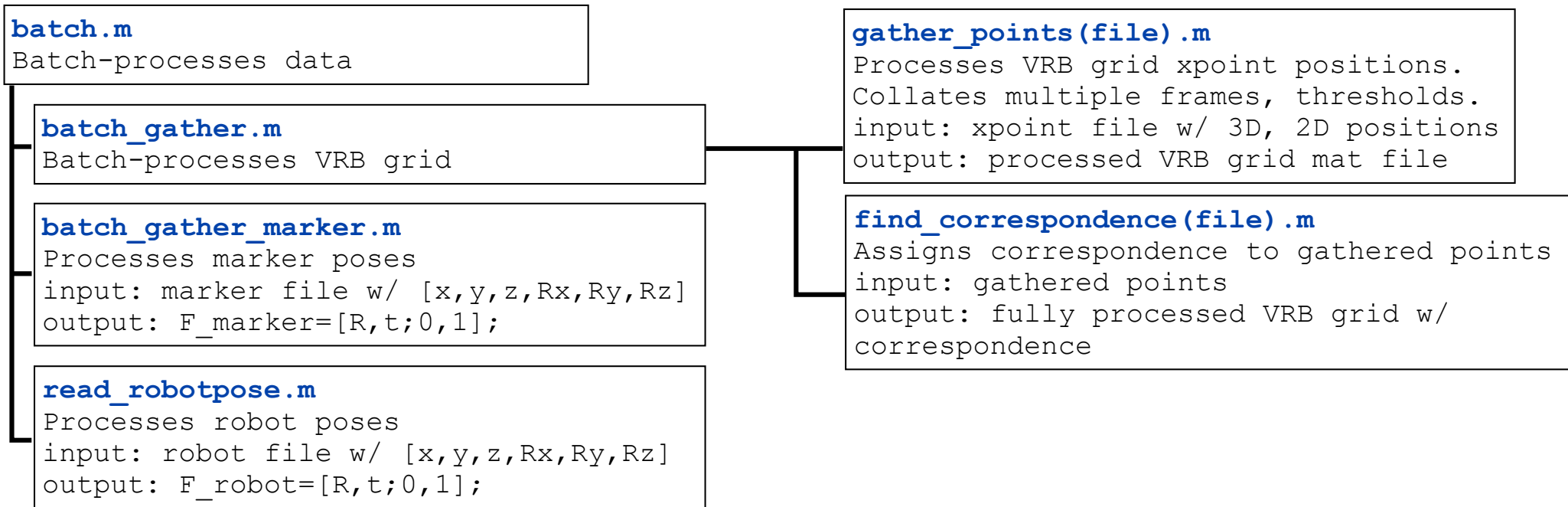


# 1. Experiment

- 1. Hardware setup: MT, UR5, VRB Sotware setup: copy `./CIS2_CODES/Robot/Robot_client/*.urp` to UR5 controller
- 2. Run software on computer
  - 1. MT : `./CIS2_CODES/MicronTracker/MTDemoCPP_dslee/Win32/Debug/MTDemoCPP.exe`
  - 2. UR5: `./CIS2_CODES/Robot/Computer_server/get_pose_serverV4.py`
- 3. Run experiment
  - 1. Run on UR5 controller: `./CIS2_CODES/Robot/Robot_client/*.urp` % Starts the motion trajectories
  - 2. Robot will move and stop at a waypoint, and wait for some seconds.
    - On the MicronTracker window, press 0 to record for same trajectory, 1 for a new trajectory.
  - 3. Repeat 1 and 2 to collect data.

# 2. Data processing



### 3. Calibration

**testCalib.m**

Calibrates VRB. Provided by Alexis

### 4. Analysis

**analyze.m**

Analyzes data

**plane\_fit(points).m**

Fits VRG grid pattern to a interpolated plane.

input: raw VRG grid

output: VRB grid projected onto fitted plane;

**registerPointsV3.m**

Recovers VRB pose (Alexis' code)

**get\_invariants.m**

Calculates t and theta invariants

input: pose

output: t, theta;

**fitplane.m**

**ransacfitplane.m**

**ransac.m**

**iscolinear.m**

RANSAC plane fitting algorithm, taken from web

**projection(a,b).m**

Projects vector a on b

Input: vectors a and b

output: projected vector a on b

**subToSuper2(subposes).m**

Finds transformation between two poses

**param\_extract(motion).m**

Extracts t and theta invariants