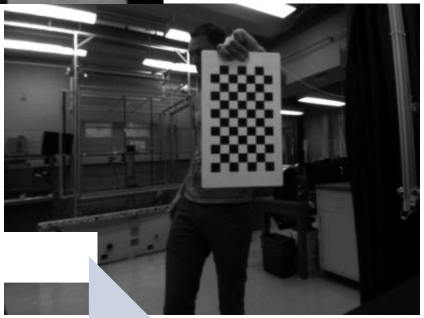
- Introduction
- Sensors Features and Acquisition
  - Time-of-Flight Camera (Optical Range Finder)
  - Stereoscopic Cameras (VERTIGO)
  - Thermocam
- Calibration Algorithm
  - ORF calibration
  - Stereo calibration
  - System calibration
- Fusion Algorithm
- Conclusions and Perspectives

Sensors Features and Acquisition

VERTIGO goggles:

Feature	Value
Frequency Domain	$717nm$ dominant (visible)
Resolution	752x480 pixels
Pixel Size	$6\mu m$
FoV	35°x35°
Output Data	10 bits monochrome



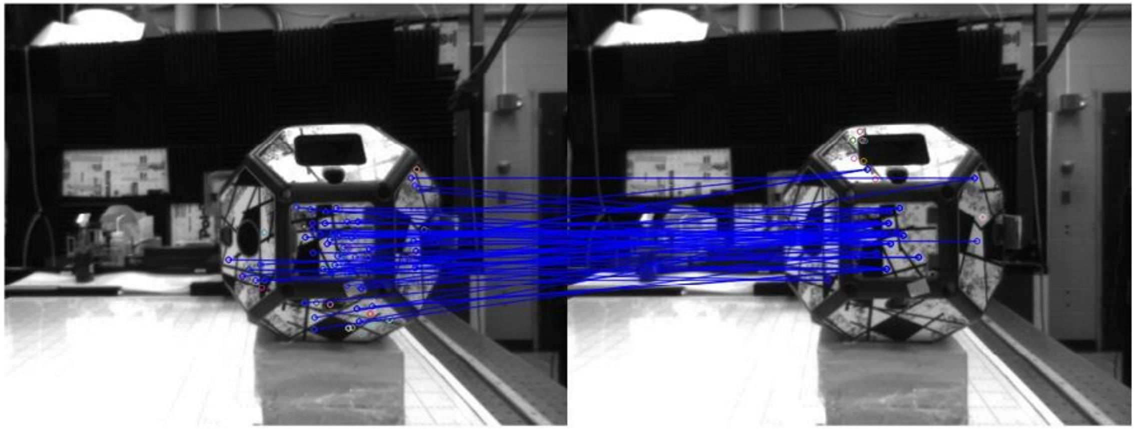


Two images are captured synchronously

Stereo Cameras → uEye API → Feature Detection and matching → Optional: Aggregation and Optimization → Triangulation

Sensors Features and Acquisition

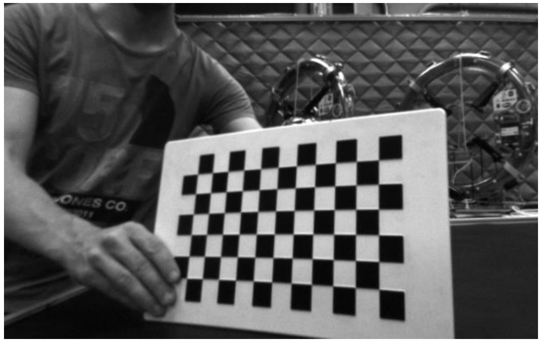
VERTIGO goggles:

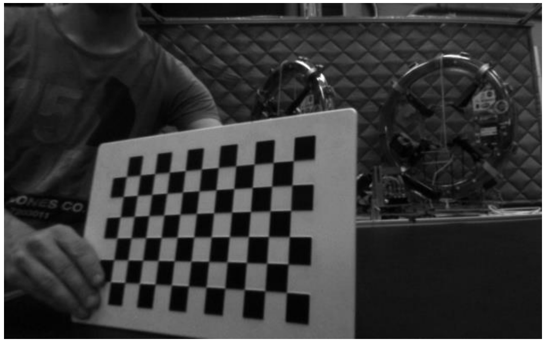


Stereo Cameras → uEye API → Feature Detection and matching → Optional: Aggregation and Optimization → Triangulation

Sensors Features and Acquisition

VERTIGO goggles:



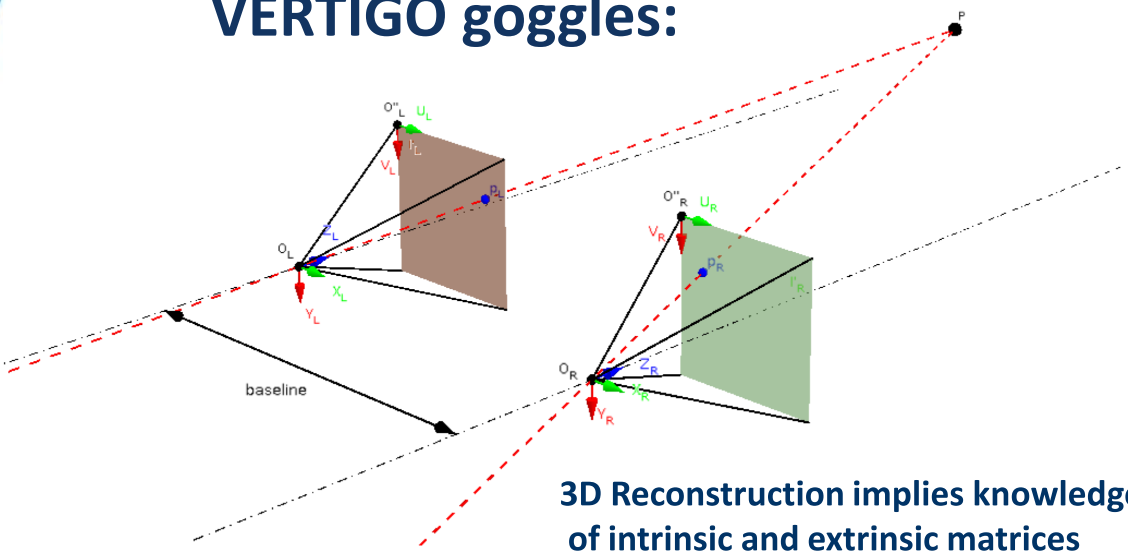


Low-textured objects lead to bad stereo 3D estimations

Stereo Cameras → uEye API → Feature Detection and matching → Optional: Aggregation and Optimization → Triangulation

Sensors Features and Acquisition

VERTIGO goggles:



$$\begin{pmatrix} u_L \\ v_L \\ 1 \end{pmatrix} = K_L * M_L * \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} u_R \\ v_R \\ 1 \end{pmatrix} = K_R * M_R * \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$\Rightarrow \begin{cases} z = \dfrac{baseline}{u_L - u_R} \\ x = \dfrac{(u_L - c_U) \; * z}{f} \\ y = \dfrac{(u_L - c_V) * z}{f} \end{cases}$$

3D Reconstruction implies knowledge of intrinsic and extrinsic matrices

Stereo Cameras → uEye API → Feature Detection and matching → Optional: Aggregation and Optimization → **Triangulation**

AEROASTRO

MIT SSL

Sensors Features and Acquisition

ORF:

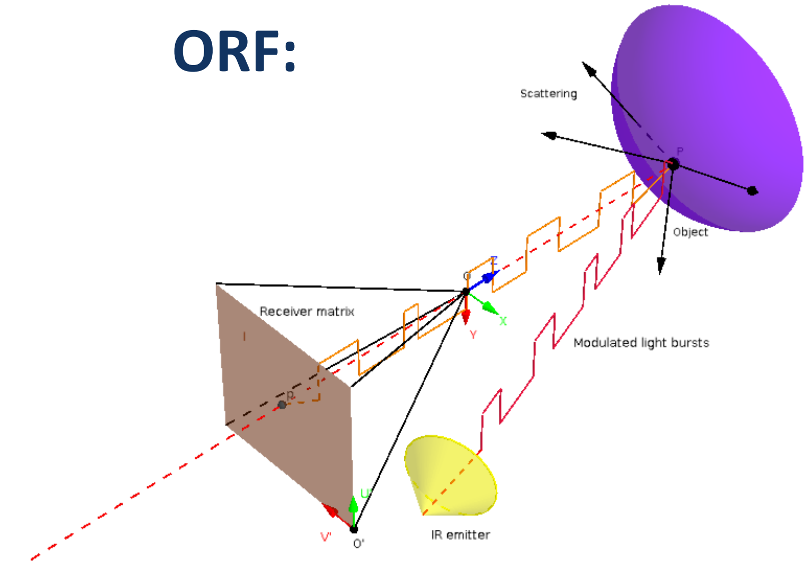
Feature	Value
Frequency Domain	$850nm$ (far IR)
Resolution	176x144 pixels
Pixel Size	$40\mu m$
FoV	69°x55°
Output Data	14 bits depth, 16 bits visual, 16 bits confidence

Three images are captures in the same time: depth, visual, confidence

ORF Camera → MESA-Imaging API → 3D Reconstruction

Sensors Features and Acquisition

ORF:

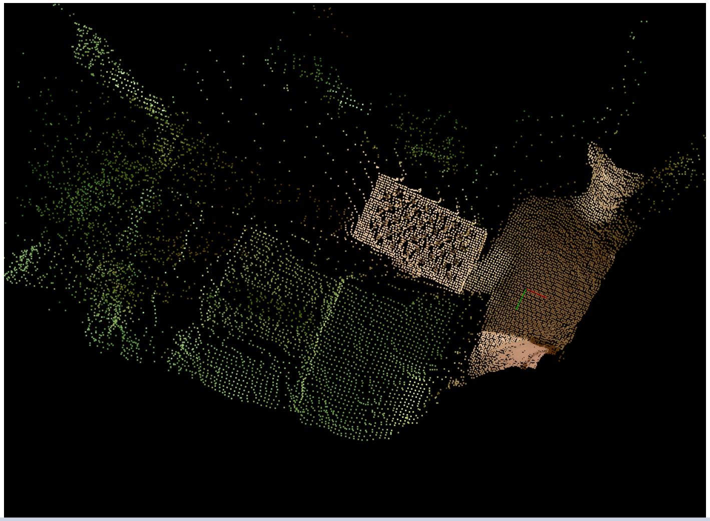


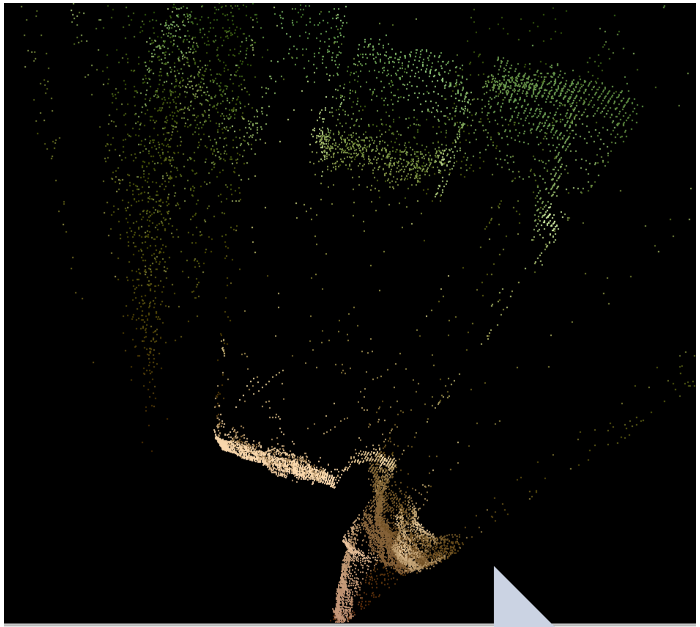
Three images are captures in the same time: depth, visual, confidence

ORF Camera → MESA-Imaging API → 3D Reconstruction

Sensors Features and Acquisition

ORF:



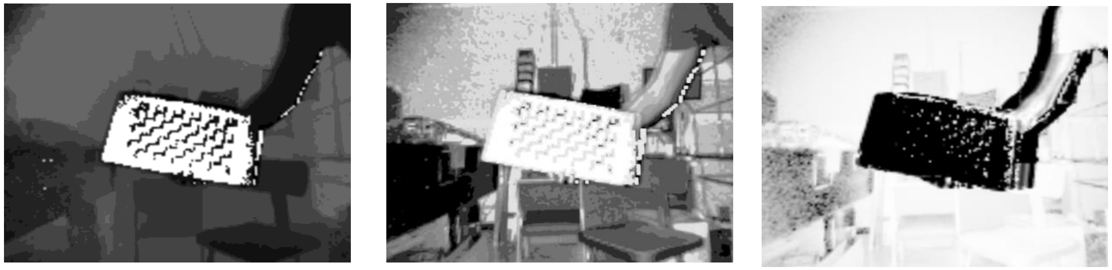


ORF Camera → MESA-Imaging API → 3D Reconstruction

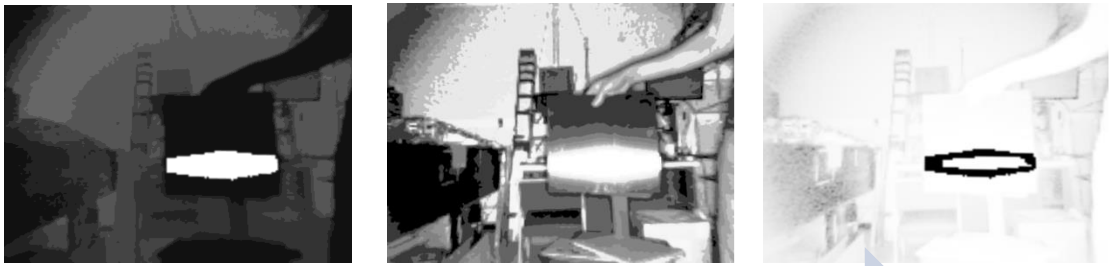
Sensors Features and Acquisition

ORF:

ORF is motion sensitive



ORF is material sensitive

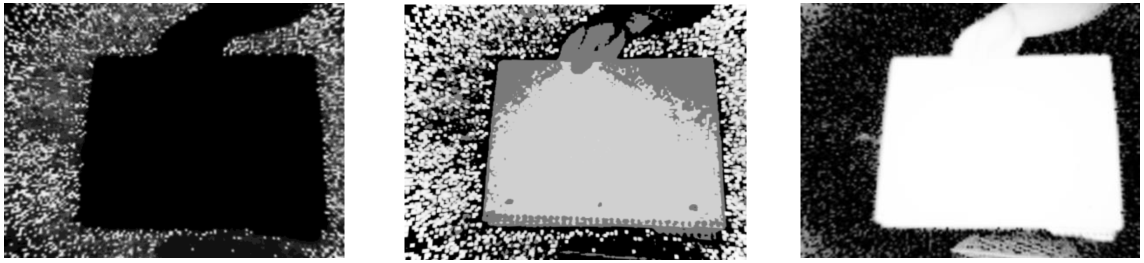


ORF Camera → MESA-Imaging API → 3D Reconstruction

Sensors Features and Acquisition

ORF:

ORF is exposure sensitive



ORF is range sensitive

ORF Camera → MESA-Imaging API → 3D Reconstruction

Sensors Features and Acquisition

Thermocam:

Feature	Value
Frequency Domain	$7.5 - 13nm$ (near IR)
Resolution	80x64 pixels
Pixel Size	$50\mu m$
FoV	44°x36°
Output Data	8 bits monochrome

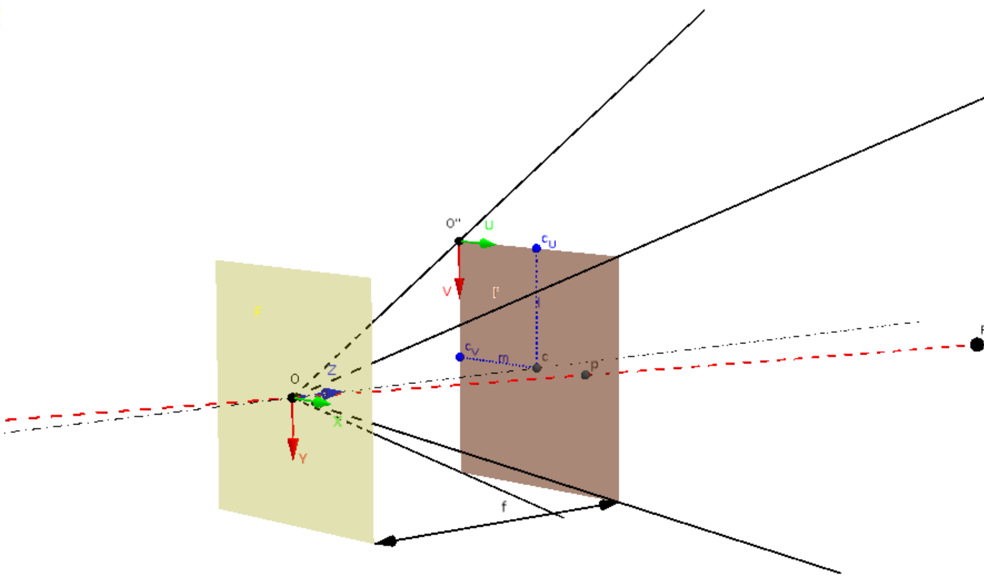
The thermocam provides data diversity but with a low resolution

Overview

- Introduction
- Sensors Features and Acquisition
  - Time-of-Flight Camera (Optical Range Finder)
  - Stereoscopic Cameras (VERTIGO)
  - Thermocam
- Calibration Algorithm
  - ORF calibration
  - Stereo calibration
  - System calibration
- Fusion Algorithm
- Conclusions and Perspectives

Calibration Algorithm

ORF Calibration (intrinsic):



Intrinsic parameters θ_T^i correspond to intrinsic matrix:

$$K_T = \begin{pmatrix} f_U & s_{UV} & c_U \\ 0 & f_V & c_V \\ 0 & 0 & 1 \end{pmatrix}$$

and distortion coefficients :

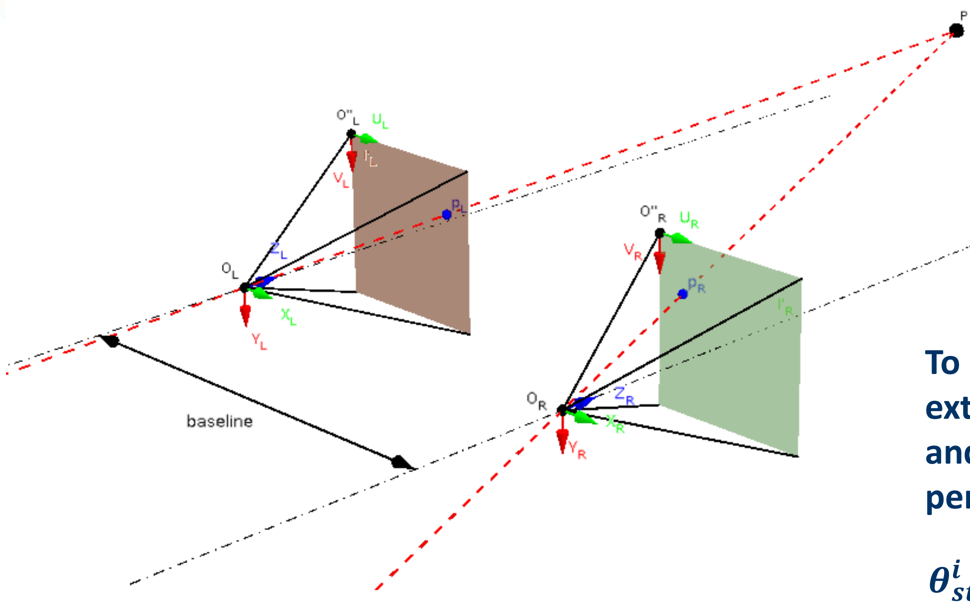
$$k_1, k_2, k_3, k_4, k_5$$

To compute extrinsic parameters, m checkerboard corners are projected and an iterative minimization process is performed:

$$\theta_T^i = arg\,min \left\{ \sum_{i=0}^{m} \left\| \mathcal{F}(\hat{\theta}_T^i) - p_T^i \right\|^2 \right\}$$

Calibration Algorithm

VERTIGO Calibration (intrinsic + extrinsic):



Extrinsic parameters corespond to the transformation matrix:

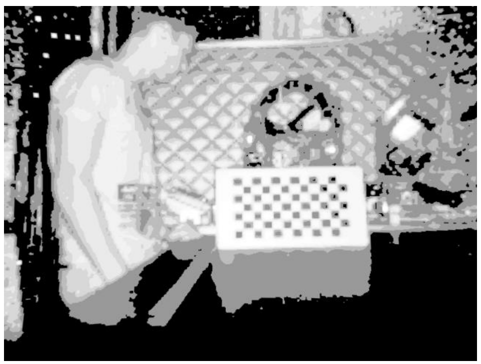
$$M_{LR} = \begin{pmatrix} r_{X^L X^R} & r_{X^L Y^R} & r_{X^L Z^R} & t_{X^L X^R} \\ r_{Y^L X^R} & r_{Y^L Y^R} & r_{Y^L Z^R} & t_{Y^L Y^R} \\ r_{Z^L X^R} & r_{Z^L Y^R} & r_{Z^L Z^R} & t_{Z^L Z^R} \end{pmatrix}$$

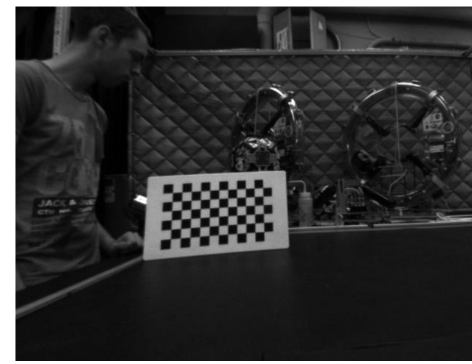
To compute stereo parameters (intrinsic + extrinsic), m checkerboard corners are projected and an iterative minimization process is performed:

$$\boldsymbol{\theta}^{i}_{stereo} = \boldsymbol{arg\,min} \left\{ \sum_{i=0}^{m} \left\| \mathcal{F}\left(\widehat{\boldsymbol{\theta}}^{i}_{stereo}\right) - \begin{pmatrix} p_L \\ p_R \end{pmatrix} \right\|^2 \right\}$$

Calibration Algorithm

HALO Calibration (extrinsic):





Three sets of points are deduced: $p_T^i = \begin{pmatrix} u \\ v \end{pmatrix}_T^i$ $p_L^i = \begin{pmatrix} u \\ v \end{pmatrix}_L^i$ $p_R^i = \begin{pmatrix} u \\ v \end{pmatrix}_R^i$

Stereo capture → Checkerboard Corners Detection → Stereo Triangulation

ORF capture → Checkerboard Corners Detection → ORF Pinhole Inversion

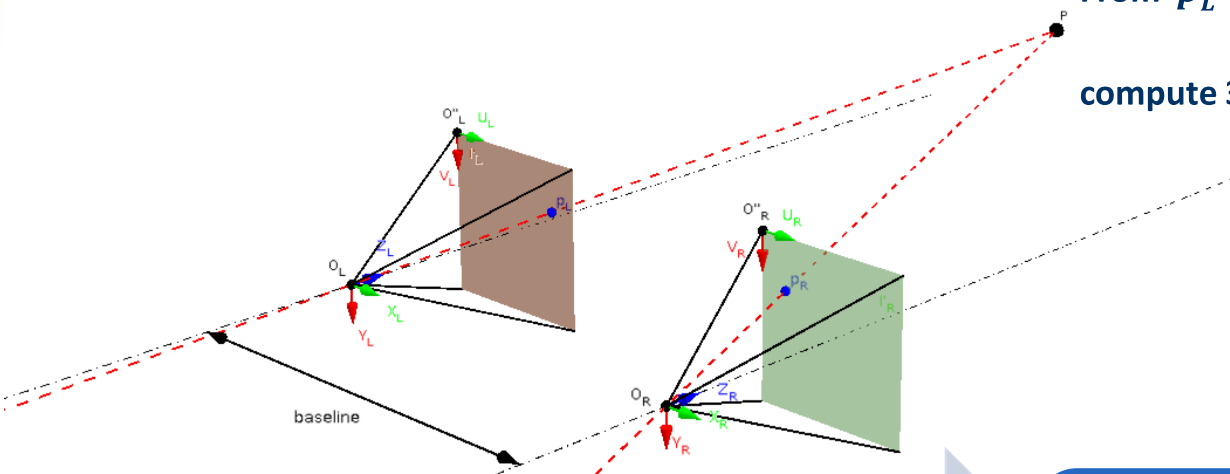
→ Horn's minimization algorithm in a RANSAC process

AEROASTRO

Calibration Algorithm

HALO Calibration (extrinsic):

From $p_L^i = \begin{pmatrix} u_L \\ v_L \end{pmatrix}^i$ **and** $p_R^i = \begin{pmatrix} u_R \\ v_R \end{pmatrix}^i$**, we compute 3D points** $P_{stereo}^i = \begin{pmatrix} x_L \\ y_L \\ z_L \end{pmatrix}$**:**



$$\begin{cases} z_L = \dfrac{t_{RL}}{u_L - u_R} \\ x_L = \dfrac{(u_L - c_U) z_L}{f} \\ y_L = \dfrac{(v_L - c_V) z_L}{f} \end{cases}$$

Stereo capture → Checkerboard Corners Detection → Stereo Triangulation →

ORF capture → Checkerboard Corners Detection → ORF Pinhole Inversion →

Horn's minimization algorithm in a RANSAC process

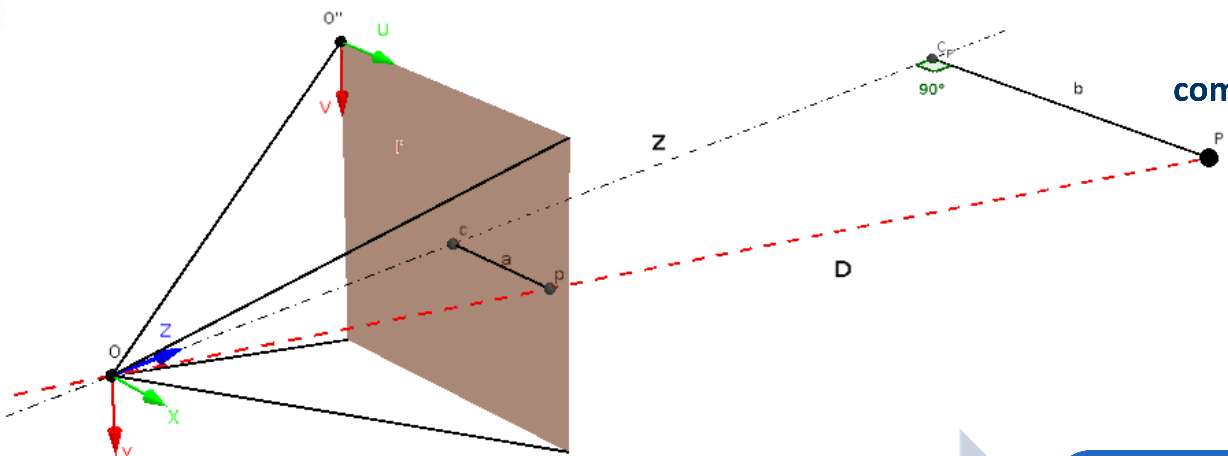
AEROASTRO

Calibration Algorithm

HALO Calibration (extrinsic):

From $p_T^i = \begin{pmatrix} u_T \\ v_T \end{pmatrix}^i$ and d_T^i, we compute 3D points $P_T^i = \begin{pmatrix} x_T \\ y_T \\ z_T \end{pmatrix}$:

$$\begin{cases} x_T = \dfrac{u_T - c_U}{z_T} f \\ y_T = \dfrac{v_T - c_V}{z_T} f \\ \dfrac{f}{z_T} = \dfrac{a}{b} \\ d_T^2 = z_T^2 + b^2 \end{cases}$$



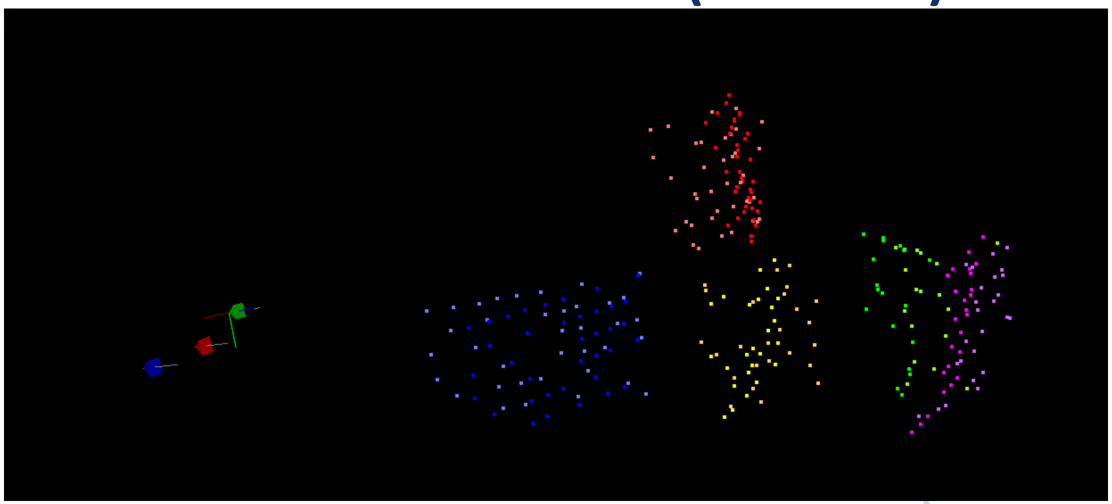
Stereo capture → Checkerboard Corners Detection → Stereo Triangulation → Horn's minimization algorithm in a RANSAC process

ORF capture → Checkerboard Corners Detection → ORF Pinhole Inversion → Horn's minimization algorithm in a RANSAC process

AEROASTRO

Calibration Algorithm

HALO Calibration (extrinsic):



The minimization process is:

$$\boldsymbol{M_{LT}} = \boldsymbol{arg\ min} \left\{ \sum_{i=0}^{m} \left\| \boldsymbol{P_T^i} - \boldsymbol{\widehat{M}_{LT}} * \boldsymbol{P_L^i} \right\|^2 \right\}$$

Stereo capture

Checkerboard Corners Detection

Stereo Triangulation

ORF capture

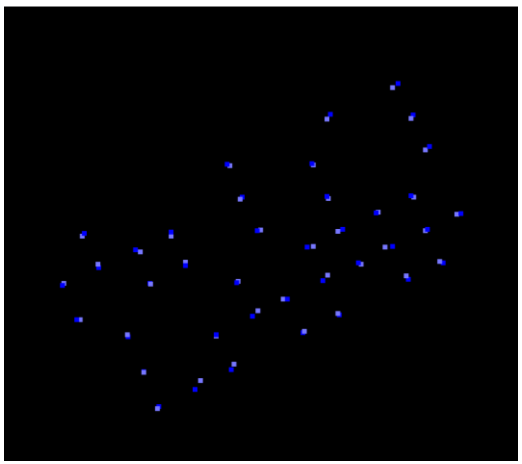
Checkerboard Corners Detection

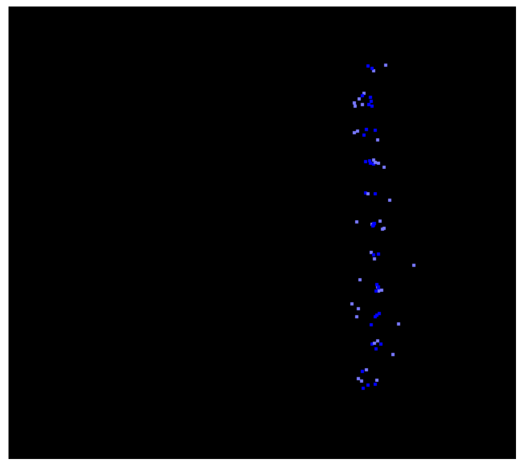
ORF Pinhole Inversion

Horn's minimization algorithm in a RANSAC process

Calibration Algorithm

HALO Calibration (extrinsic):





The minimization process is:

$$M_{LT} = \arg\min \left\{ \sum_{i=0}^{m} \left\| P_T^i - \widehat{M}_{LT} * P_L^i \right\|^2 \right\}$$

Stereo capture → Checkerboard Corners Detection → Stereo Triangulation

ORF capture → Checkerboard Corners Detection → ORF Pinhole Inversion

→ Horn's minimization algorithm in a RANSAC process

AEROASTRO

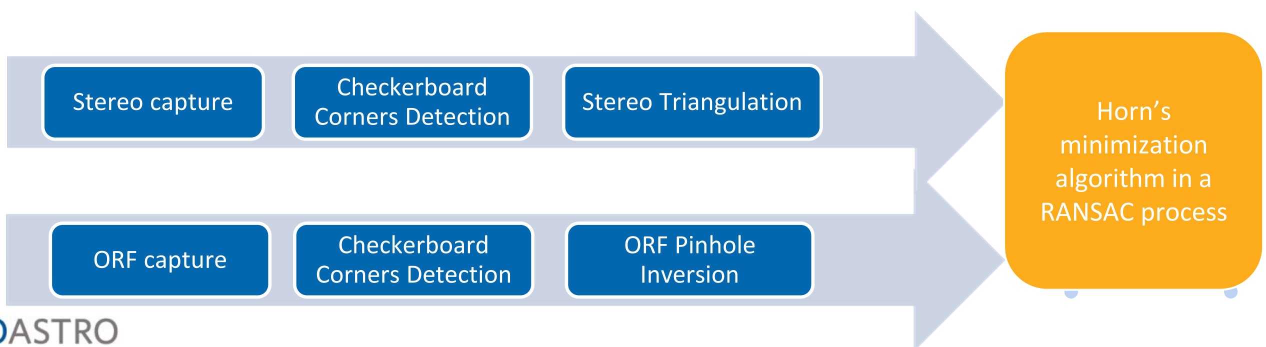
Calibration Algorithm

HALO Calibration (extrinsic):

Translation axis	CAD Model	Calib with RANSAC	Calib without RANSAC
X	$16.09\ cm$	$16.6\ cm$	$14\ cm$
Y	$2.01\ cm$	$0.77\ cm$	$-0.67\ cm$
Z	$4.74\ cm$	$5.89\ cm$	$8.46\ cm$

The minimization process is:

$$M_{LT} = \arg\min \left\{ \sum_{i=0}^{m} \left\| P_T^i - M_{LT} * P_L^i \right\|^2 \right\}$$



AEROASTRO

Overview

- Introduction
- Sensors Features and Acquisition
  - Time-of-Flight Camera (Optical Range Finder)
  - Stereoscopic Cameras (VERTIGO)
  - Thermocam
- Calibration Algorithm
  - ORF calibration
  - Stereo calibration
  - System calibration
- Fusion Algorithm
- Conclusions and Perspectives

Fusion Algorithm

Goals of multi-sensor data fusion:

For this algorithm:

- **Increase *accuracy***
- **Increase *certainty***
- **Increase *completeness***
- **Increase *representation***

Fusion Algorithm

Goals of multi-sensor data fusion:

For a future algorithm:

- **Increase *accuracy***
- **Increase *certainty***
- **Increase *completeness***
- **Increase *representation***

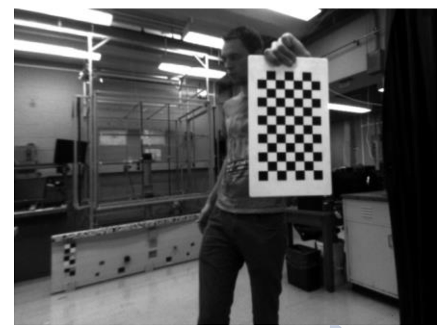
Fusion Algorithm


ORF Depth


ORF Confidence


Stereo Left

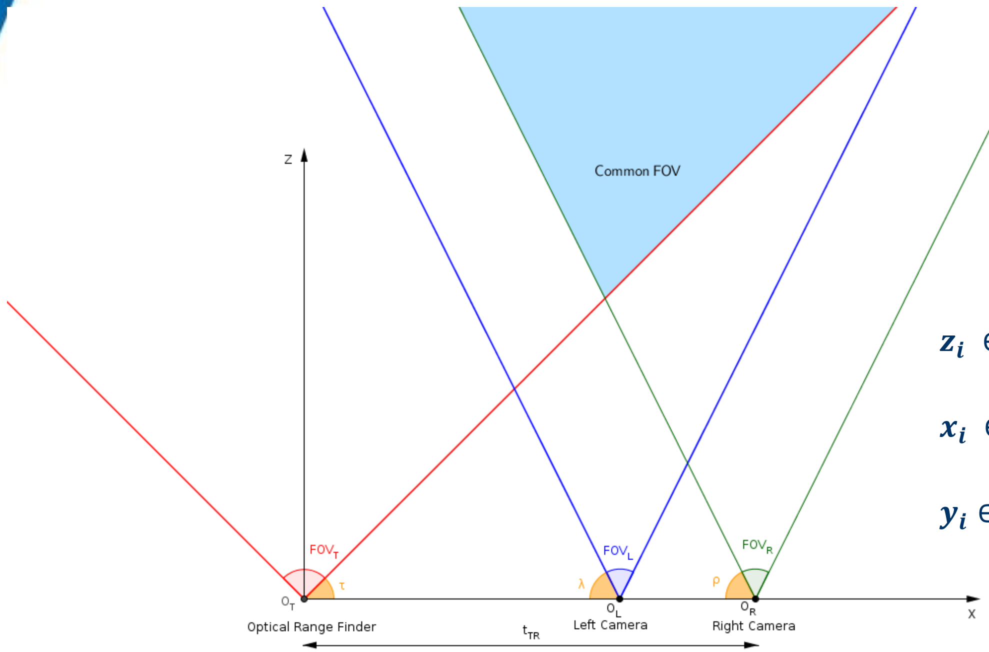

ORF Visual


Stereo Right

- Acquisition
- Determination of Points Framed by all Sensors
- ORF Noise Model
- Noise Interval Discretization
- Stereo projection
- Probability maximization

AEROASTRO

Fusion Algorithm



$$z_i \in \left[min\left(\frac{t_{TR} \tan(\rho)\tan(\tau)}{\tan(\rho)+\tan(\tau)}, range_{min} \right); range_{max} \right]$$

$$x_i \in \left[\frac{-z_i - t_{TR}\tan(\rho)}{\tan(\rho)}; \frac{z_i}{\tan(\tau)} \right]$$

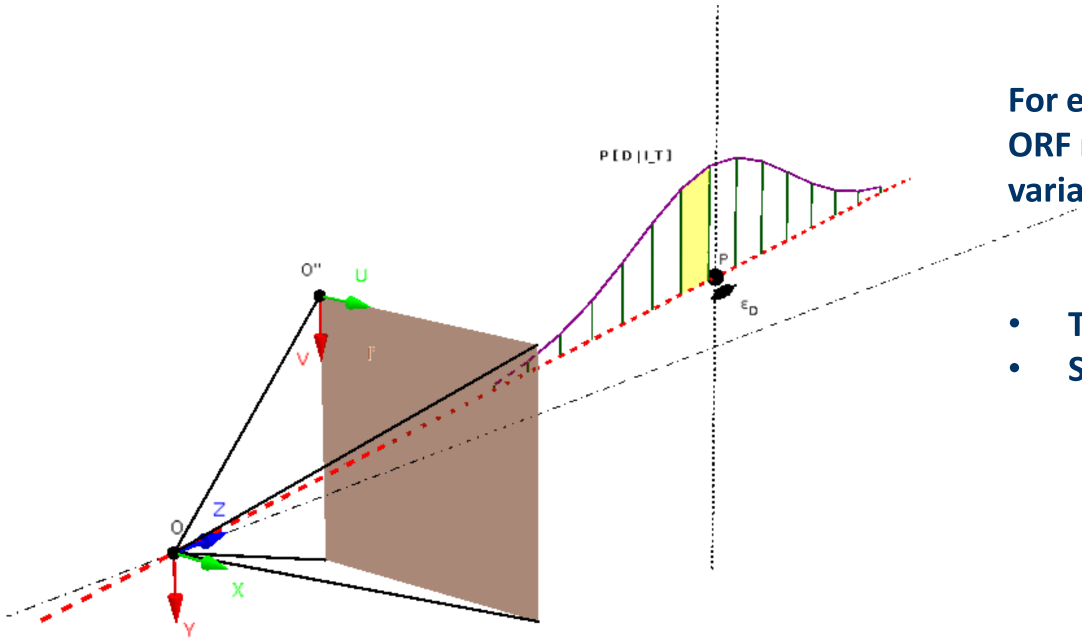
$$y_i \in \left[min\left(\frac{-z_i}{\tan(\rho)}, \frac{-z_i}{\tan(\tau)} \right); min\left(\frac{-z_i}{\tan(\rho)}, \frac{-z_i}{\tan(\tau)} \right) \right]$$

Acquisition → **Determination of Points Framed by all Sensors** → ORF Noise Model → Noise Interval Discretization → Stereo projection → Probability maximization

AEROASTRO

MIT SSL

Fusion Algorithm



For each 3D point constructed by the ORF, ORF noise can be reduce to a Normal with variance σ_w including:

- **Thermal noise with variance σ_t**
- **Scattering noise with variance σ_s**

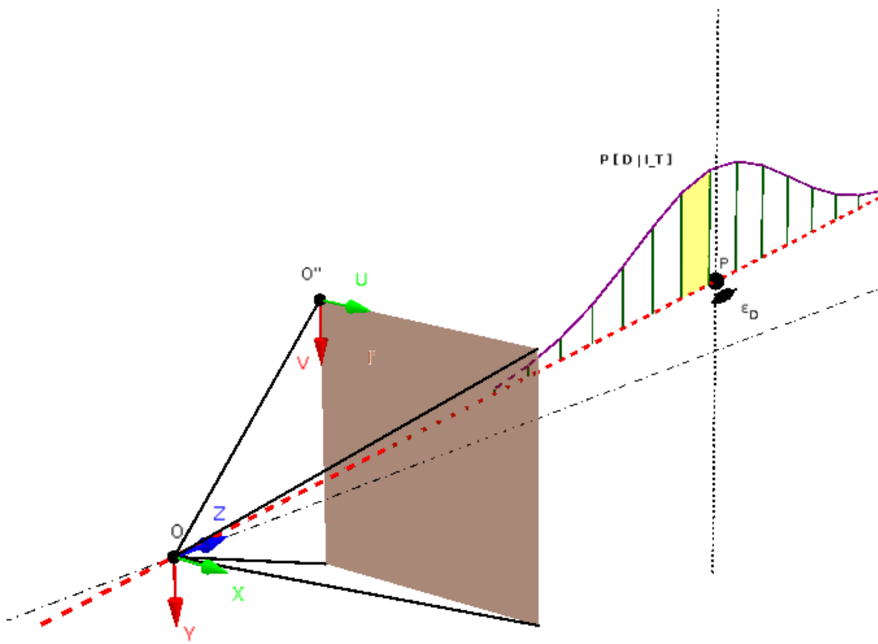
Acquisition | Determination of Points Framed by all Sensors | ORF Noise Model | Noise Interval Discretization | Stereo projection | Probability maximization

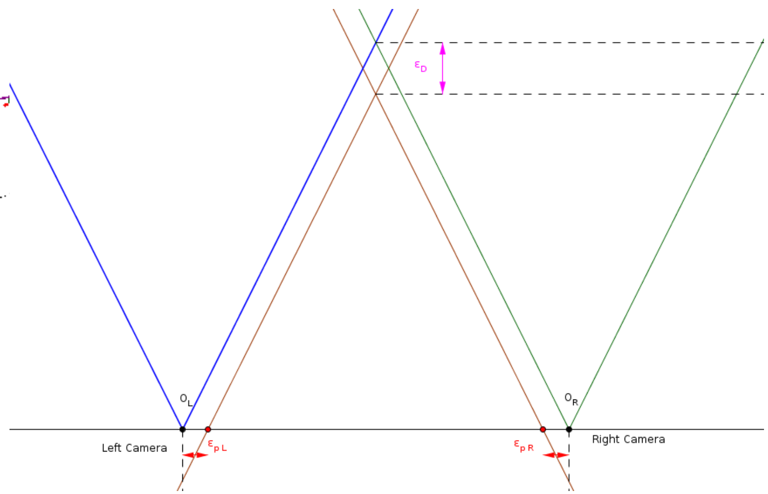
AEROASTRO

MIT SSL

Fusion Algorithm

The interval $[d_i - 3\sigma_w; d_i + 3\sigma_w]$ can be split in m steps of size $\epsilon_d = \frac{d^2}{b\,f}\epsilon_p$

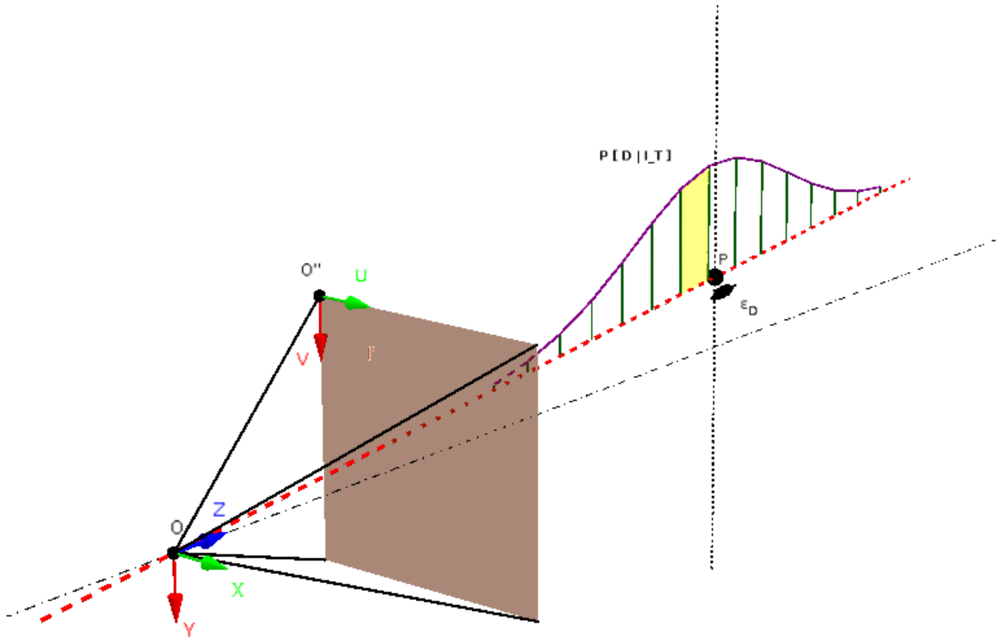


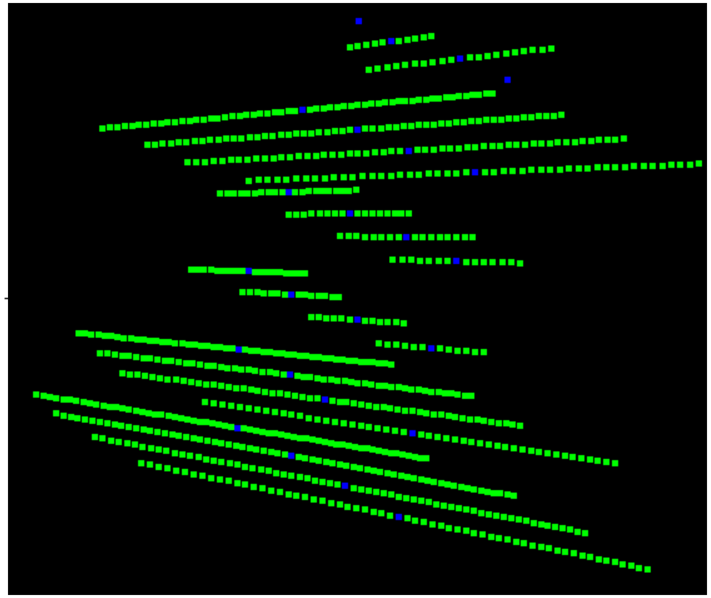


Acquisition → Determination of Points Framed by all Sensors → ORF Noise Model → **Noise Interval Discretization** → Stereo projection → Probability maximization

AEROASTRO

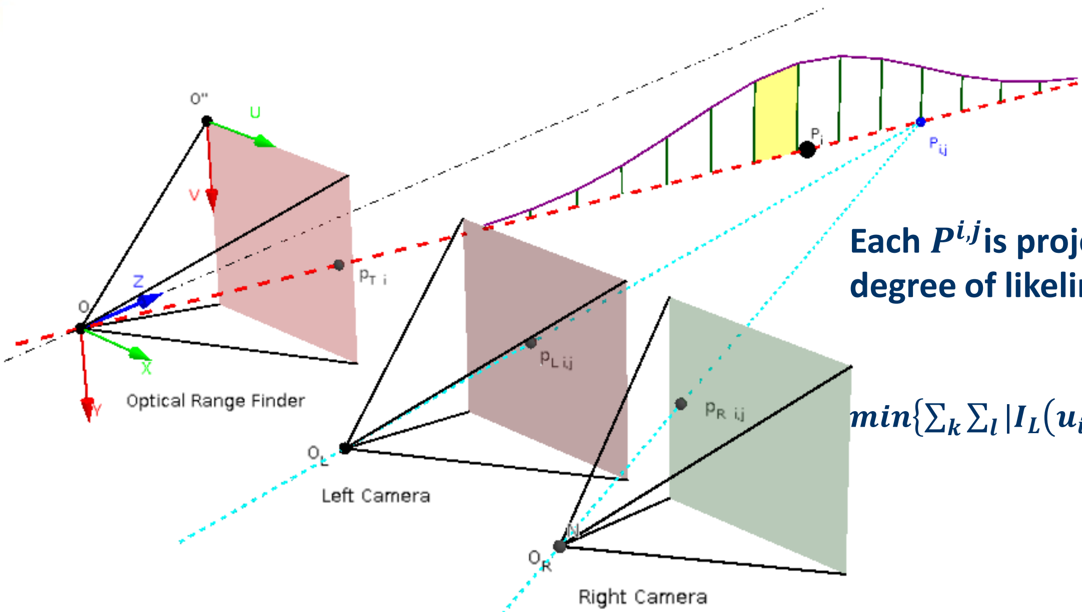
Fusion Algorithm





Acquisition → Determination of Points Framed by all Sensors → ORF Noise Model → Noise Interval Discretization → Stereo projection → Probability maximization

Fusion Algorithm

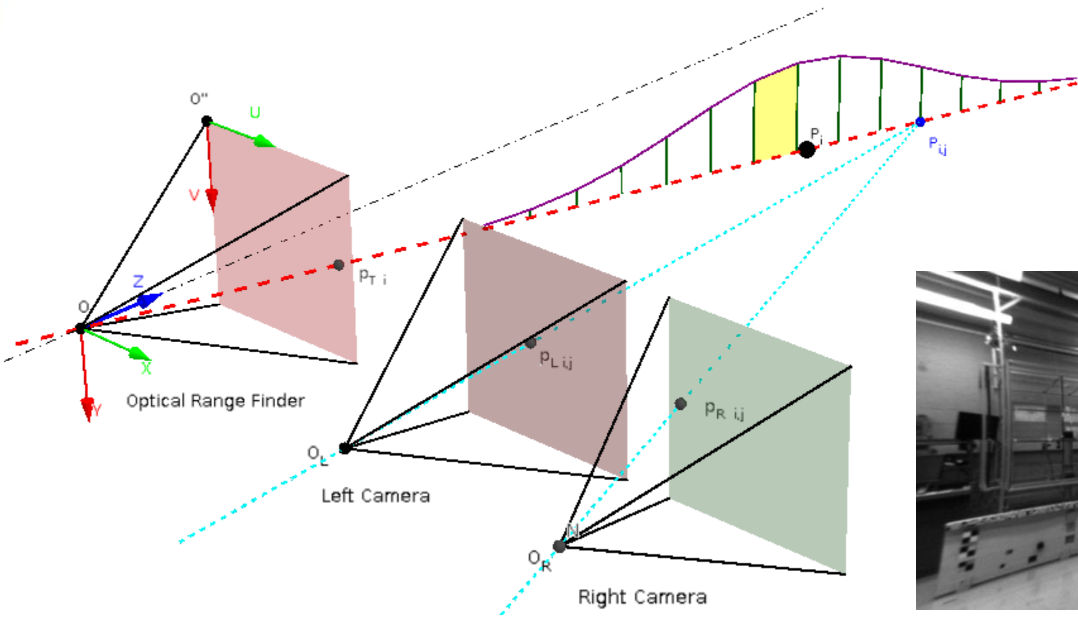


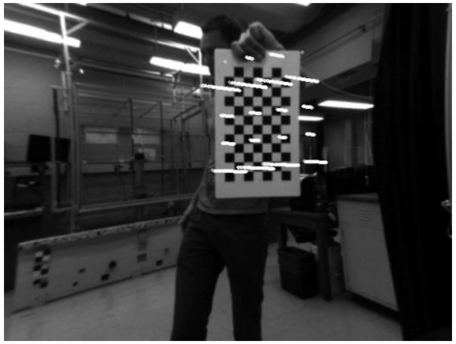
Each $P^{i,j}$ is projected in left and right images and a degree of likeliness is computed around each points:

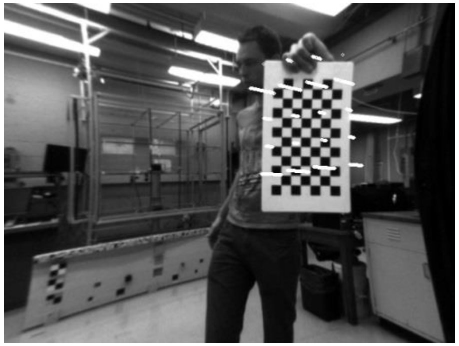
$$c_{i,j} = min\{\sum_k \sum_l |I_L(u_{i,j}+k, v_{i,j}+l) - I_L(u_{i,j}+k, v_{i,j}+l)|, thresh\}$$

Acquisition → Determination of Points Framed by all Sensors → ORF Noise Model → Noise Interval Discretization → **Stereo projection** → Probability maximization

Fusion Algorithm







Acquisition → Determination of Points Framed by all Sensors → ORF Noise Model → Noise Interval Discretization → **Stereo projection** → Probability maximization

Fusion Algorithm

Probability is computed as:

$$P\left[d^i = d^{i,j}\right] = P\left[d^i = d^{i,j} \middle| I_T\right] * P\left[d^i = d^{i,j} \middle| I_L I_R\right]$$

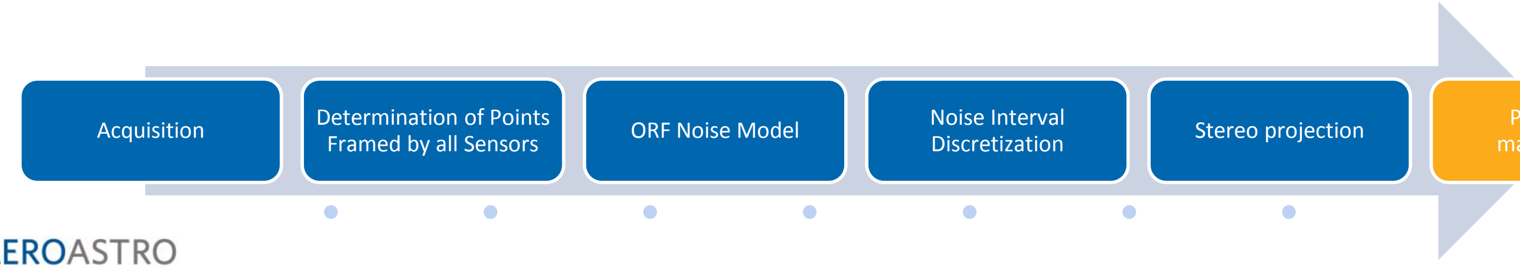
Where:

$$P\left[d^i = d^{i,j} \middle| I_T\right] = \frac{1}{\sigma_w^i \sqrt{2\pi}} e^{-\frac{\left(d^{i,j} - d^i\right)^2}{2\, {\sigma_w^i}^2}}$$

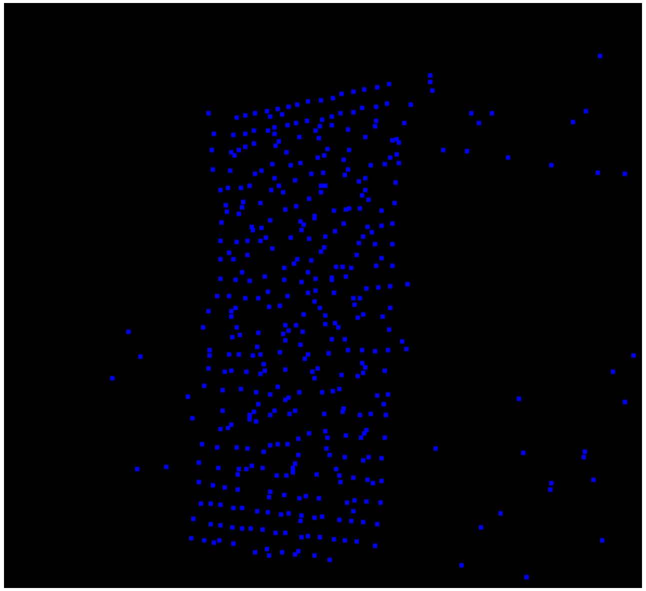
$$P\left[d^i = d^{i,j} \middle| I_L I_R\right] = e^{\frac{-c_{i,j}}{\sigma_n^i}}$$

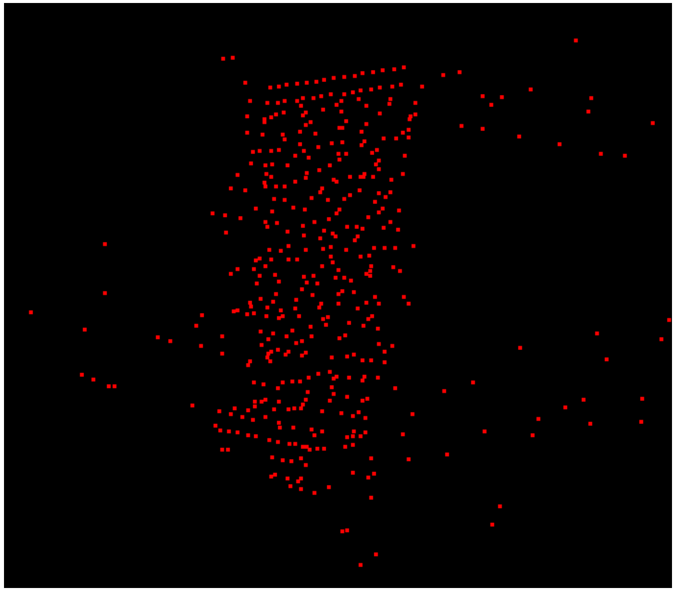
And finally,

$$\hat{d}^i = max_j \left\{P\left[d^i = d^{i,j}\right]\right\}$$



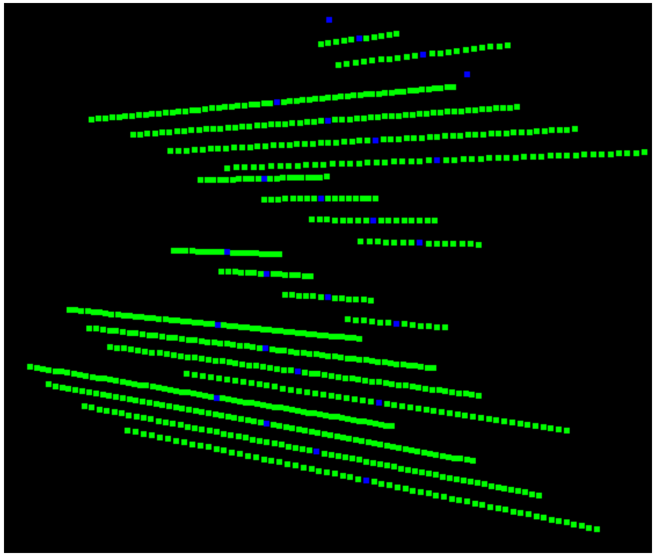
Fusion Algorithm

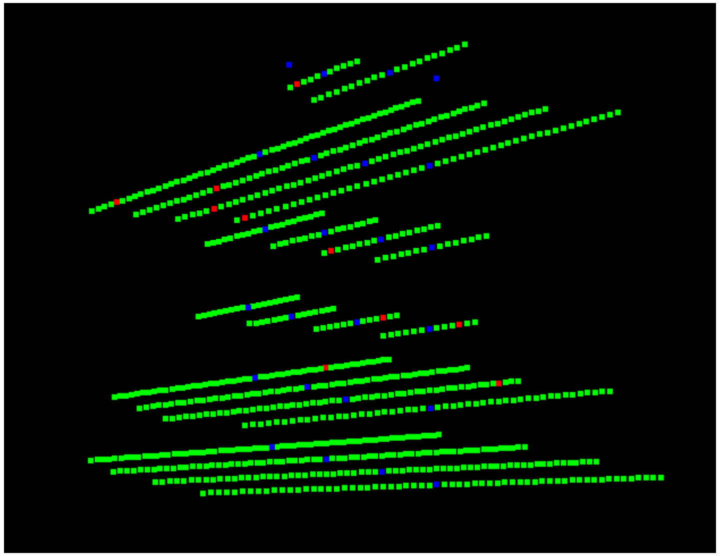




In the first results with INSPECT, fusion (right) does not improve accuracy of the ORF cloud (left)

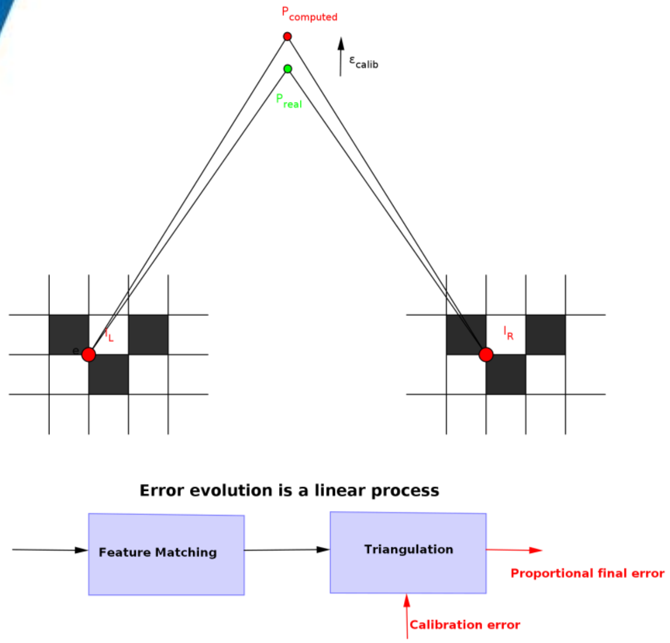
Fusion Algorithm

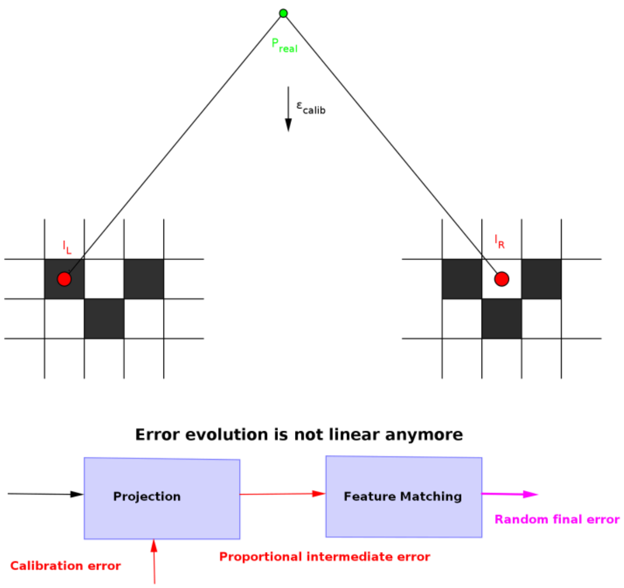




In the first results with INSPECT, fusion (right) does not improve accuracy of the ORF cloud (left)

Fusion Algorithm





Errors seem to come essentially from stereo calibration matrices accuracy and non-linearity of the process

Overview

- Introduction
- Sensors Features and Acquisition
  - Time-of-Flight Camera (Optical Range Finder)
  - Stereoscopic Cameras (VERTIGO)
  - Thermocam
- Calibration Algorithm
  - ORF calibration
  - Stereo calibration
  - System calibration
- Fusion Algorithm
- Conclusions and Perspectives

Conclusions and Future Perspectives

- **Acquisition seems to give acceptable results on the ground and during RGA flights.**
- **The calibration algorithm gives good results but could be improved by:**
  - **Adding the thermocam by using a cold/hot checkerboard pattern**
  - **Using better stereo calibration parameters during system calibration**
- **The fusion algorithm does not increase accuracy for the moment (on the ground and during RGA flights) and will not for sure increase completeness and certainty:**
  - **A way to improve it would be a better ORF noise computation or stereo likeliness determination**
  - **The thermocam should be added to create a second point cloud**
  - **Other algorithms have been found that are more adapted to SLAM, avoid the stereo re-projection and consider full objects instead of separated points**

AEROASTRO