Digital Geometry Processing (master IPCV)

> Course (Semester 9) of <a href="ipcv.eu"> Master Erasmus Mundus IPCVai </a> (Artificial Intelligence for Image Processing and Computer Vision)

> [name=Jacques-Olivier Lachaud][time=Dec 2025][color=#907bf7]

tags: `lecture` `digital geometry` `geometry processing` `DGtal` `3d`

This course aims at presenting the field of digital geometry and its applications to the analysis and processing of (mainly) 3D images. Digital geometry views shapes as subsets of the lattice grids in arbitrary dimension. 2D shapes are thus sets of pixels, while 3D shapes are sets of voxels. It provides consistent definitions for contours and surfaces, and convergent definitions of tangents, normal vectors, curvatures. It is thus a sound counterpart to Euclidean geometry, but adapted to discrete data as provided by imaging devices. It also develops a sound discrete exterior calculus on digital surfaces, thus providing all the necessary tools for manipulating vector fields on such objects, or doing spectral geometry processing.

The course takes place from December 9 to December 11, 2025, in Bordeaux. This page provides some information and ressources useful for this lecture.

[TOC]

Lectures

- <a href="https://jacquesolivierlachaud.github.io/talk/introduction-to-digital-geometry/"> Introduction to Digital Geometry </a>

Preparing practicals

Practicals will use the DGtal library (https://dgtal.org), an open-source generic C++ library well adapted to digital geometry processing. Coding will be in C++. The practicals are variants of the DGtal tutorial given at conference DGMM'2022 in Strasbourg, France.

:::danger
Before starting this course, **make sure** you have a working environment for **compiling C++** (typically Visual Studio with C++ extensions on Windows, g++ or clang++ on MacOS or Linux with some editors). Install also **cmake** from Kitware (https://cmake.org).
:::

1. Visit and clone repository https://github.com/JacquesOlivierLachaud/IPCV-practicals

Note that this repository contains configuration files for compilation, some example c++ programs using DGtal (exampleBoard2D.cpp, examplePolyscope.cpp), some skeleton c++ programs for each practical, and some data files (in `data/`).

:::info
You will gain a lot of time if you are able to compile and test the already written template codes in the repository.

:::

2. Download all dependencies and set-up the environment for compiling practicals

After cloning IPCV-practicals, set-up the compiling environment in a terminal with

```
cd IPCV-practicals
mkdir build
cd build
# configure the compilation in Release mode (for performance).
cmake -DCMAKE_BUILD_TYPE=Release -DCMAKE_POLICY_VERSION_MINIMUM=3.5 ..
```

It will automatically download `boost`, `eigen`, `polyscope` and `dgtal`.
This should finish with the following lines:

```
-- Configuring done (1.9s)
-- Generating done (0.1s)
-- Build files have been written to: ${HOME}/.../IPCV-practicals/build
```

After that just type `make` to compile all the provided programs, which are the skeleton c++ files of the different practicals.

3. Have a look at datas and provided programs

A few 3D vol images are provided in `data` subdirectory. You may have a look at some binary or gray-level 3d image with `volViewer`. In `build` directory, you may try:

```
# A 3d shape (binary image)
./volViewer ../data/fertility-128.vol
# A thoracic CT scan, showing fat, lungs, heart, and the pulmonary vascular
system.
./volViewer ../data/PA05-img-low.vol
# The segmentation of its pulmonary vascular system.
./volViewer ../data/PA05-lbl-low.vol
```

Data comes from VolGallery (https://github.com/dcoeurjo/VolGallery) or parse2022 (https://parse2022.grand-challenge.org/Dataset/).

Practicals

- TP1 Filtering and segmentation of 3D CT thoracic image (https://codimd.math.cnrs.fr/s/VSydn_j1G)

  > 3d image filtering, digital surfaces, connectedness

- TP2 Homotopic thinning (https://codimd.math.cnrs.fr/s/bVqUMFbM5)

  > 3d shapes, digital topology, simple points, cubical complexes

- TP3 Geometric estimations on 3d digital surfaces (https://codimd.math.cnrs.fr/s/s2pNRQuga)

3d shapes, geometric estimators, multigrid convergence

- TP4 Scale axis transform (https://codimd.math.cnrs.fr/s/hQb0tRXHP)

    3d shapes, distance transform, scale axis, geometry processing