ETH Zürich
Mathematical Foundations of
Computer Graphics and Vision
Spring 2019

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

EXERCISE 2

GLOBAL OPTIMIZATION

In this exercise, branch and bound algorithm for consensus set maximization is implemented. The context which the algorithm is applied is the detection of the inlier correspondences between two aerial images.

To run the code, *../code* must be the working directory. Then, running *main.m* performs all the necessary calculations required, outputs the found inliers and outliers to the command window and generates plots.

1. Reformulation of the Problem for Linear Programming

- The set S is the input data, inlier-set is $S_I \subseteq S$ and outlier-set is $S_O = S \setminus S_I$
- The translational model $\Theta = (T_x, T_y)$
- The point correspondences are (p_i, p_i') where $p_i = (x_i, y_i)$ and $p_i' = (x_i', y_i')$
- The threshold for inlier-outlier discrimination is δ

Consensus set maximization problem in our context can be expressed as:

$$\max_{S_I,\Theta} \quad card(S_I)$$

$$\boldsymbol{s.t.} \quad |x_i + T_x - x_i'| \le \delta, \forall i \in S_I \subseteq S$$

$$|y_i + T_y - y_i'| \le \delta, \forall i \in S_I \subseteq S$$

$$\underline{T_x} \le T_x \le \overline{T_x} \quad and \quad \underline{T_y} \le T_y \le \overline{T_y}$$

To be able to solve the problem using linear programming, expressions can rewritten and relaxed (z values) as follows:

$$\max_{\mathbf{z},\Theta} \quad \sum_{i=1}^{N} z_i$$

$$\boldsymbol{s.t.} \quad z_i|x_i + T_x - x_i'| \le z_i\delta$$

$$z_i|y_i + T_y - y_i'| \le z_i\delta$$

$$0 \le z_i \le 1$$

$$\forall i = 1 \dots N$$

$$\underline{T_x} \le T_x \le \overline{T_x} \quad and \quad \underline{T_y} \le T_y \le \overline{T_y}$$

By writing the constraints that have absolute values in open form and introducing auxiliary variables in place of bilinear terms ($w_{ix} = z_i T_x$ and $w_{iy} = z_i T_y$), all the constraints can be further rewritten as:

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

$$z_i(x_i - x_i' - \delta) + w_{ix} \leq 0$$

$$z_i(x_i' - x_i - \delta) - w_{ix} \leq 0$$

$$z_i(y_i - y_i' - \delta) + w_{iy} \leq 0$$

$$z_i(y_i' - y_i - \delta) - w_{iy} \leq 0$$

$$0 \leq z_i \leq 1$$

$$\forall i = 1 \ldots N$$

$$\underline{T_x} \leq T_x \leq \overline{T_x} \quad and \quad \underline{T_y} \leq T_y \leq \overline{T_y}$$

The bilinearities (occurring because of the multiplication of two constrains, z and T) in the formulation above, can be relaxed using concave and convex envelopes (only w_{ix} are shown, w_{iy} are similar):

$$w_{ix} \geq \max\left(\underline{z_i}T_x + \underline{T_x}z_i - \underline{z_i}\,\underline{T_x} \;,\; \overline{z_i}T_x + \overline{T_x}\,z_i - \overline{z_i}\,\overline{T_x}\right)$$

$$w_{ix} \leq \min\left(\overline{z_i}T_x + \underline{T_x}z_i - \overline{z_i}\,\underline{T_x} \;,\; \underline{z_i}T_x + \overline{T_x}z_i - \underline{z_i}\,\overline{T_x}\right)$$

If w is larger or equal to the max term, it is larger or equal to both (similarly for min term). Also, since the lower bound for every z value is 0 (means corresponding point pair is an outlier match) in our problem definition, the constraints can be rewritten as:

$$w_{ix} \geq \underline{T_x}z_i$$

$$w_{ix} \geq T_x + \overline{T_x}\,z_i - \overline{T_x}$$

$$w_{ix} \leq T_x + \underline{T_x}z_i - \underline{T_x}$$

$$w_{ix} \leq \overline{T_x}z_i$$

When we combine the constraints that we found earlier with the four constraint for w_{ix} and the four constraints for w_{iy} and put them in a nicer form which can be observed well as the parameters of the linear program, whole problem can now be expressed as:

$$\max_{\mathbf{z},\Theta} \quad \sum_{i=1}^{N} z_i$$

$$\mathbf{s.t.}$$

$$z_i(x_i - x_i' - \delta) + w_{ix} \leq 0$$

$$z_i(x_i' - x_i - \delta) - w_{ix} \leq 0$$

$$z_i\underline{T_x} - w_{ix} \leq 0$$

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

$$z_i\overline{T_x} - w_{ix} + T_x \leq \overline{T_x}$$

$$-z_i\underline{T_x} + w_{ix} - T_x \leq -\underline{T_x}$$

$$-z_i\overline{T_x} + w_{ix} \leq 0$$

$$z_i(y_i - y_i' - \delta) + w_{iy} \leq 0$$

$$z_i(y_i' - y_i - \delta) - w_{iy} \leq 0$$

$$z_i\underline{T_y} - w_{iy} \leq 0$$

$$z_i\overline{T_y} - w_{iy} + T_y \leq \overline{T_y}$$

$$-z_i\underline{T_y} + w_{iy} - T_y \leq -\underline{T_y}$$

$$-z_i\overline{T_y} + w_{iy} \leq 0$$

$$0 \leq z_i \leq 1$$

$$\forall i = 1 \ldots N$$

$$\underline{T_x} \leq T_x \leq \overline{T_x} \quad and \quad \underline{T_y} \leq T_y \leq \overline{T_y}$$

For the canonical form,

$$\min_{\boldsymbol{x}} f^T \boldsymbol{x}$$

$$s.t.$$

$$A\boldsymbol{x} \leq b$$

$$l_b \leq \boldsymbol{x} \leq u_b$$

$$\boldsymbol{x} = \left(\mathrm{z_1}, \ldots, \mathrm{z_n}, \mathrm{w_{1x}}, \ldots, \mathrm{w_{nx}}, \mathrm{w_{1y}}, \ldots, \mathrm{w_{ny}}, \mathrm{T_x}, \mathrm{T_y}\right)^T$$

$$l_b = \left(0, \ldots, 0, -\infty, \ldots, -\infty, -\infty, \ldots, -\infty, \underline{\mathrm{T_x}}, \underline{\mathrm{T_y}}\right)^T$$

$$u_b = \left(1, \ldots, 1, +\infty, \ldots, +\infty, +\infty, \ldots, +\infty, \overline{T_x}, \overline{T_y}\right)^T$$

$$f = (\underbrace{-1, \ldots, -1}_{\text{N times}}, 0, \ldots, 0)^T$$

As observed, f values that are multiplied with corresponding z values are -1 in order to convert original maximization problem into canonical minimization problem. For each point correspondence $\boldsymbol{A}_{\boldsymbol{12n}\,\mathrm{x}\,(\boldsymbol{3n+2})}$ matrix is constructed from the coefficients of the inequality constraints annotated as blue, red and green above. Finally, for each point correspondence $\boldsymbol{b}_{\boldsymbol{12n}\,\mathrm{x}\,\boldsymbol{1}}$ is constructed from orange annotated values. This section is implemented in *solveLP.m*.

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

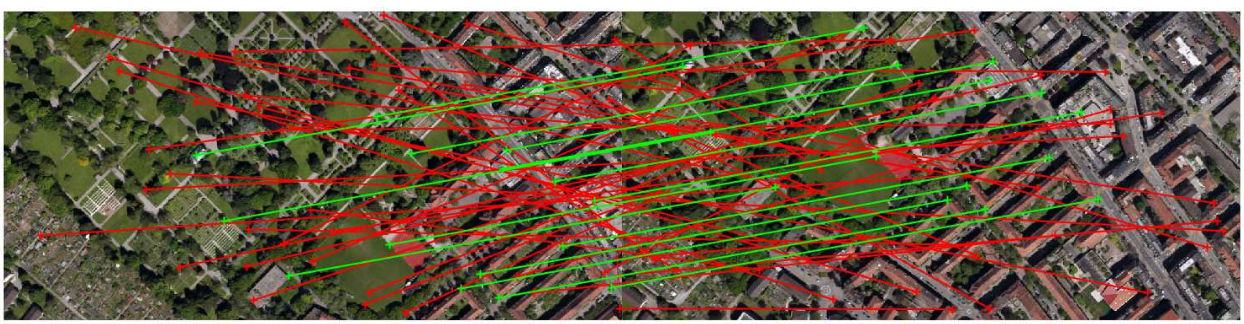
2. Branch and Bound Implementation

Problems (i.e. subspaces) are stored in a struct which consists of five list structures as suggested in the exercise sheet.

At each iteration:

- The best candidate is found (*findBestCandidate.m*) and considered for termination or branching.
    - To find the best candidate, a comparison scheme is implemented in *compareBounds.m*.
    - While two candidates are being compared:
        - The one with the larger upper bound is better.
        - If upper bounds are equal, the one with the larger lower bound is better.
        - If both bounds are equal, the one that placed later in the list is better (implicit DFS).
- If the termination criterion is not satisfied for the candidate being considered, it is divided into two branches (*branchAndSolve.m*).
    - Current candidate is removed from the problem list (*removeProblem.m*)
    - Each new branch is solved with linear programming (*solveLP.m*) to find the upper bound for the number of inliers.
    - Each new branch is tested with the translation model that linear programming finds in order to set the lower bound for the number of inliers (*testModel.m*)
    - Each new branch is added to the problem list (*addNewProblem.m*)
- Problem list is iterated and the subspaces that definitely do not contain the optimal solution are removed from the list (*removeBadCandidates.m*)

3. Results



For δ = 3, the globally optimal solution for the translational model found by my implementation is,

$$\widehat{\Theta} = \left(\hat{T}_x, \hat{T}_y\right) = (-232.0, -154.0)$$

According to this translational model, found inlier matches are indicated with green and outliers are indicated as red in the above image. The number of inliers are 15.

Inlier indices: 3, 8, 9, 15, 16, 20, 26, 31, 32, 34, 35, 40, 42, 45, 51

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

The convergence of the bounds is shown in the below graph. Note that the algorithm terminates in 6th iteration as the difference between lower bound and the upper bound for the number of inliers becomes smaller than 1. As observed, at the time of convergence, the lower bound for the cardinality of the inlier set is 15 and the upper bound is 15.88.