Lifting 2D object detections to 3D a geometric approach in multiple views

Monday, December 28, 2020 3:24 PM

Front end

Given the object detections, we use a tracking by detection method, (3D traffic scene understanding from movable platforms). to associate the bounding boxes.

- It computes a distance matrix using patch appearance and associate detections using the Hungarian method for bipartite matching.

We assume the object is bounded by a rectangular region B_i in the image i.

- In 3D, each B_i defines a semi-finite pyramid Q_i with its apex in the camera center.
- In two-view case, object's projections are bounded by rectangles B_1, B_2 in the image, the object in space must lie within a polyhedron D.
- D can be obtained by intersecting the two semi-infinite pyramids defined by the two rectangles B_1, B_2. and centers of projection C_1, C_2.
- In n-view case, the object is inside the polyhedron formed by the intersection of the n semi-infinite pyramids generated by the rectangles $B_1 \cdots B_n$:

$$D = Q_1 \cap Q_2 \cdots \cap Q_n.$$

$$D = \{ X \in \mathbb{R}^3 : \exists x_i \in B_i, i = 1 \cdots n, \text{ s.t. } \forall i : x_i = \pi_i(X) \}$$

π is the known perspective projection onto the i-th image.

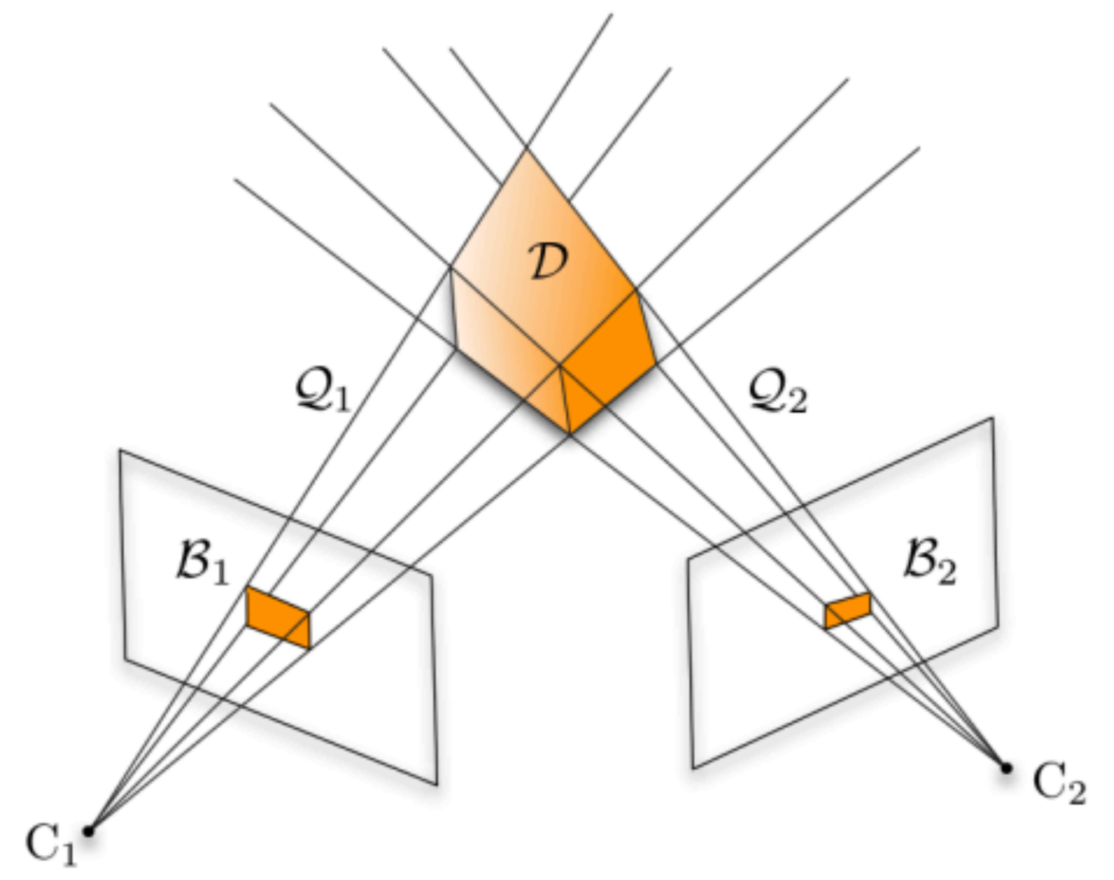

Fig. 1. Bounding the object in 3D from 2D detections. Here a graphical example with two images, where the semi-infinite pyramid is defined from the centre of projection and the bound $\boldsymbol{B}_i$.

Vetex enumeration solution.

The semi-infinite pyramid Q_i can be written as the intersection of the four negative half spaces $H_1^i, H_2^i, H_3^i, H_4^i$, defined by its supporting planes.

$\therefore$ The solution set D can be expressed as the intersection of $4n$ negative half-spaces:

$$D = \bigcap_{\substack{i=1 \cdots n \\ \ell = 1 \cdots 4}} H_\ell^i \quad \text{(H-representation)}$$

However, we aim at an explicit description of D in terms of vertices and edges, also called a V-representation.

The problem of producing a V-representation from an H-representation is called the vertex-enumeration problem. in Computational geometry. (CG)

- this approach is called the CG approach. We also bound the size using maximum object size from Shapenet, called the CGb method.
- the solution set can be enclosed with an axis-aligned box using Interval Analysis, dubbed IA approach.

Interval analysis

- Underscores and overscores will represent lower and upper bounds of intervals.
- $\mathbb{IR}$ stands for the set of real intervals.
- If $f(x)$ is a function defined over an interval x then range(f, x) denotes the range of $f(x)$ over x.
- A natural interval extension of f is $f(x)$, obtained by replacing variables with intervals
  - Eg. $f_1(x) = x^2 - x$, and $f_2(x) = x(x-1)$ are natural interval extensions of the same function.

Interval based triangulation.

Assume we can write a closed form expression that relates the 3D point X to its projections $x_1 = \pi_1(X)$ and $x_2 = \pi_2(X)$ in two images

$$X = f(x_1, x_2).$$

If we let x_1, x_2 vary in B_1, B_2, then range$(f, B_1 \times B_2)$ describes the polyhedron D that contains the object.

Interval anaysis gives a way to compute an axis-aligned bounding box, containing D, by simply evaluating $f(x_1, x_2)$, the natural interval extension of f, with $B_1 = x_1$ and $B_2 = x_2$.

In summary, the IA approach yields a rectangular axis-aligned bounding box $f(x_1, x_2)$ that contains the polyhedron D.

This method is faster and easier to implement using interval arithmetic library (eg INTLAB). than the CG, but the enclosure is an overestimate.