Local symmetry detection in Earth observation (EO) data

Progress report

Maribor, 11th April 2022

David PODGORELEC

University of Maribor
Faculty of Electrical Engineering and Computer Science
Institute of Computer Science
Laboratory for Geospatial Modelling, Multimedia and Artificial Intelligence

Motivation

GeoSym Objective O3: *Integration of symmetry detection into the methodology of semantic segmentation and object recognition in EO data in order to improve accuracy and enlarge the set of recognized classes, validated in a dedicated set of applications.*

Due to the nature of EO data and their acquisition, it seems that only the **local & approximate** symmetry detection can realize O3. While waiting for adequate solutions from Pilsen, we are trying to provide some inputs (local reflectional and rotational symmetries in EO data) ourselves.

EO data

1) Sampled data
   - Discrete point cloud, voxel or raster grid…
2) Top view
   - 3D data, (mostly) acquired „down“ from a satellite, airplane…
   - Much more data collected from the visible top faces than from the side and bottom faces.
   - Lower (side and bottom) points may be included in datasets, depending on view angle and/or multiple reflections.
3) Usually, width and length of the considered area are much greater than the range of altitudes.
   - Bigger geographic areas are relatively flat.
   - 2D or 2.5D data. Altitude as an attribute in 2D GIS data.

EO data

- Implications on symmetry detection

1. Due to sampling: points of an „original" part and mirrored/rotated part(s) rarely match exactly.
2. Due to higher density of data on visible top sides: it is more likely to detect symmetric parts there.
3. Due to „flatness" of acquired areas: more likely to explore symmetries from above than from side.

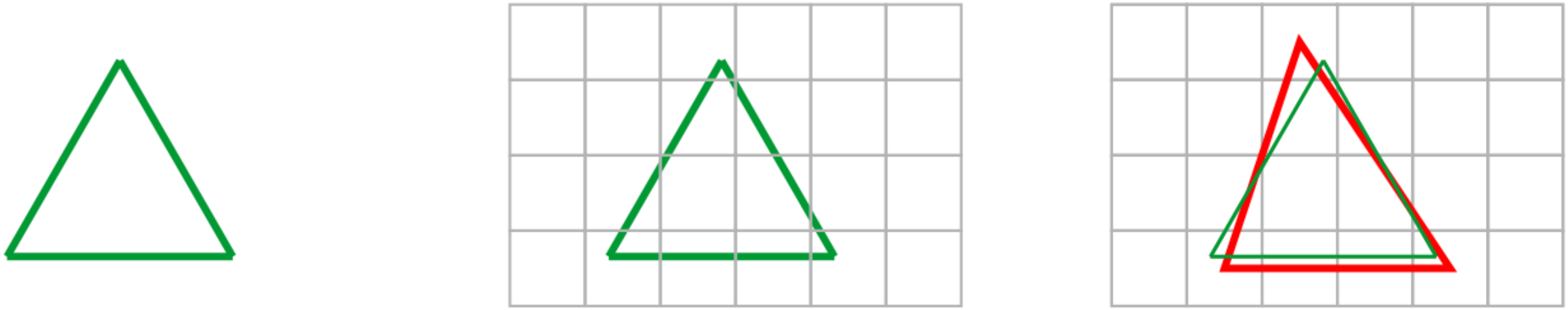
- For these reasons (and simplicity ☺), we initially focus on rotational and reflectional local symmetries **with vertical symmetry axes/planes**.
- GIS platform initially designed for 2D visualization, too.

Concept

- Due to sampling and visibility limitations, approximate symmetries can only be considered.
  - Surface reconstruction, tollerances or voxelization?
  - Interval arithmetic must be defined.
- Due to restriction to vertixal symmetry axes/planes, it suffices to detect 2D symmetries in horizontal slices and then merge them with respect to detected vertical symmetry axes/planes (and rotation angles).

Voxelization

- Any point inside a voxel is replaced by the voxel's centre.
  - Left, front, bottom boundaries also part of the voxel.
- Voxels containing EO data points are interesting voxels.
- Straigtforward, but...
- Lengths of line segments not preserved → relations lost.



- Example: an equilateral triangle turns into a scalene one!
- Interval arithmetic!

Concept

- Bottom-up approach (in each slice): Find basic symmetries and construct larger ones by merging.
- Basic symmetry: symmetry (or candidate for symmetry) between two geometric primitives.
- Primitives to be used: points (voxels), line segments, or more complex structures? Our choice are line segments.
- Line segment (LS) is a pair of voxels (end-points of LS). It is characterized with its length and structure (distribution of interesting voxels along LS).
- Core idea is that each LS which appears in some symmetry should have a symmetric pair (copy) or more of them with the same length somewhere (in the slice).

Concept

Voxelization.

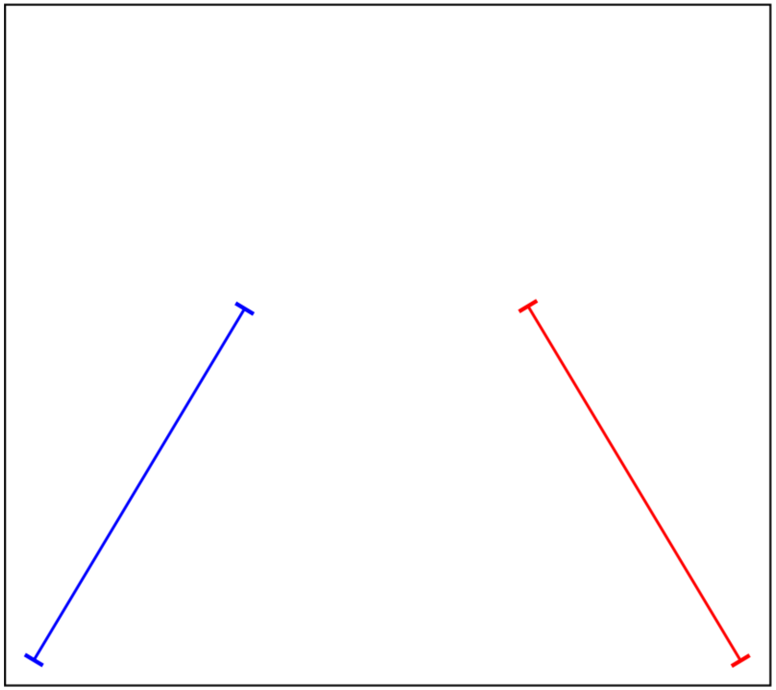
For each horizontal slice of voxel space

- Identify interesting voxels (pixels in slice).
- Create complete graph G on the interesting voxels.
- Cluster edges of G with respect to lengths (and structure).
- For each cluster
  - Establish basic symmetries among pairs of edges.
  - Merge symmetries.
- Merge symmetries from different clusters.

Merge simmetries from different slices.

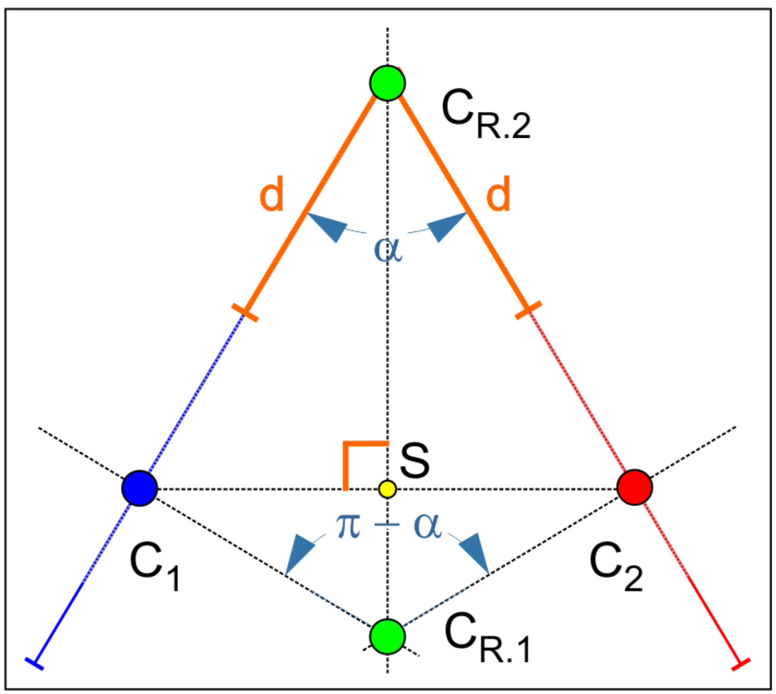
Establishing basic symmetries

- Are two LSs OF THE SAME LENGTH candidates for either rotational or reflectional symmetry (or for both)?



Establishing basic symmetries

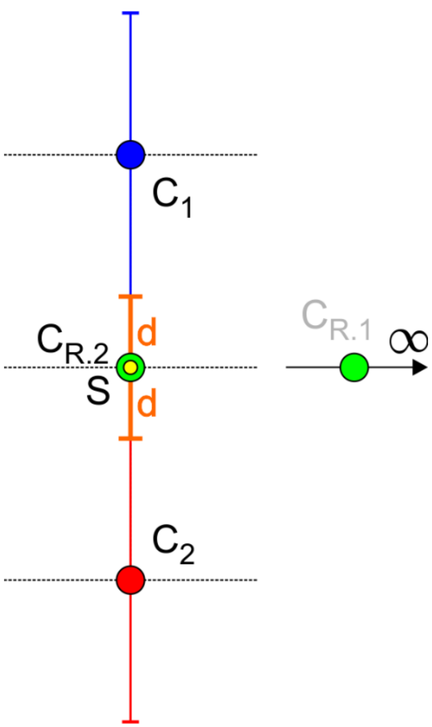
- Are two LSs OF THE SAME LENGTH candidates for either rotational or reflectional symmetry (or for both)?

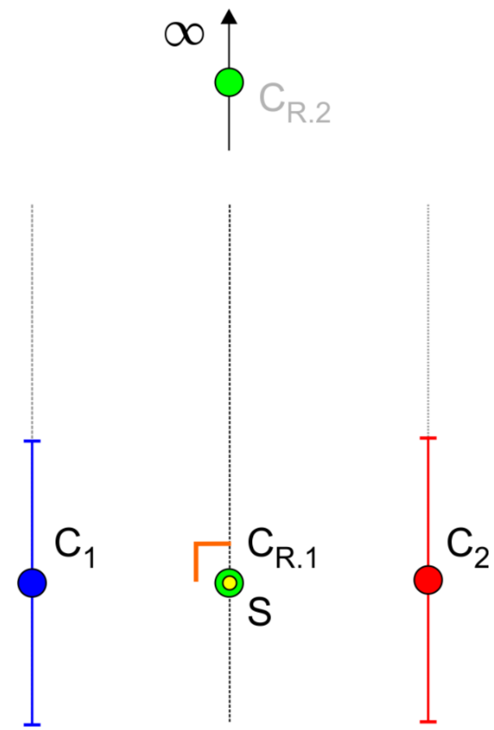


- If both LSs are equidistant (d) from the $C_{R.2}$, then $C_{R.1}$ and $C_{R.2}$ may be potential centres of rotational symmetries, and line ($C_{R.1}$, $C_{R.2}$) is the axis of reflectional symmetry.
- 90° in S is a consequence.
- Nice to simultaneously handle the reflectional and rotational symmetries!

Establishing basic symmetries

- In both parallel exceptions, one candidate rotational symmetry confirmed (for 180° or fewer), another refused. Reflectional symmetry remains.





Interval arithmetic

- $d(V_i, V_j) = [min(d(p, q)), max (d(p, q))]$
- Ok, but the problem is the intersection detection (point and angle) between two voxelised line segments.
- Particularly when two LSs are nearly parallel, there is quite a lot of intersection voxels, increasing the combinatorial complexity.
- While trying to solve (optimize) this problem, we (our students) have implemented another (brute-force) solution, at least to get some results.

Concept 2

Voxelization.

For each horizontal slice of voxel space

  Identify interesting voxels (pixels in slice).

  For each pixel P as a potential centre of rotational symmetry

    Detect rotational symmetries for different radii around P.

    Merge detected rotational symmetries.

  For each potential axis of reflectional symmetry(*)

    Detect mirror-symmetric pairs of voxels.

Merge simmetries from different slices.(**)

(*) Potential axis is any LS between a pair of pixels on different edges of the bounding rectangle.

(**) Two rotational symmetries with the same axis and with the rotation steps *s* and *k**s* may/must be merged into the symmetry with the step *s*.

Gray lines are the same as in the initial concept.

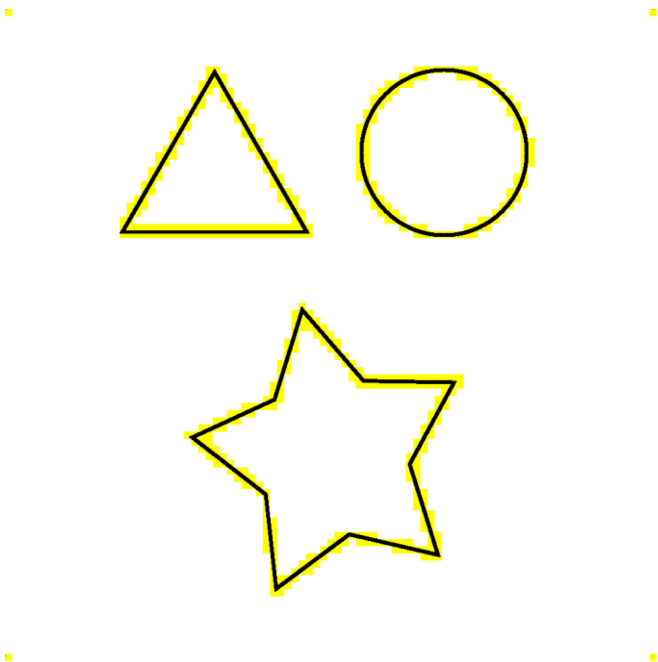
- Polynomial time, but quite slow.

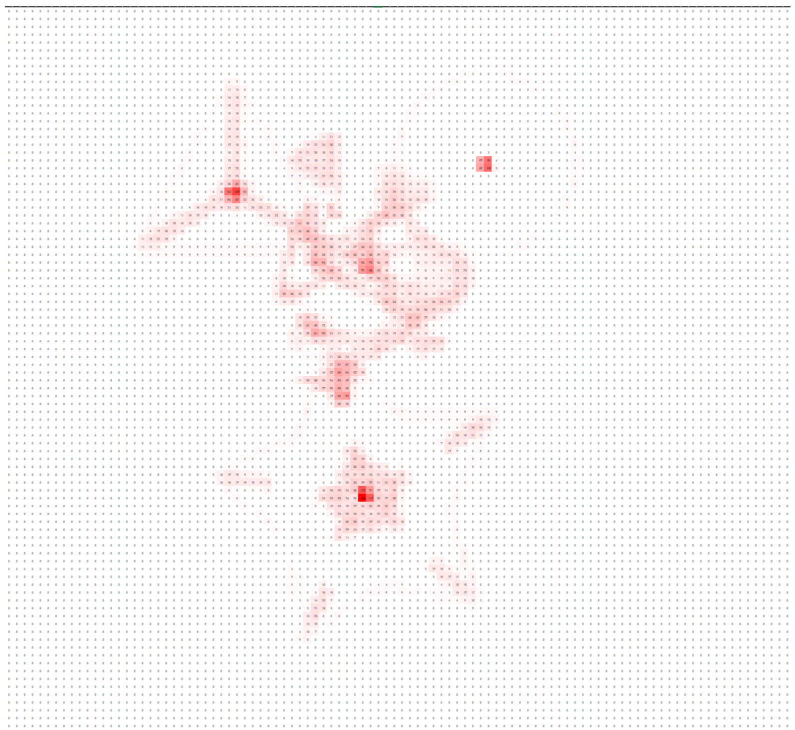
Concept 2: results

- 2D rotational symmetry:
  - Implemented, visualised, tested, presented & analysed.
  - Below 5 sec for presented cases with 10.000 pixels
- 2D reflectional symmetry:
  - Implemented, visualized, being tested.
- 3D rotational symmetry:
  - Implemented, tested, being visualized
  - Few minutes for 10.000 pixels; depending on altitude range.
- 3D reflectional symmetry:
  - Being in the implementation phase.

Concept 2 (2D rotational): test 1

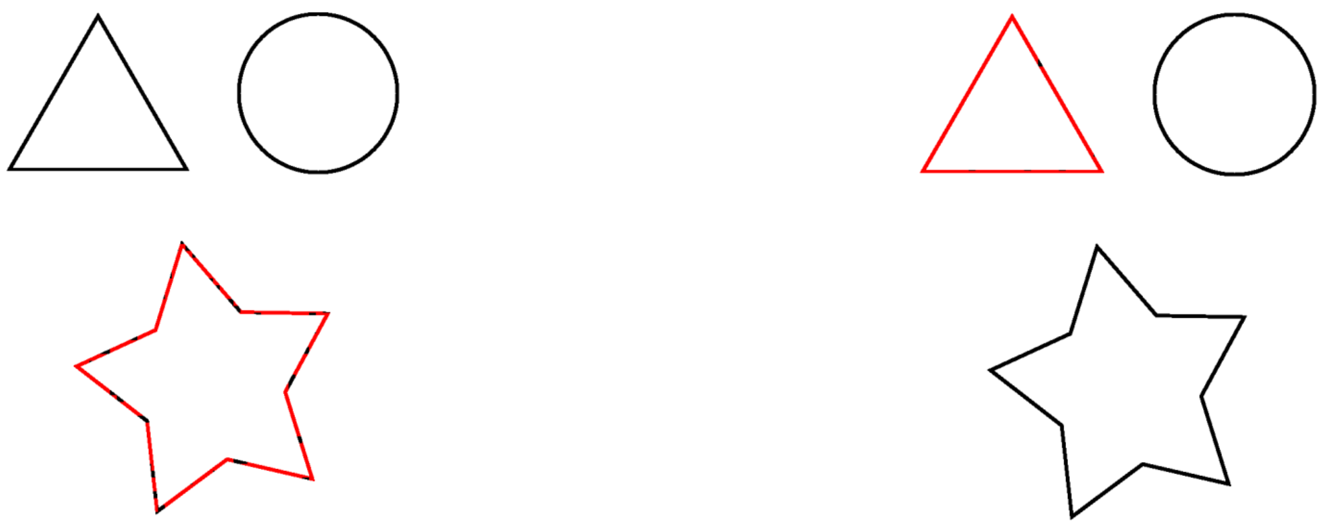
- Left: scene (black) and interesting voxels (yellow)
- Right: centres of detected symmetries (darker is stronger).





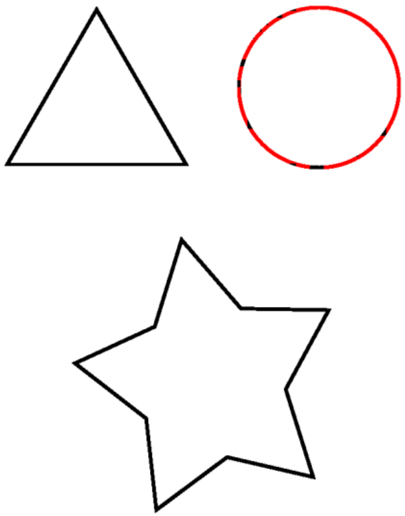
Concept 2 (2D rotational): test 1

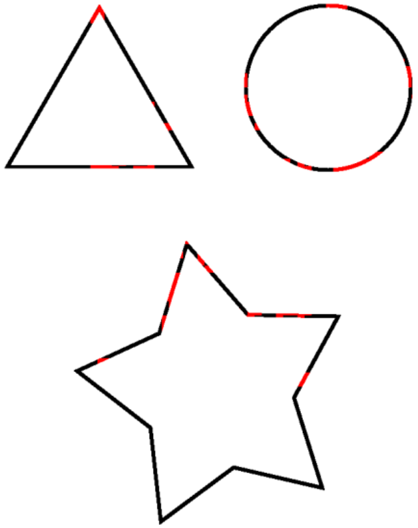
- Two strongest symmetries (due to number of interesting voxels involved): left of 5 rotations, right of 3 rotations.



Concept 2 (2D rotational): test 1

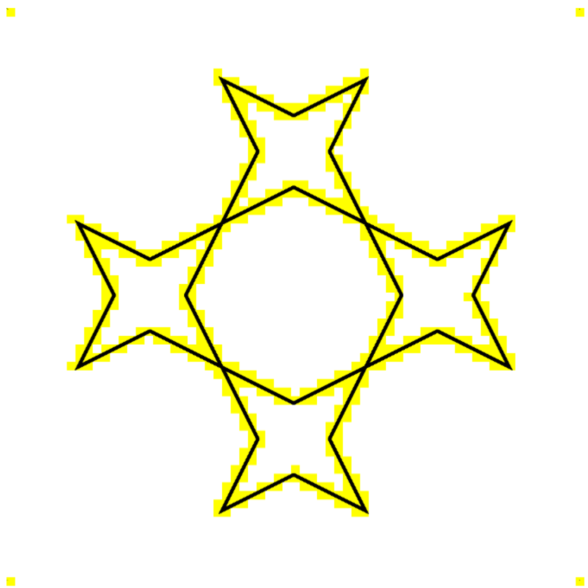
- Third and fourth strongest symmetries: left of „∝“ rotations, right irrelevant case due to objects interference.

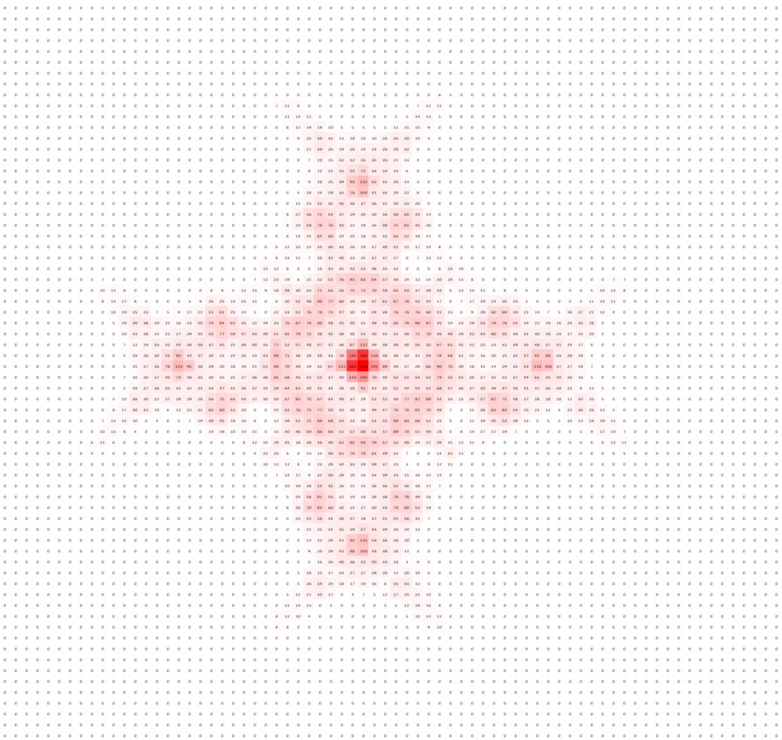




Concept 2 (2D rotational): test 2

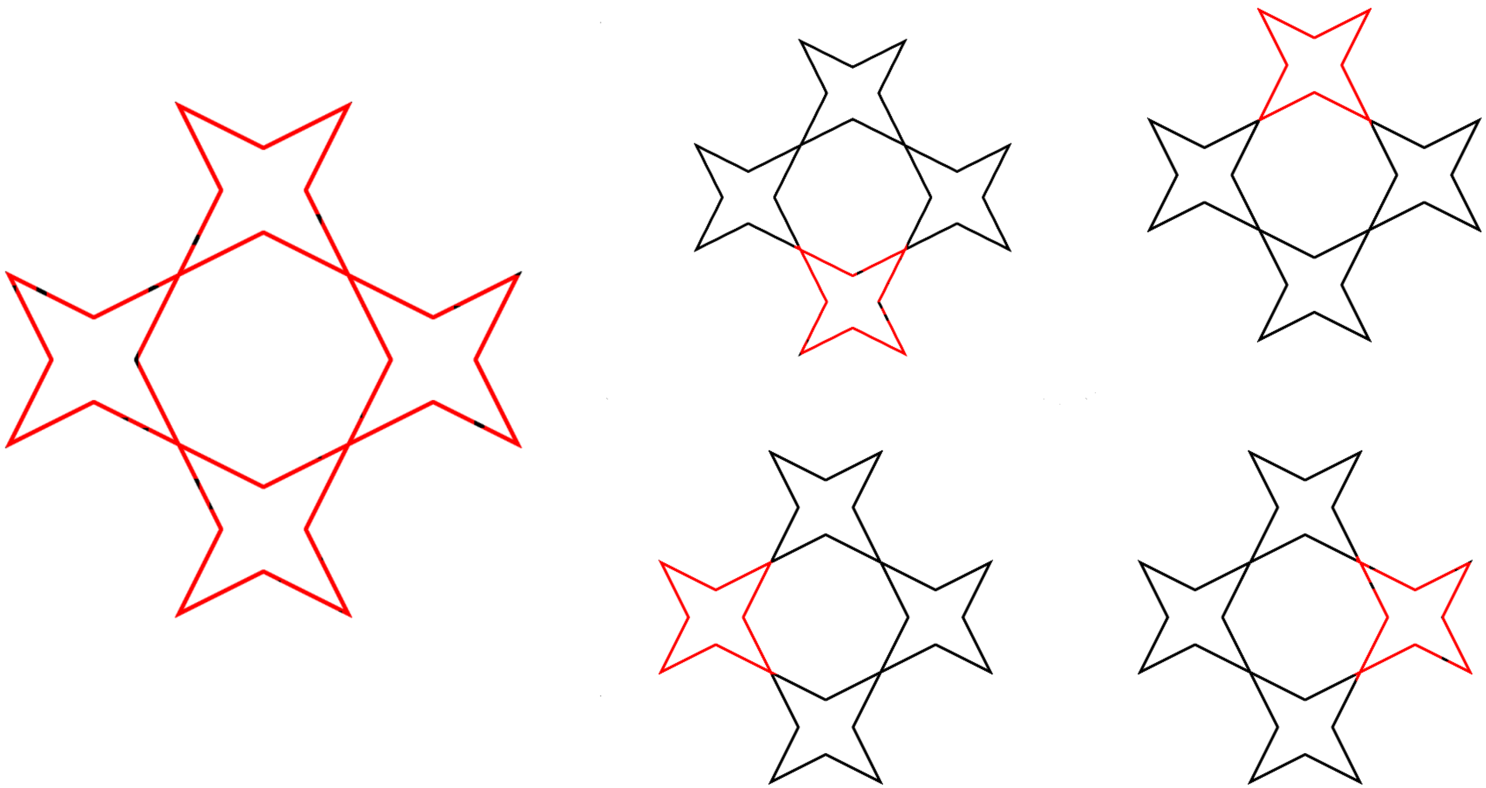
- Left: scene (black) and interesting voxels (yellow)
- Right: centres of detected symmetries (darker is stronger).





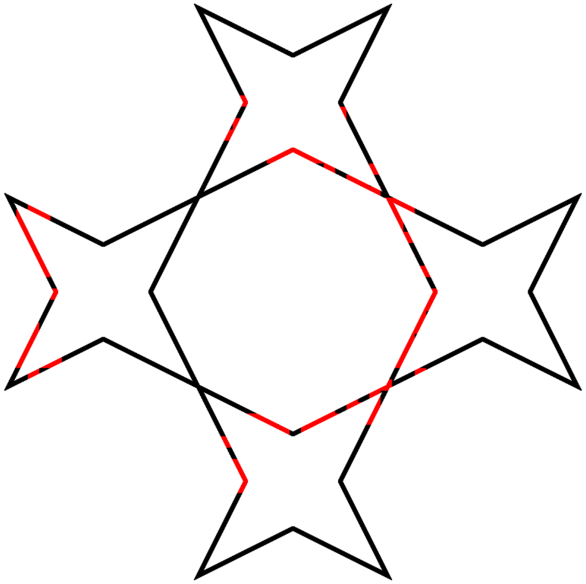
Concept 2 (2D rotational): test 1

- Five strongest symmetries (all of rotation step 4)



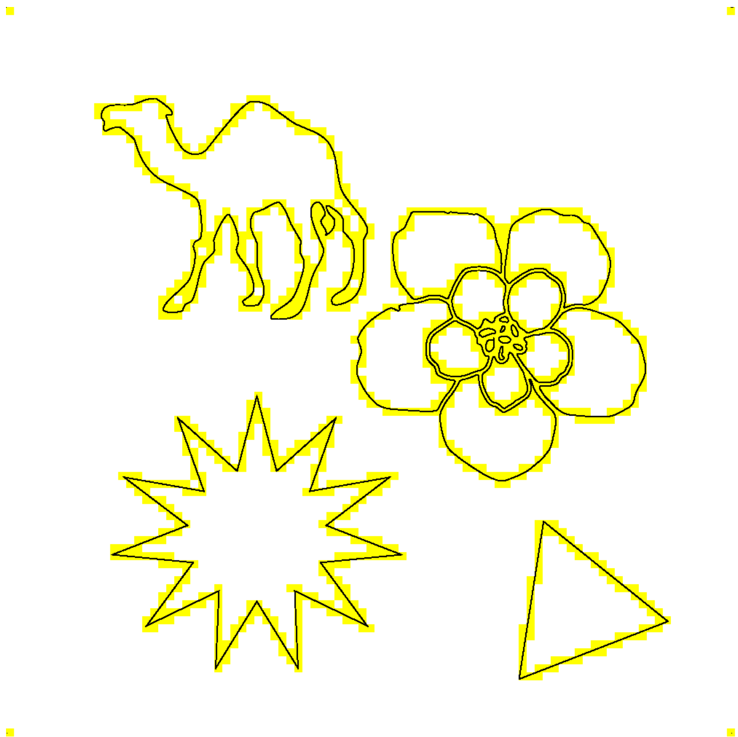
Concept 2 (2D rotational): test 2

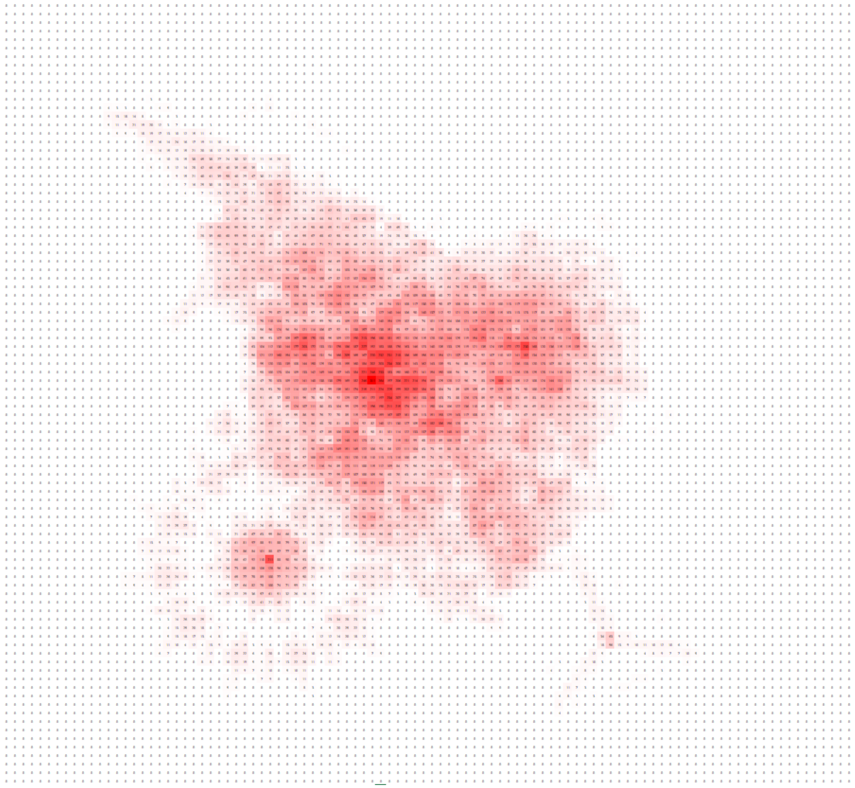
- 6th strongest symmetry (inter-object, irrelevant)



Concept 2 (2D rotational): test 3

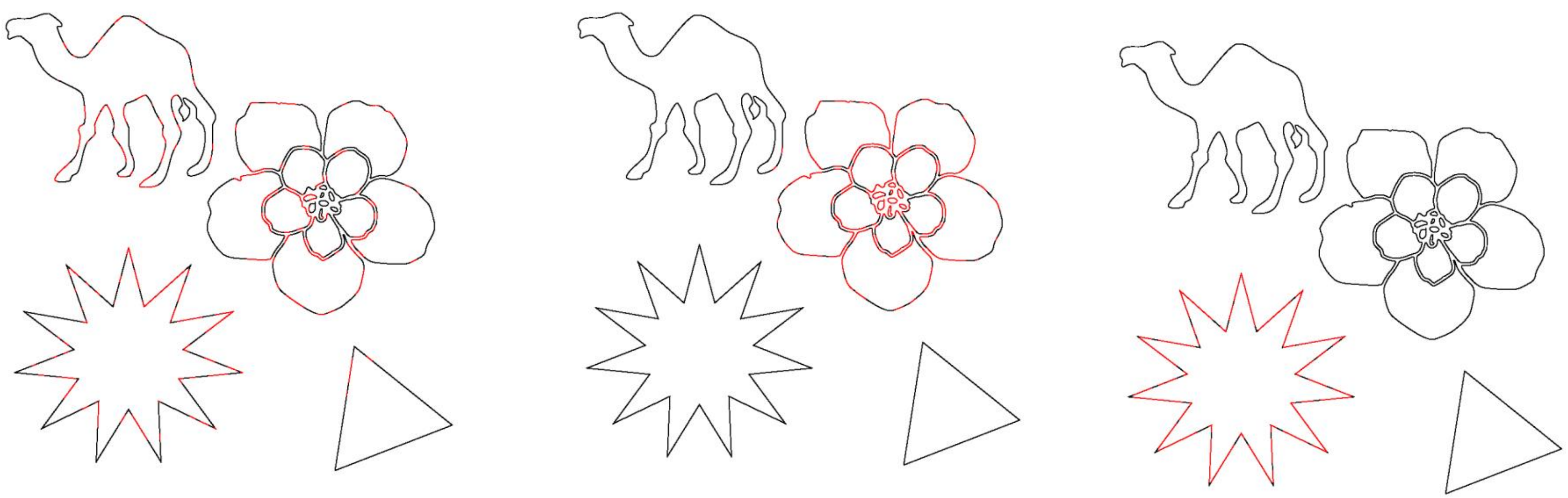
- Left: scene (black) and interesting voxels (yellow)
- Right: centres of detected symmetries (darker is stronger!!!).





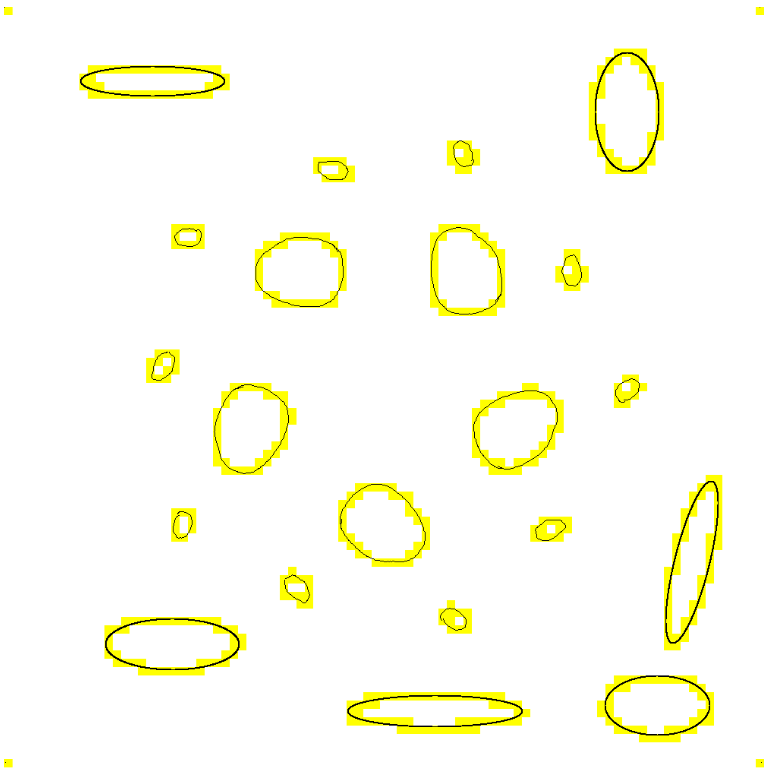
Concept 2 (2D rotational): test 3

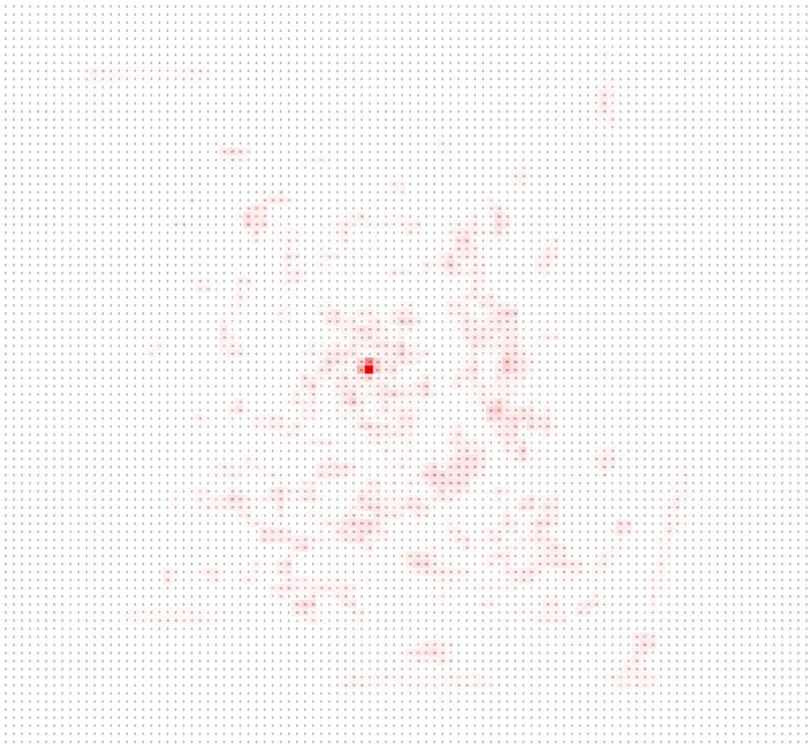
- 1st (rotation step 3), 5th (step 5) and 9th (step 11) strongest symmetries



Concept 2 (2D rotational): test 4

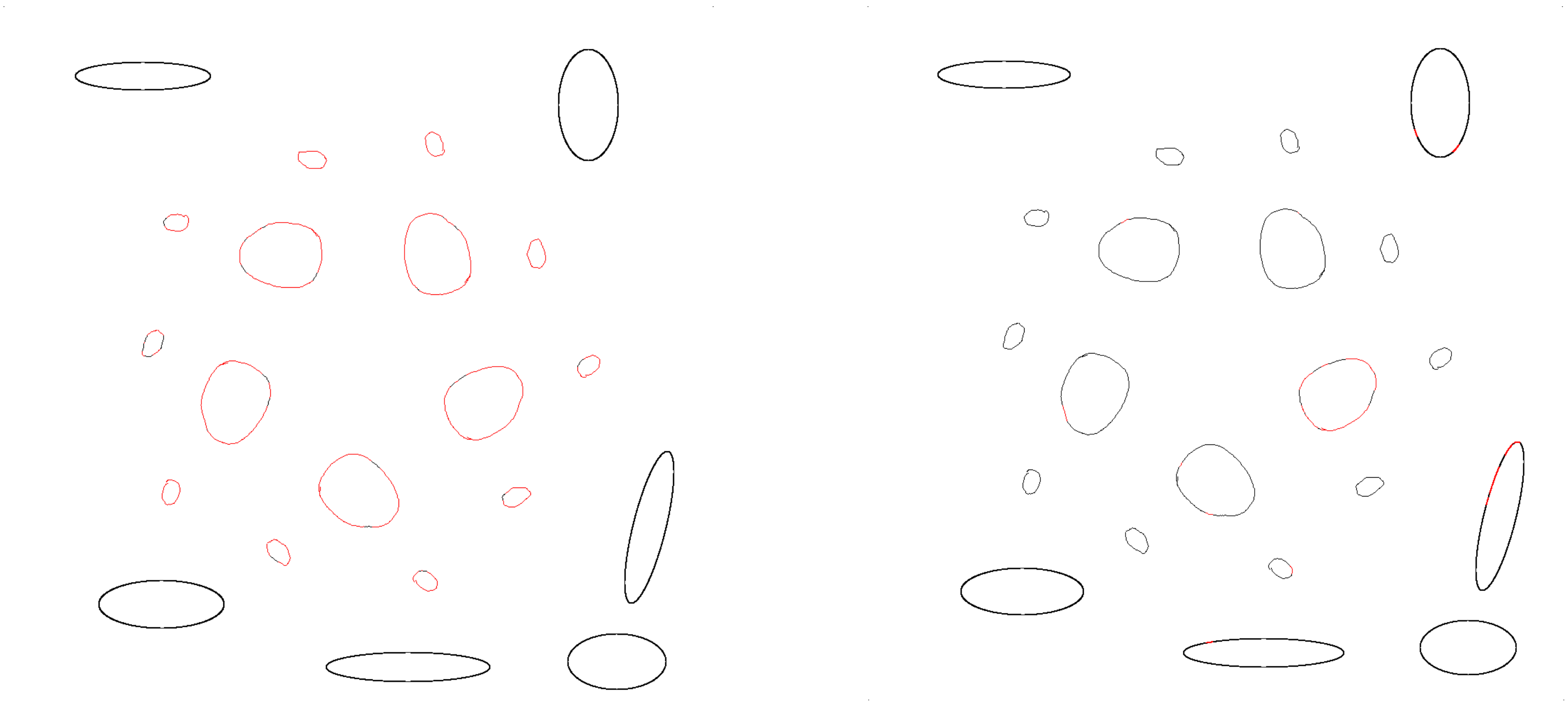
- Left: scene (black) and interesting voxels (yellow)
- Right: centres of detected symmetries (darker is stronger!!!).





Concept 2 (2D rotational): test 4

- 1st (step 5) and one of the irrelevant remaining symmetries.



Future work

- Completion of Concept 2 tasks
- Implementation of the initial concept (based on line segments)
- Optimisation of both concepts (e.g. hierarchical voxelization)
- Multiple (shifted) voxelisations for 0.5 voxel size up, left, back. Additional challenge here is identification which symmetries from different voxelisations are actually the same.

Thanks

- *to UM FERI students Matic Rašl and Luka Lukač for implementation, discussion, ideas…*
- *to you for surviving this presentation.*