GBM6700E – Fall 2025 Practical Assignment 2
3D Reconstruction of Coronary Arteries by Angiography

Polytechnique Montréal Biomedical Engineering program

Please update the python library before starting this assignment ! In your conda environment (activated with *conda activate*), run :

```
python -m pip install git+https://github.com/vtrno/PyGBM6700E.git
```

General objective

The main objective of this assignment is to study several factors involved in the acquisition, image feature matching and 3D reconstruction of cardiac structures (coronary arteries) acquired by the technique of X-ray angiography. Starting from a 3D reference model of a left coronary artery obtained by CT-scan, you will simulate different angiographic views and study the influence of calibration errors on the matching and 3D reconstruction of these same structures. Then, you will see how the self-calibration technique can improve the result obtained. Finally, you will consider the case of real angiographic images of a patient.

X-ray angiography is a medical imaging technique commonly used to examine a patient's cardiac structures. In this technique, the patient lies in supine position on a table and clinicians take a series of X-ray pictures or video sequences of the heart with a device (commonly called a C-arm) that can move around the patient. Beforehand, a catheter is directed to the root of the coronary arteries and a radio-opaque contrast agent is injected in order to clearly visualize these blood vessels, which feed the heart muscle (myocardium). Figure 1 shows the radiographic system used at Sainte-Justine Hospital in Montreal.

Figure 1: C-arm at Sainte-Justine Hospital : Infinix-CFI BP by Toshiba

The image formation model considered here is related to the pinhole camera model. The following parameters are the main ones that describe the acquisition of a view with the angiographic system:

- SID (Source to Imager Distance): distance from the source to the detector (image plane)
- SOD (Source to Object Distance): distance from the source to the object (patient)
- DP (Dimension of Pixel): horizontal or vertical size of a pixel in the image plane
- Primary angle (alpha): orientation of detector in the patient's axial plane; positive toward the patient's left, negative toward the patient's right

- Secondary angle (beta): orientation of detector in the patient's sagittal plane; positive toward the patient's head (cranial), negative toward the feet (caudal)

In addition, the intersection of the optical axes corresponding to the different views taken during an angiography session is defined as the isocentre. The isocentre is normally located at the object of interest, i.e. the patient's heart. Figure 2 illustrates the different parameters that we have just described.

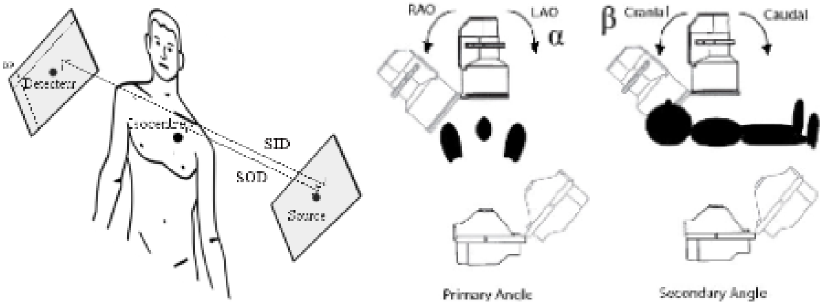

Figure 2: Main parameters characterizing image acquisition with the C-arm

Data files and functions provided

3D reference model

The segmentation of a CT-scan of the heart has enabled a 3D reference model of the coronary arteries to be obtained in the form of volume data. The space is discretized in the form of a 3D grid composed of rectangular cells (voxels); a set of these cells constitutes the geometry of the blood vessels (APPENDIX A provides a diagram illustrating the main branches of the coronary arteries).

Next, we extracted from the shape of the left coronary artery its 3D skeleton, namely a set of curves that represents the central line of the vessels and that preserves their topology. Each of these curves is discretized into a set of 3D points. Figure 3 shows the shape of the left coronary artery (approximated by an isosurface) with the skeleton inside. The file 'Coronary.mat' contains the structure. Read it using the *lab_2.io.load_mesh()* function.

NOTE: the isosurface is provided for visualisation purposes; it is the skeleton that you will manipulate to carry out the work.

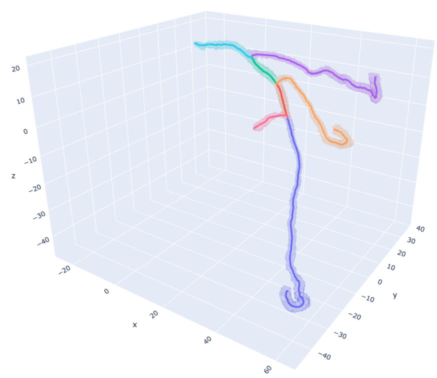

Figure 3: Left coronary artery and its extracted 3D skeleton

Real angiographic images

The files PATIENT_XXX_RAW.tif contain original image file of left coronary artery.

Python library

The documentation is available on this webpage It contains all the necessary functions to complete this assignment, including :

- *lab_2.geometry.build_view_geometry* : function that constructs the geometry of image acquisition for a given viewpoint from the 5 DICOM parameters and image dimensions given as input
- *lab_2.geometry.compute_8points_algorithm* : function that applies the normalized 8 point algorithm to compute the fundamental matrix and the epipoles
- *lab_2.utils.interpolate* : function that interpolates the input point set, in order to over- or under-sample a 2D or 3D curve
- *lab_2.geometry.rectify*: function that computes the image transformations and modified projection matrices to rectify the stereo images;
- *lab_2.geometry.refine_camera_param*: function that applies the Levenberg-Marquardt algorithm to refine the explicit parameters of a view
- *lab_2.geometry.get_dicom_from_calib*: function that retrieves the DICOM parameters describing the view from the explicit calibration parameters;

Part A (8 points)

Question 1 (4 points)

Starting From the CT volume of the left coronary artery (in fact, from the 3D skeleton already extracted), simulate two angiographic views of the left coronary artery (its skeleton) by choosing the DICOM parameters based on the information provided in APPENDIX B. Display the two images obtained. The resulting images will be those to use in the following questions.
Then, calculate the fundamental matrix (F) relating the two viewpoints, following three methods:

(a) directly from the known calibration parameters

(b) using the 8-point algorithm (with provided function), with a set of 8 corresponding points from the two images

(c) using the 8-point algorithm, but with all the available corresponding points from the two images

What do you observe about the three F matrices obtained?
What would happen in the presence of noise in the images?

Question 2 (4 points)

The problem of image matching consists in determining a set of pairs of points, each pair formed by a point from image I_1 and a point from image I_2. The image points are the projections of physical points in the 3D scene. The challenge lies in correctly identifying, in each of the images, the projection of the same point from the scene. This matching step is of great importance for the 3D reconstruction step. Indeed, an imprecise matching can lead to important errors in 3D reconstructions.

There exist many point matching methods, but the common element is always to apply constraints to restrict the set of solutions. The epipolar constraint is one of the intrinsic geometrical constraints of a stereoscopic system. It allows us to reduce the set of initial solutions (i.e. the whole image), to a line in the image, called the epipolar line.

Use the following steps to exploit the epipolar constraint to establish the correspondence between the central lines (the branches of the skeleton) of the vessels between the two views:

1. Select one of your images as View 1 and the other as View 2
2. Resample the branches in View 2 to double the density of points (using the provided function)
3. For each point in each branch in View 1:
   (a) Compute the epipolar line in View 2
   (b) Find the closest point in View 2 to that line
   (c) Retain that point as the match for the point in View 1

Keep the obtained sets of corresponding 2D points for the subsequent questions. For illustration purposes, provide two figures showing examples of the epipolar lines for a selected branch, in each of the two views. (For the epipolar lines in View 1, these are the conjugates of those in View 2). Answer the following questions:

- What can you observe in the figures showing the epipolar lines ?
- In general, why is it useful to use the resampling + epipolar constraint approach to find matching points in curvilinear structures (e.g. in medical images)?

Part B (9 points)

Question 3 (5 points)

In the case of a stereoscopic system that is not fully calibrated, calculating a **depth map** can provide a first approximation of the 3D shape of an object in the scene. In this question, you will construct such a depth map and compare it to the real 3D structure of the coronary tree.

Construct the depth map of the vessel central lines with the following steps:

1. Compute the disparity between the pairs of corresponding points obtained in Question 2
2. Compute the depth Z at each point in View 1, using the following formula:

$$Z = \frac{f * B}{d}$$

   where

   - B is the base, i.e. the distance (in mm) between the two sources, use the known geometric parameters to calculate it.
   - d is the disparity (in pixels) at a given 2D points in the 1st view
   - f is the focal length (in pixels) of the camera

Next, construct another version of the depth map, by first **rectifying** the two images (using the provided function), recalculating the disparities between point pairs and recalculating the Z map.

Display as 2D figures with color maps: first the disparity maps, then the depth maps in each case. Now, construct the **real (ground truth)** depth map of the vessels' skeleton. To do this, you must first use the known geometric system parameters to transform the reference model to the coordinate frame of the 1st source. Consider the Z coordinates of the transformed model as the depth map.

Display this depth map similarly to the previous ones.

Compare the different figures (maps) obtained **qualitatively** (visually): do they make sense? When compared to the real depth map, which of the maps computed from disparities is most similar to the ground truth? Discuss your results.

NOTE : to plot your results, you can both use matplotlib or plotly, look at their respective documentations to learn more about colorbars and plot types

Bonus (2 points) : Determine a way to compare the different depth maps computed from disparities quantitatively with the ground truth map. What do these new results tell you about the effect of rectifying the images?

Question 4

Since you have a fully calibrated system (all geometric and image formation parameters are known), you can compute the 3D reconstruction of the vessels from their 2D projections, as in LAB 1.
Using the method learned in the 1^{st} practical assignment, reconstruct the central lines of the vessels in the ideal case (based on the calibration parameters that you used to generate the simulated views), using the 2D point coordinates and the correspondences found in Question 2.
Display as a 3D figure the reference model and this new reconstructed model superimposed (in different colours). Are there any discrepancies?
Next, perform the **back-projections** of the reconstructed 3D model to the two image planes (keeping the same viewpoints as in previous questions). Referring to the formulas for RMS errors given in LAB 1, evaluate the errors between the 3D reconstruction and reference model, and the errors in each image plane between the 2D back-projected points and the initial image points, i.e. those resulting from Question 2 (after resampling and matching).

Now, **progressively add noise** to the DICOM geometrical parameters (SID, SOD, DP), by adding an error as an increasing percentage of the starting value, and in the same range for angles alpha and beta on the 5 parameters at the same time. In each case, perform the steps of 3D reconstruction and 2D back projection, and compute the RMS errors with respect to the ground truth model and images.

Plot the curves of the 3D and 2D errors with respect to the amplitude of the added noise. Discuss the different results.

NOTE: when adding the noise to the parameters, you can introduce random elements to the amounts added to – or subtracted from – the original values, while keeping the overall error proportional to the percentage being tested. Also, you can take the average of several random trials to compute the RMS curves. Keep one of the trials for Question 5.

Part C (8 points)

Question 5 (5 points)

We now want to see how to use **self-calibration** to try to correct the errors introduced in Question 4 and retrieve their real values. To simulate a real-world case, you will use the initial (non-noisy) 2D image data, but the **noisy** 3D model data, as inputs to the refinement algorithm. (Indeed, in a typical medical imaging situation, e.g. angiography using C-Arm, one would have the real images captured by the system, but not necessarily the exact imaging parameters. Those included in the

DICOM data are nominal values that often differ from the real ones.)

Use the provided function (which implements the Levenberg-Marquardt algorithm) to refine (correct) the system calibration parameters for each view, for the same error levels as tested in Question 4. In each case, using the geometric parameters output from the function (K, R, t) for the two views, perform the 3D reconstruction and the 2D back-projections, and evaluate the different RMS errors, as in Question 4.

Plot the new curves of the 3D and 2D errors with respect to the amplitude of the added noise. What happens to 2D back-projection errors? What happens to 3D reconstruction errors?

Now, using the provided function, convert the (K, R, t) parameters found in each case by the refinement algorithm back into their corresponding DICOM parameters (SID, SOD, alpha, beta). Compare the noisy and corrected values of these 4 parameters, by plotting them e.g. as bar charts. Does the self-calibration succeed in finding the correct system parameters? Discuss the different results.

NOTE: when using the conversion function get_dicom_from_calib, the value of DP must be provided as input. Use its initial (non-noisy) value for this purpose.

Bonus (1 point): Based on the material covered in the course, propose a better means of self-calibrating the system parameters.

Question 6 (3 points)

Now consider the real clinical data provided with this assignment (PA and LAT angiographic views of a pediatric case taken with the C-Arm at Sainte-Justine). How would you go about reconstructing the arteries in 3D from the angiographic images? What difficulties do you foresee in the real case?

NOTE: This last question is meant as a discussion question. You are not expected to actually implement a reconstruction method with these images. Furthermore, the images provided are meant as examples of the kind of images to use, but your proposed approach could employ other viewpoints than PA and LAT.

Report and submission

Your written report for this assignment should be concise, clear, and must contain:

- an introduction (worth 2 points) providing a description of the problem being addressed and the techniques used in the lab
- the answers to the questions asked, the steps followed, figures showing your results, discussions where appropriate and any comments you deem useful
- a conclusion (worth 3 points), summarizing the things learned in the assignment, the limits of the approaches utilized and ideas on how to improve those approaches
- bibliographic references, if any.

REMARK: the report should not contain any code listings (in Matlab or any other language used to obtain the results).

Appendix A

Here is a diagram illustrating the main branches of the left and right coronary arteries, in antero-posterior (AP) view.

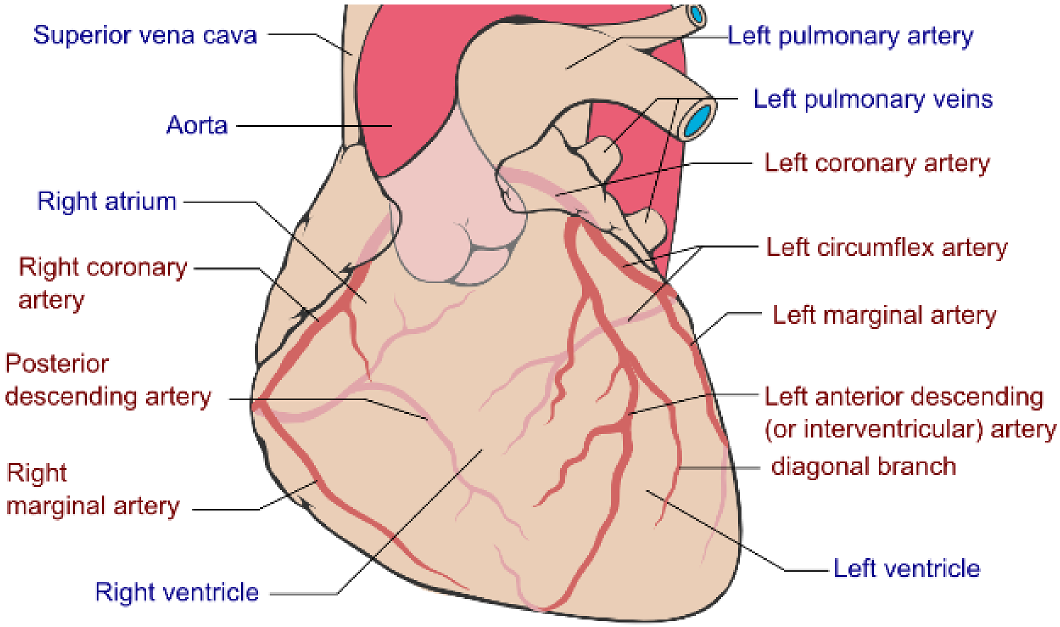

Figure 4: Nomenclature of the segments of the left and right coronary arteries
Taken from here

Appendix B

For the choice of parameters to simulate images taken with the angiography device, in all cases, the following three values can be taken as fixed values:

- **SID** : 1050 mm;
- **SOD** : 750 mm;
- **DP** : 0.1953 mm.

Next, for the primary and secondary angles, these may vary within the following ranges:

- **Primary angle** : -90 deg. $\leq$ **alpha** $\leq$ 90 deg.
- **Secondary angle** : -40 deg. $\leq$ **beta** $\leq$ 40 deg.

The following are typical examples of viewing angulations taken in the cardiology clinic:

- Principal plane views
  - **Antero-posterior view (AP)** : **alpha** = 0 deg; **beta** = 0 deg.
  - **Lateral view (LAT)** : **alpha** = 90 deg; **beta** = 0 deg.
- Oblique views
  - **30 LAO, 25 CRA** : **alpha** = 30 deg; **beta** = 25 deg.

- **30 LAO, 25 CAU** : **alpha** = 30 deg; **beta** = -25 deg.
- **30 RAO** : **alpha** = -30 deg; **beta** = 0 deg.
- **30 RAO, 25 CAU** : **alpha** = -30 deg; **beta** = -25 deg.
- **40 RAO, 15 CAU** : **alpha** = -40 deg; **beta** = -15 deg.
- **45 LAO, 30 CRA** : **alpha** = 45 deg; **beta** = 30 deg.

You can choose other combinations of the two angles, the idea being to find two projections that allow a good visualization of the different vessels of the left coronary artery tree.

Finally, in all cases, take the image size as fixed and equal to 1024 x 1024 pixels and take the principal point as the centre of the images.