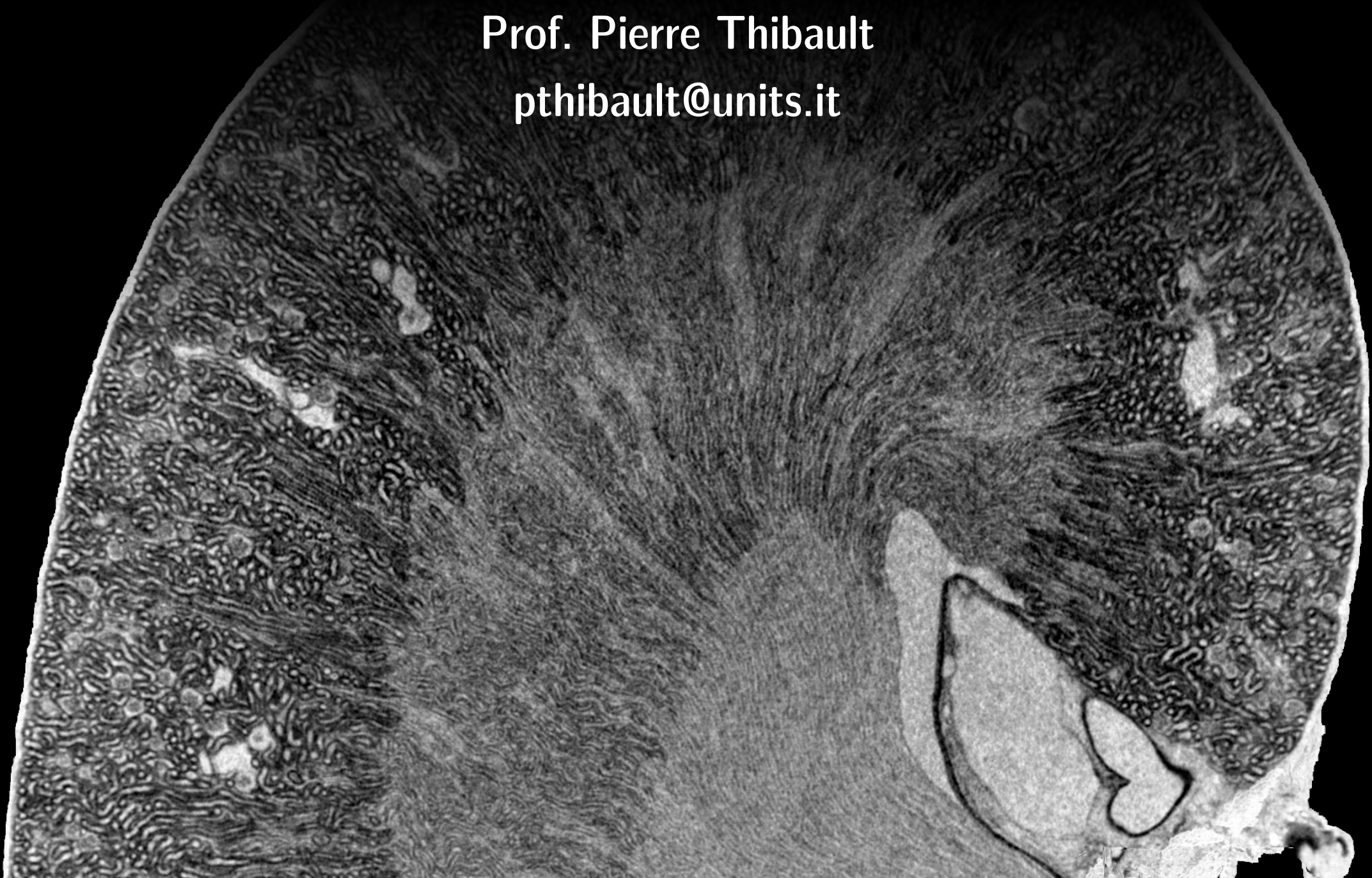# Image Processing for Physicists
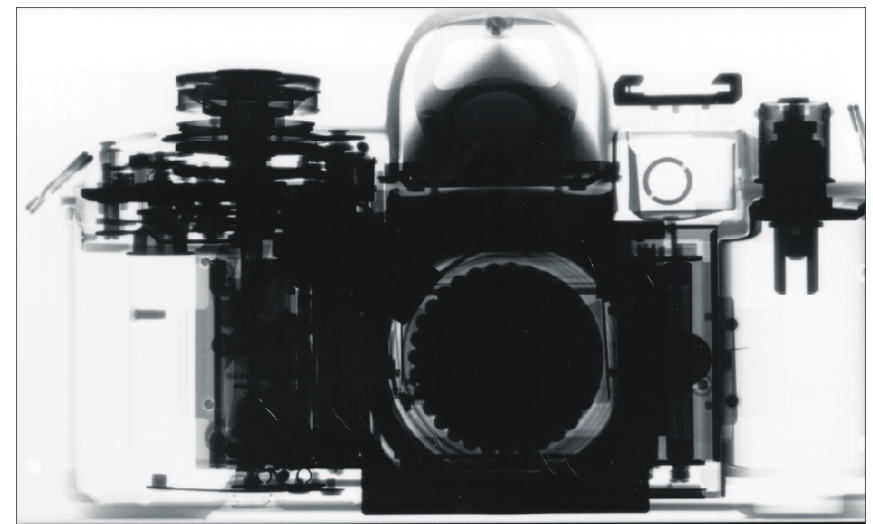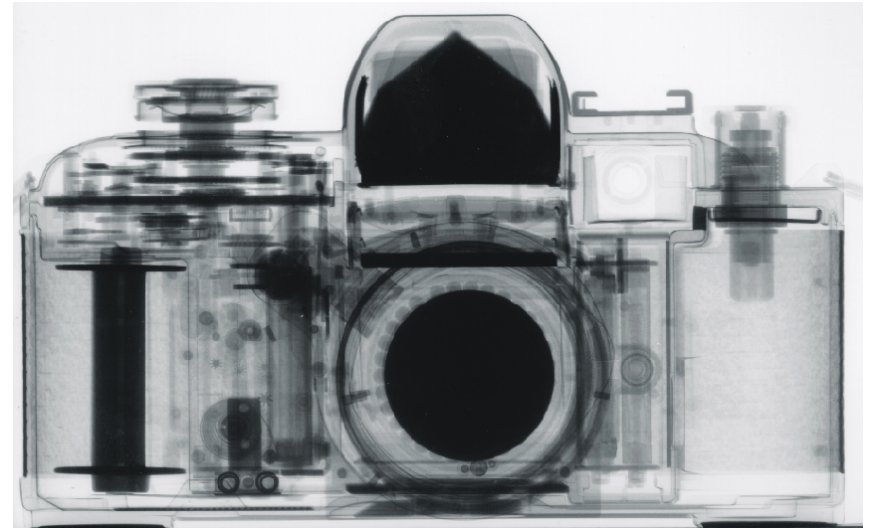
Prof. Pierre Thibault

pthibault@units.it

Questions addressed

- What is an image? How are images acquired, stored, analysed?

- What mathematical, numerical and physical methods are used to:

  - generate images?

  - denoise images?

  - segment images?

  - classify images?

  - compress images?

  - reconstruct images?

- What is "legit" and what isn't?

Syllabus

1. Spatial Domain

2. Fourier Domain

3. Sampling & Interpolation

4. Image Representations

5. Characterization Of Detection Systems

6. Imaging Systems & Wave Propagation

7. Interferometric Imaging And Imaging Of Far-Field Fourier Amplitudes

8. Tomography

9. Least Squares Optimization

10. Constrained Optimization & Maximum Likelihood Optimization

Philosophy

Techniques

Core knowledge of most common mathematical and numerical tools for imaging and microscopy, from a physicists view point.

Coding skills

Opportunity to improve (python) coding proficiency (more *scripting* than *programming*).

Decoding

Learning the terminologies to understand quickly research work that use imaging.

Critical thinking

Lean to identify the proper tools for a specific imaging need, analyse and criticise image processing operations found in the literature.

Admin

- Online only

- Mon - Wed 14:00-16:00

- Office hours (online): Wed 9:00-10:00

- Lectures $+$ tutorials

- Weekly exercises (python)

- Final assessment: oral

Example

Package imports

```
In [57]:  import numpy as np
          from matplotlib import pyplot as plt
          from scipy import ndimage as ndi
          from skimage.segmentation import watershed
          from skimage.morphology.convex_hull import convex_hull_image
          from skimage.feature import peak_local_max
          from skimage.measure import regionprops
          import rasterio
```
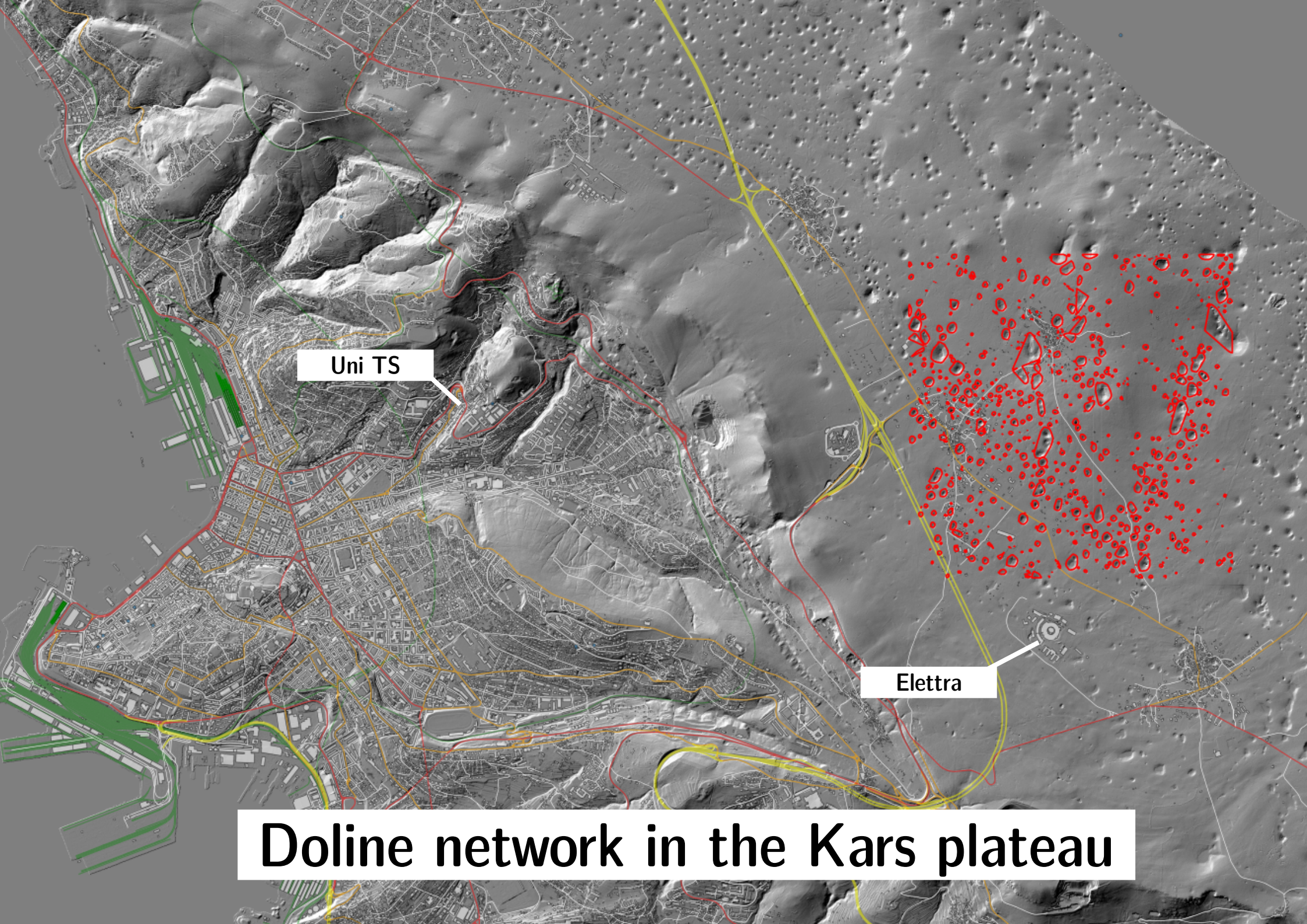
Load image

```
In [83]:  with rasterio.open('stitched_topo_TS.tiff') as raster:
              img0 = raster.read(1)
          img = img0[::2, ::2] + 0.
          img *= (img>0)
          cimg = img[1400:2600,3400:4600].copy()

          fig, ax = plt.subplots(figsize=(24,24))
          plt.imshow(cimg, cmap='gray', vmin=350, vmax=450)
```

```
Out[83]:  <matplotlib.image.AxesImage at 0x7f715e32ecd0>
```
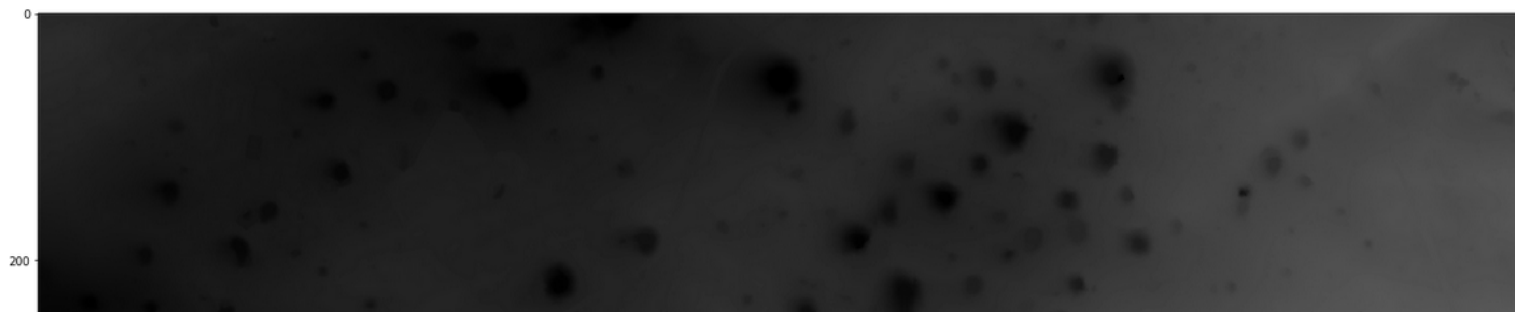
Doline network in the Kars plateau