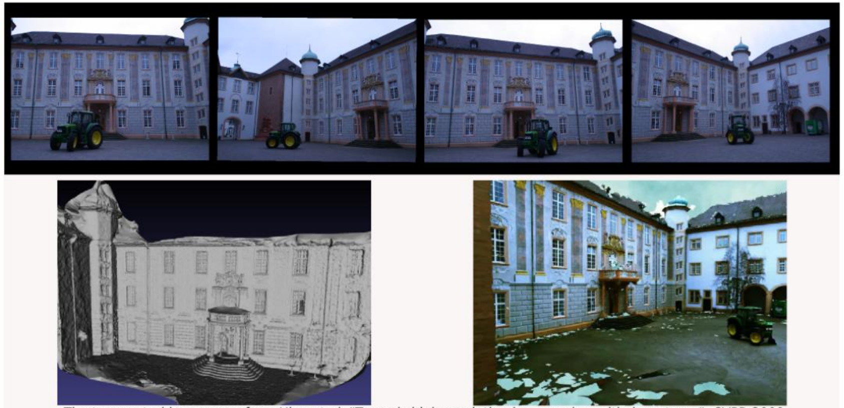
3D Vision

Group members:

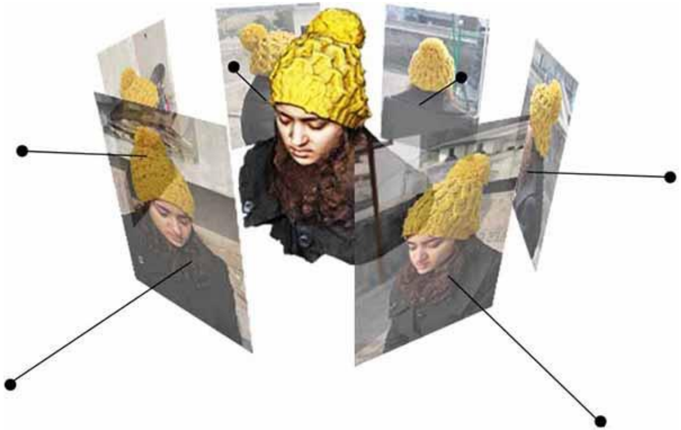
Alex Martín, Oriol Catalan Yarza, Jose Manuel López, Marcos V. Conde

AIM

3D reconstruction from N non calibrated cameras.

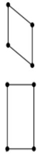
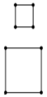
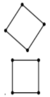


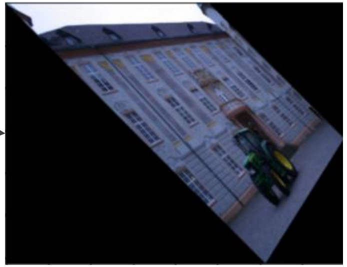
The two central images are from Hiep et al. "Towards high-resolution large-scale multi-view stereo", CVPR 2009



INDEX

Image transformations and rectification

Group	Matrix	Distortion	Invariant properties
Projective 8 dof	$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$		Concurrency, collinearity, **order of contact**: intersection (1 pt contact); tangency (2 pt contact); inflections (3 pt contact with line); tangent discontinuities and cusps. cross ratio (ratio of ratio of lengths).
Affine 6 dof	$\begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$		Parallelism, ratio of areas, ratio of lengths on collinear or parallel lines (e.g. midpoints), linear combinations of vectors (e.g. centroids). The line at infinity, $\mathbf{l}_\infty$.
Similarity 4 dof	$\begin{bmatrix} sr_{11} & sr_{12} & t_x \\ sr_{21} & sr_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$		Ratio of lengths, angle. The circular points, **I**, **J** (see section 2.7.3).
Euclidean 3 dof	$\begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$		Length, area



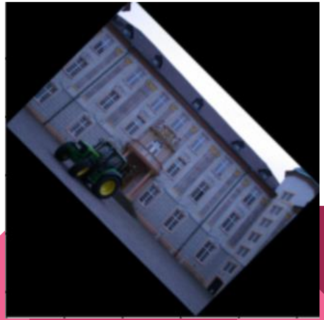


Image transformations and rectification

A projective transformation can be decomposed in:

$$\mathrm{H} = \mathrm{H_P}\,\mathrm{H_A}\,\mathrm{H_S} = \begin{bmatrix} \mathrm{I} & \mathbf{0} \\ \mathbf{v}^\mathsf{T} & 1 \end{bmatrix} \begin{bmatrix} \mathrm{K} & \mathbf{0} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix} \begin{bmatrix} s\mathrm{R} & \mathbf{t} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix}$$

H_P (2 dof) moves the line at infinity

H_A (2 dof) affects the affine properties, but does not move the line at infinity.

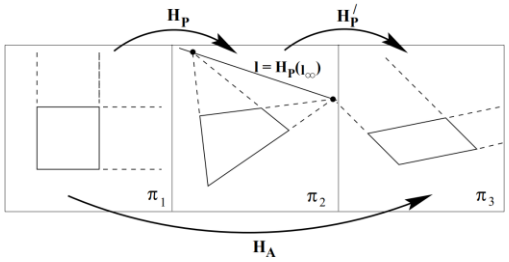
H_S (4 dof) which does not modify the affine or projective properties.

Image transformations and rectification

Affine rectification

For 2D:

The goal is to map l back to the vanishing line in its canonical position $l_\infty = (0, 0, 1)^T$. So we have to apply a homography to the image that satisfies the following:



$$H^{-T}(l_1, l_2, l_3)^T = (0, 0, 1)^T$$

$$H = H_A \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{bmatrix}$$

Image transformations and rectification

Affine rectification

For 3D:

The goal is to map $\boldsymbol{\pi}$ back to the plane at infinity in its canonical position $\boldsymbol{\pi} = (0, 0, 1)^T$.
So we have to apply a homography to the image that satisfies the following:

$H^{-T}\boldsymbol{\pi} = (0, 0, 0, 1)^T$

$$H = \begin{bmatrix} I \mid \mathbf{0} \\ \boldsymbol{\pi}^T \end{bmatrix}$$

Image transformations and rectification

Metric rectification

<u>For 2D:</u>

In the same way that the affine rectification, we can perform a metric rectification by transforming the **circular points, I, J** to their canonical position using **conic dual to the circular points, C_∞^***, which enables both the **projective and affine** components of a projective transformation to be determined. From the SVD of C_∞^* we can obtain the **rectifying homography U**.

$$C_\infty^{*}{}' = HC^*H^T = \begin{bmatrix} KK^T & KK^T\mathbf{v} \\ \mathbf{v}^TKK^T & \mathbf{v}^TKK^T\mathbf{v} \end{bmatrix}$$

$$C_\infty^{*}{}' = U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} U^T$$

Image transformations and rectification

Metric rectification

For 2D:

If the image was **affinely rectified**, we require two constraints to specify the 2 degrees of freedom left, 2 orthogonal lines:

$$\mathbf{l}'^{\mathsf{T}} C^{*\prime}_{\infty} \mathbf{m}' = 0 \qquad \begin{pmatrix} l'_1 & l'_2 & l'_3 \end{pmatrix} \begin{bmatrix} \mathrm{KK}^{\mathsf{T}} & \mathbf{0} \\ \mathbf{0}^{\mathsf{T}} & 0 \end{bmatrix} \begin{pmatrix} m'_1 \\ m'_2 \\ m'_3 \end{pmatrix} = 0 \qquad (l'_1 m'_1, l'_1 m'_2 + l'_2 m'_1, l'_2 m'_2)\, \mathbf{s} = 0,$$

If we have the original perspective :

$$(l_1 m_1, (l_1 m_2 + l_2 m_1)/2, l_2 m_2, (l_1 m_3 + l_3 m_1)/2, (l_2 m_3 + l_3 m_2)/2, l_3 m_3)\, \mathbf{c} = 0$$

c, and hence **C_∞***, is obtained as the null vector.

Image transformations and rectification

Metric rectification

<u>For 3D:</u>

In this case it will be used the **image of the absolute conic w** instead of C_∞^* which it can be obtained by solving $A\omega v = \mathbf{0}$ which arranges the constraints imposed to **w.**

$$A = \begin{pmatrix} u_1 v_1 & u_1 v_2 + u_2 v_1 & u_1 v_3 + u_3 v_1 & u_2 v_2 & u_2 v_3 + u_3 v_2 & u_3 v_3 \\ u_1 z_1 & u_1 z_2 + u_2 z_1 & u_1 z_3 + u_3 z_1 & u_2 z_2 & u_2 z_3 + u_3 z_2 & u_3 z_3 \\ v_1 z_1 & v_1 z_2 + v_2 z_1 & v_1 z_3 + v_3 z_1 & v_2 z_2 & v_2 z_3 + v_3 z_2 & v_3 z_3 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}$$

Image transformations and rectification

Metric rectification

For 3D:

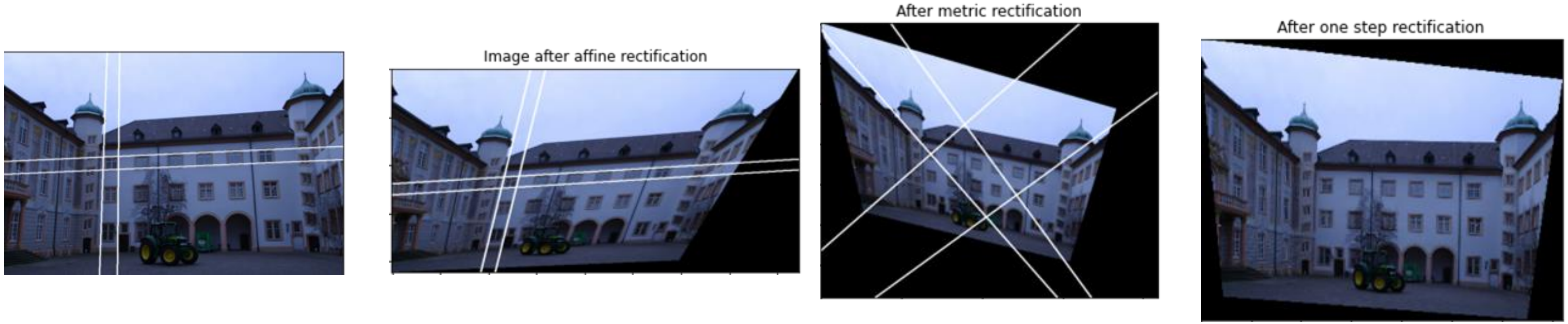
Having **w** and also the camera matrix $P = [M|m]$ we can obtain the homography

$$H_{e \leftarrow a} = \begin{pmatrix} A^{-1} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix}$$

from

$$AA^T = (M^T \omega M)^{-1}$$

Image transformations and rectification



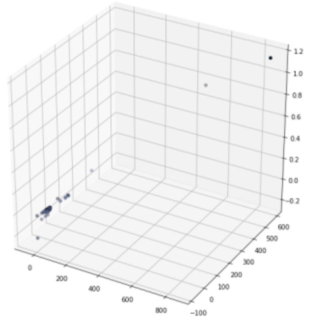
The proper application of the matrix H is fundamental to perform any transformation.

We try to remove projective and affine distortion from images through means of two different rectification method (stratified, single-step). Both methods are sensitive to choices.

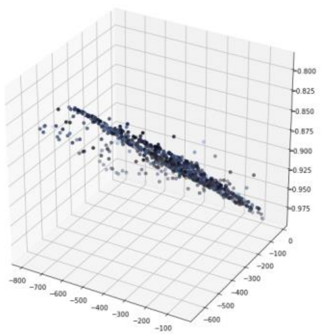
The main drawback of the applied method is the requirement of having parallel and orthogonal lines, previously detected/annotated.

Image transformations and rectification

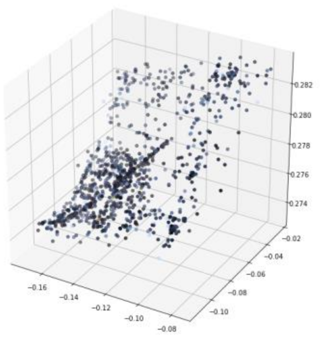
Results: Lab 5



3D points after triangulation



3D points after affine reconstruction



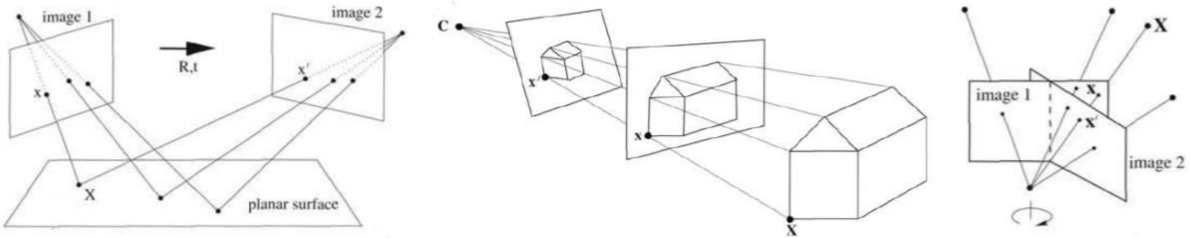
3D points after metric reconstruction

The results are hard to interpret with this plotting and are reliable on the quality of the detected keypoints.

Homography estimation and applications

A homography relates 2 images if:

- Both belong to the **same plane** in the real world.
- Both were obtained by just **rotating** the **camera** around its **own axis**.
- Both were obtained by just **modifying** the **focal length**.
- Both are taken from enough **distance** to be considered **infinite** (can be seen as being on the **same plane**).



Homography estimation and applications

In **Lab 2** we explored **homography estimation** with different methods:

- Robust Normalized DLT (algebraic method).
- Gold-Standard algorithm (geometric method).

Additionally we reviewed some homography **applications**:

- Image Mosaics.
- Logo detection/insertion.

Homography estimation and applications

Robust Normalized DLT (Direct Linear Transformation)

Given a sufficient set of point **correspondences**, estimate homography matrix *H.*

Relationship between corresponding points is: $c \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = H \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$ where $H = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{pmatrix}$

Can be rewritten in matrix form: $A_i = \begin{pmatrix} -x & -y & -1 & 0 & 0 & 0 & ux & uy & u \\ 0 & 0 & 0 & -x & -y & -1 & vx & vy & v \end{pmatrix}$ $\vec{h} = (h_1\ h_2\ h_3\ h_4\ h_5\ h_6\ h_7\ h_8\ h_9)^T$

That can be estimated: $\text{minimize}\|A\vec{h}\| \quad s.t.\ \|h\| = 1$ with solution eigenvector from $A^T A$ with the smallest eigenvalue.

Robustness comes from RANSAC: compute inliers for scoring candidate models based on a certain threshold.

Homography estimation and applications

The Golden Standard Algorithm

Refine estimated homography by minimizing the geometric (reprojection) cost function.

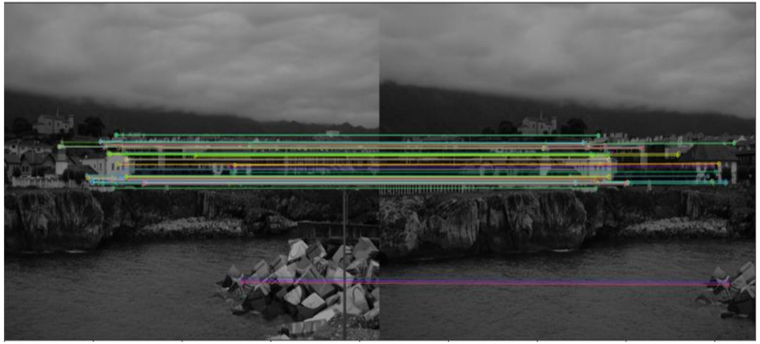
$$\sum_i d(\mathbf{x}_i, \hat{\mathbf{x}}_i)^2 + d(\mathbf{x}'_i, \hat{\mathbf{x}'}_i)^2 \quad \text{where } \hat{\mathbf{x}'}_i = \hat{H}_i \hat{\mathbf{x}}_i$$

Then, using keypoints $\mathbf{x_i}$, we compute an estimation of the variables $\hat{\mathbf{x}}_\mathbf{i}$ and minimize the cost over $\hat{\mathbf{x}}_\mathbf{i}$ and $\hat{H}$

Homography estimation and applications

Image Mosaic

We can build mosaic from images if we estimate the refined homographies



Llanes samples. These are the best results because images were taken from a big distance (infinite) so they can be seen as belonging to the same plane.

Homography estimation and applications

Image mosaic for the castle set.
Required some parameter tuning (RANSAC).

Image mosaic for the aerial/site13.

Homography estimation and applications

Logo Detection/Insertion

We first detect keypoints between image and logo and then RANSAC is applied to ensure the usage of inliers. Then we use the estimated homography to insert the logo.

Fundamental matrix estimation and applications

Properties

- Relates corresponding points between uncalibrated cameras $\mathbf{x}'^{\mathsf{T}}\mathbf{F}\mathbf{x} = 0$
- **F** is singular (rank 2)
- Defined up-to-scale

Applications

- Photo-sequencing
- Camera matrix recovery

Fundamental matrix estimation and applications

Estimation: 8-point algorithm

$$\begin{bmatrix} u' & v' & 1 \end{bmatrix} \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} u_1u_1' & v_1u_1' & u_1' & u_1v_1' & v_1v_1' & v_1' & u_1 & v_1 & 1 \\ u_2u_2' & v_2u_2' & u_2' & u_2v_2' & v_2v_2' & v_2' & u_2 & v_2 & 1 \\ u_3u_3' & v_3u_3' & u_3' & u_3v_3' & v_3v_3' & v_3' & u_3 & v_3 & 1 \\ u_4u_4' & v_4u_4' & u_4' & u_4v_4' & v_4v_4' & v_4' & u_4 & v_4 & 1 \\ u_5u_5' & v_5u_5' & u_5' & u_5v_5' & v_5v_5' & v_5' & u_5 & v_5 & 1 \\ u_6u_6' & v_6u_6' & u_6' & u_6v_6' & v_6v_6' & v_6' & u_6 & v_6 & 1 \\ u_7u_7' & v_7u_7' & u_7' & u_7v_7' & v_7v_7' & v_7' & u_7 & v_7 & 1 \\ u_8u_8' & v_8u_8' & u_8' & u_8v_8' & v_8v_8' & v_8' & u_8 & v_8 & 1 \end{bmatrix} \begin{bmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{bmatrix} = 0 \rightarrow Wf = 0$$

we minimize $\|W \cdot f\|$ under the constraint $\|f\| = 1$.

Fundamental matrix estimation and applications

Application: Photo sequencing

F has been estimated, we can use it to compute the epipolar lines.

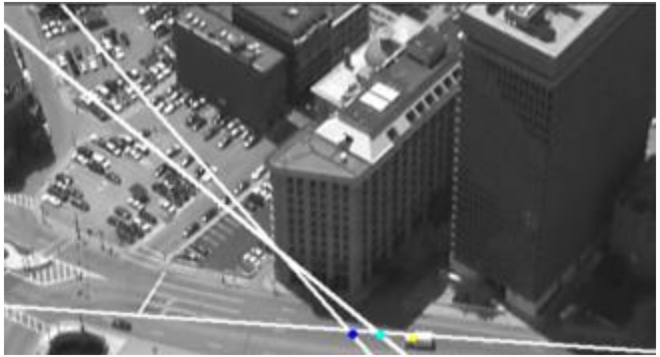
$$ep_2 = F_{12}^T \mathbf{p_2}$$

$$ep_3 = F_{13}^T \mathbf{p_3}$$

1. Find the trajectory line.
2. As we assume that the van only moves following a straight line, we just need to find the intersection between those epipolar lines and the line trajectory by taking its cross product

Fundamental matrix estimation and applications

From top to bottom: first, second and third frames



Locations of the van in the second and third frames displayed in the first one (blue points)

Essential matrix estimation and applications

Application: Camera matrix recovery

In this case, first we have to obtain the camera matrices, which can be retrieved from **F** up to a **projective transformation of 3-space** and from the **E** up to scale and a four-fold ambiguity.

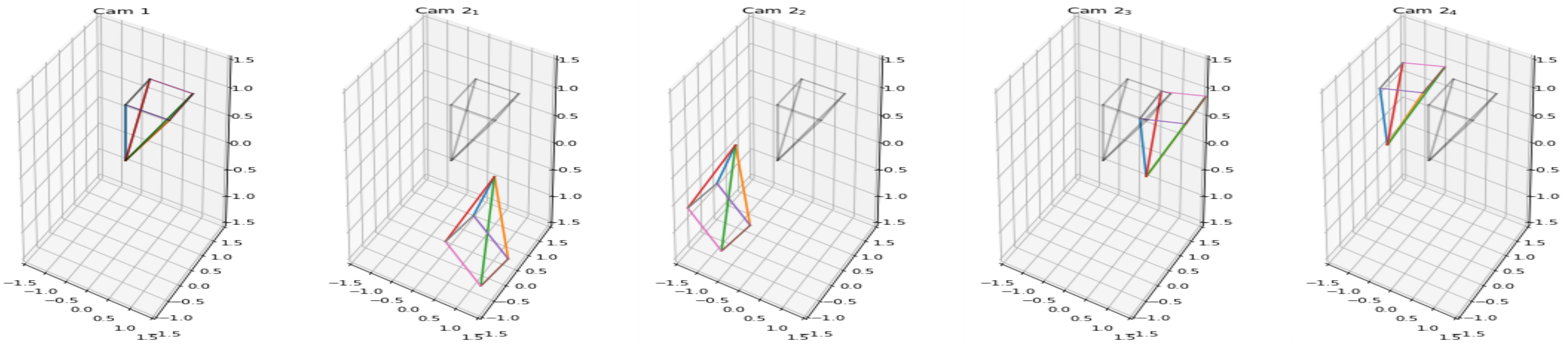
If we know the internal parameters, we can:

1. Compute the essential matrix $\mathbf{E} = (\mathbf{K}')^\top \ \mathbf{F} \ \mathbf{K}$
2. Assume P = [I|0] and estimate the other camera matrix with 4 different options (2 due to T and 2 due to R).

$$\mathtt{P}' = [\mathtt{UWV}^\mathsf{T} \mid +\mathbf{u}_3] \ \text{ or } \ [\mathtt{UWV}^\mathsf{T} \mid -\mathbf{u}_3] \ \text{ or } \ [\mathtt{UW}^\mathsf{T}\mathtt{V}^\mathsf{T} \mid +\mathbf{u}_3] \ \text{ or } \ [\mathtt{UW}^\mathsf{T}\mathtt{V}^\mathsf{T} \mid -\mathbf{u}_3] \qquad \mathtt{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Essential matrix estimation and applications

Application: Camera matrix recovery



Plots of the camera matrices recovered

Essential matrix estimation and applications

Application: Triangulation

Once we have the camera matrices we can perform triangulation and extract the 3D position **X** from the coordinates of two images with an analogous method to the DLT.

$$\mathtt{A} = \begin{bmatrix} x\mathbf{p}^{3\mathsf{T}} - \mathbf{p}^{1\mathsf{T}} \\ y\mathbf{p}^{3\mathsf{T}} - \mathbf{p}^{2\mathsf{T}} \\ x'\mathbf{p}'^{3\mathsf{T}} - \mathbf{p}'^{1\mathsf{T}} \\ y'\mathbf{p}'^{3\mathsf{T}} - \mathbf{p}'^{2\mathsf{T}} \end{bmatrix}$$

$$\min_{\mathbf{X}} \|\mathbf{AX}\|_2$$

$$\text{such that } \|\mathbf{X}\|_2 = 1$$

Essential matrix estimation and applications

Application: Camera matrix recovery

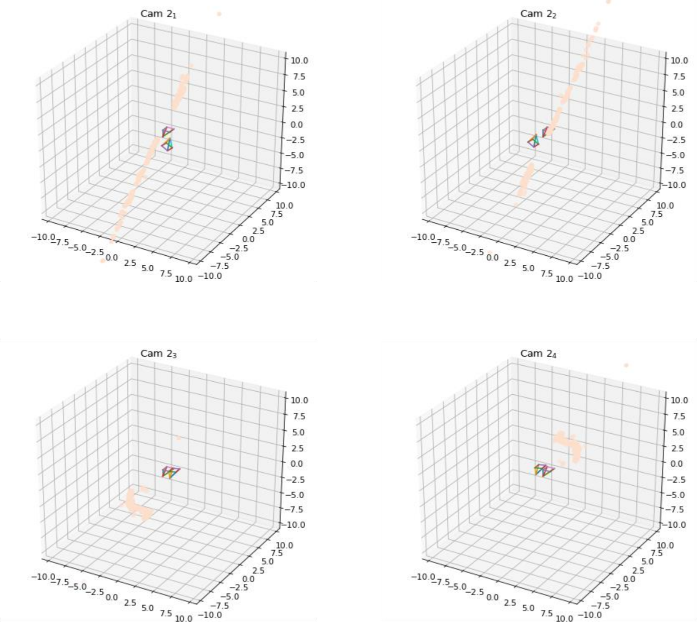
For the uncalibrated case we will extract the camera matrices directly from F, assuming the first one also to be P = [I|0] and the second one by computing the epipole **e’** as we know, $\mathbf{e}'\mathrm{F} = \mathbf{0}$

$$\mathrm{P}' = [[\mathbf{e}']_{\times}\mathrm{F} + \mathbf{e}'\mathbf{v}^{\mathsf{T}} \mid \lambda\mathbf{e}']$$

Which then can be used in the structure from motion pipeline.

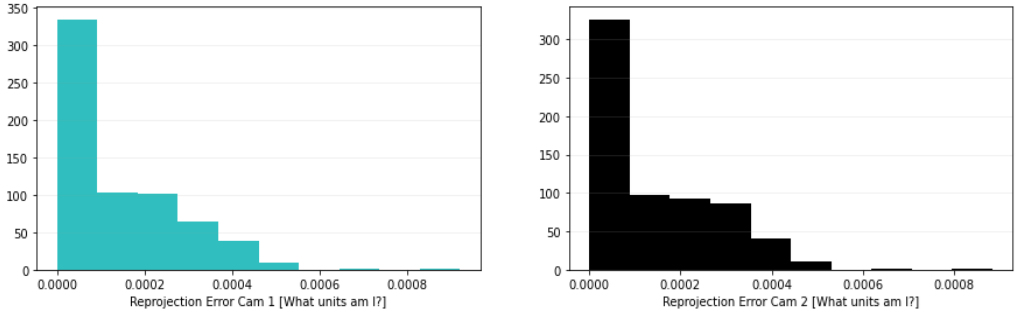
Essential matrix estimation and applications

Application: Triangulation



Plots of the camera matrices with the triangulated points

```
Mean reprojective error for cam1 = 0.02967202539504447
Mean reprojective error for cam2 = 0.03059344645817963
```



Reprojection error between 2 cameras

Depth map computation with local methods

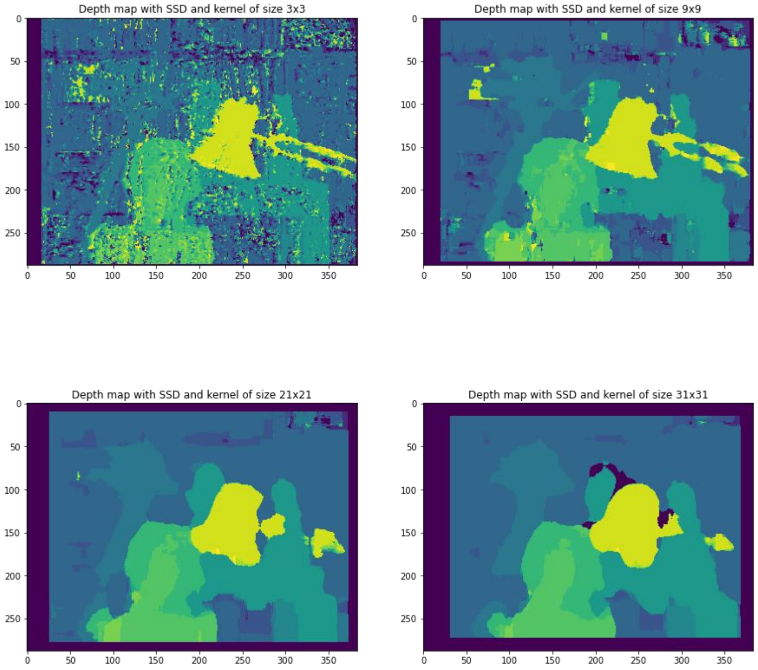
- We build a local method for stereo matching and apply it to a pair of rectified images.
- For each pixel from the first image, we create a sliding window along the same line in the second image and compare the content to the reference window.
- Pixel with minimum matching cost is selected.
- The differences in the X coordinate from the winner and the original pixel correspond to the disparity.
- 2 matching cost functions:
  - Sum of Squared Differences (SSD)
  - Normalized Cross Correlation (NCC): robust to different light conditions

$$SSD(p,d) = \sum_{q \in N_p} w(p,q)(I_1(q) - I_2(q+d))^2$$

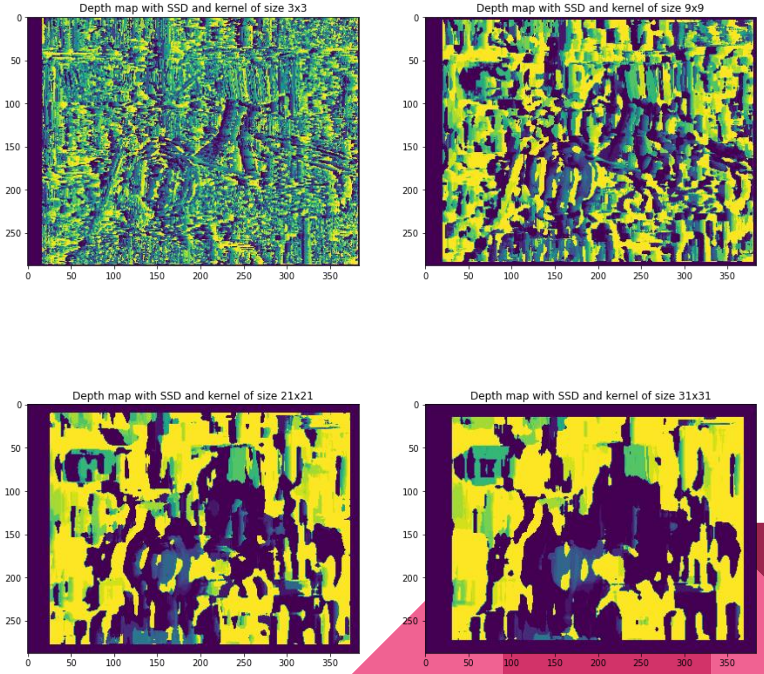
$$NCC(p,d) = \frac{\sum_{q \in N_p} w(p,q)(I_1(q) - \hat{I}_1)(I_2(q+d) - \hat{I}_2)}{\sigma_{I_1}\sigma_{I_2}}$$

Depth map computation with local methods

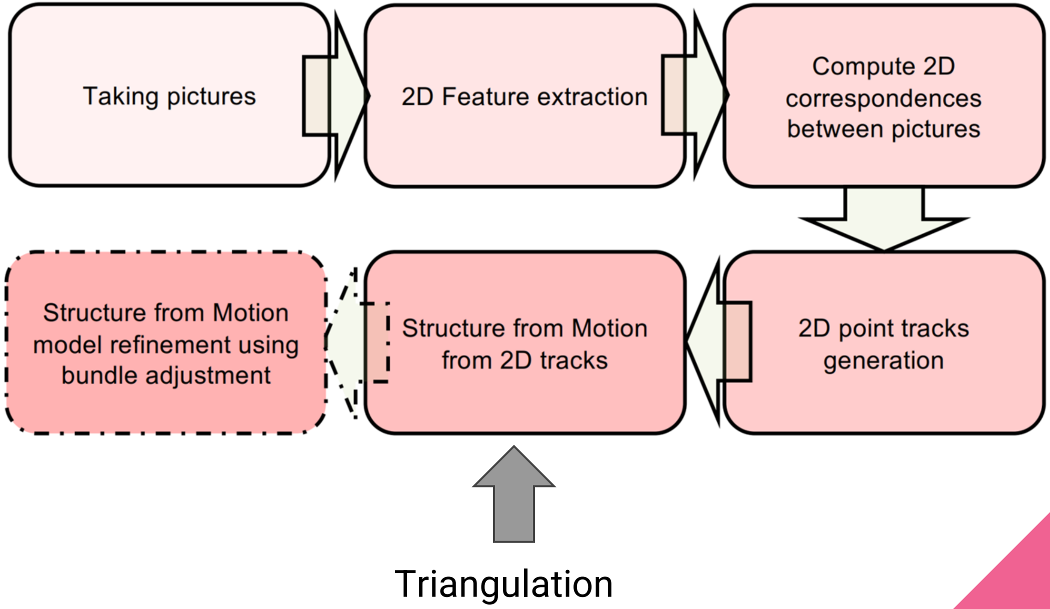
Results with SSD



Results with NCC



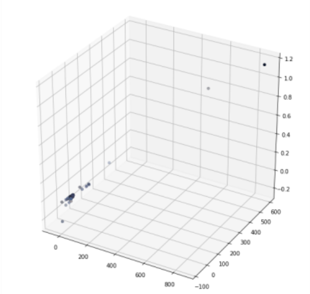
Structure from motion



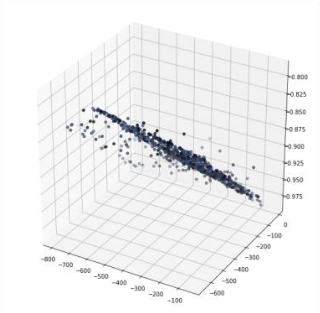
Structure from motion

Img 0 (614, 922, 3)

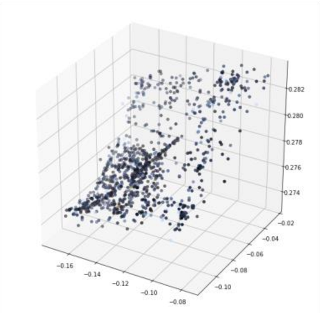
Img 1 (614, 922, 3)



3D points after triangulation



3D points after affine reconstruction



3D points after metric reconstruction

We were not able to implement properly the bundle adjustment to minimize the re-projection error and refine our results.

$$\min_{\mathbf{R}^i,\mathbf{X}_p} \sum_{i=1}^{i} \sum_{p=1}^{P} d(\mathbf{R}^i\mathbf{X}_p, \hat{\mathbf{x}}_p^i)^2$$

Conclusions

- The tested methods are theoretically robust, but rely on the keypoint or line detection.
- Refinement is mostly needed (RANSAC, Gold-Standard, etc).
- Abstract concepts difficult to interpret (e.g image of the absolute conic) and not easy to implement in some cases.
- Very active field in Computer Vision with plenty of real world applications.
- No deep learning is required, despite it is competitive in most cases.