

Track 4: “Indoor Mobile Robot Positioning” – Practical issues

INDEX

1	PRESENTATION	2
2	SYNCHRONIZATION	2
2.1	MANUAL SYNCHRONIZATION.....	2
2.2	AUTOMATIC SYNCHRONIZATION.....	2
3	SCORING PROCEDURE	3

1 Presentation

This document describes some of the practical procedures that will be used during the *IPIN 2016 Indoor Competition-Track 4: Indoor mobile robot positioning* that is going to be held during the IPIN 2016 Conference, 4-7 October 2016, at the School of Engineering of the University of Alcalá, in Alcalá de Henares, Madrid (Spain).

2 Synchronization

It is important to note that an extremely precise timing information will not be required, given the scoring approach that will be used in the competition, so that time shifts in the milliseconds or centiseconds range can be perfectly accepted.

In order for the participants to be able to generate their time labels, we provide two synchronization mechanisms to define the “start time” of the robot:

2.1 Manual synchronization

In the competition area there will be a screen showing a visual and audible countdown (starting at 5 seconds). When the countdown reaches zero, the robot will start moving. So, the participants can use the zero count event to manually generate a “start command” for their position logging module (the command can be issued by any means: manual switch, remote signal sent from a participant’s device, etc.).

In <http://www.geintra-uah.org/archivos/video/track4-countdown.wmv> we provide a video showing the countdown application running.

2.2 Automatic synchronization

The countdown described above will also be controlling a “timing start” server that can be polled by the participants’ systems to check if the robot started moving. The server will be a simple socket based application listening to a given port in a given IP address (that will be provided by the organizers).

The protocol is very simple:

- The server will be listening at a given IP:port, waiting for a connection from the participant’s client.
- When the connection is established, the server will continuously wait for any data sent from the client (sending just one byte will be enough).
- For every data chunk sent, the server will send back a string that can be “0” if the robot is stopped, or “1” if the robot started to move.
- The polling frequency can be as high as the participant’s client application wish, with the only limitations due to the data transfer rate given the size of the data sent and received.

To ease developing the client, we provide participants with the source code of the server, the countdown application, and a client example. The code is known to compile and run properly in Ubuntu based systems. Even though we make no provision for code working in windows environments, the client-server approach is standard and simple enough, so that developing a client in windows is not expected to be complex.

To ease debugging and testing the client application, the organizers will allow the participants to remotely test their clients, by running the server application the week before the competition, within an agreed schedule with each of the participants.

If your team is willing to use the automatic synchronization approach, please contact Javier Macías-Guarasa at macias@depeca.uah.es to request the source code and, if necessary, to agree on the time slot for remotely testing the server.

3 Scoring procedure

Participants have to report the trajectory: robot position and timestamp at least every 0.1 seconds (minimum frame-rate). All measurements have to be reported together with a maximum delay of 2 minutes after the end of the team participation. Measurements presented after the deadlines will not be considered. The measurement report must be a plain text file (*.txt), with three columns, as follows:

- position x and y coordinates in millimeters with respect to the initial robot position
- timestamp in milliseconds with respect to the movement starting time

Each line in the text file will correspond to a new estimated position

An example, in which the results are given every 5ms, is shown below.

X coordinate (millimeters)	Y coordinate (millimeters)	Timestamp (milliseconds)	
00000.0	00000.0	000000	
00025.0	00000.0	000050	→ Position estimation 1
00050.0	00000.0	000100	→ Position estimation 2
00075.0	00000.0	000150	
00100.0	00001.4	000200	
00124.8	00004.2	000250	
00149.5	00008.4	000300	
00173.8	00014.0	000350	
00197.8	00020.9	000400	
00221.4	00029.1	000450	
00244.5	00038.7	000500	
00267.1	00049.5	000550	
00288.9	00061.6	000600	
00310.1	00074.9	000650	
00330.5	00089.4	000700	
00350.0	00105.0	000750	

The evaluation will be based on the differences between the ground truth trajectory and the one obtained by the participants. The metric will be computed as the third quartile of the localization errors. The localization error is the Euclidean distance between the estimated positions and the ground truth trajectory. Fig. 1 shows an example with a fictitious round shape trajectory for the robot (note that the real path followed by the robot is not available in advance). The line in blue indicates the ground truth, that is, the real trajectory followed by the robot. The black crosses are the positions estimated by the participant, and in green the Euclidian distance to the ground truth appears. Note that temporal information is taken into account (past and future estimated positions) so if an outlier appears the Euclidian distance is computed to the correct section of the ground truth trajectory. Final scores are computed considering the third quartile (75th percentile), with that purpose the CDF is drawn and the error at 75% is considered as the accuracy metric (see Fig. 2 for an example, where an error of 97mm is considered for comparison with other teams). In case of tie, the error at 50% of the CDF will be evaluated.

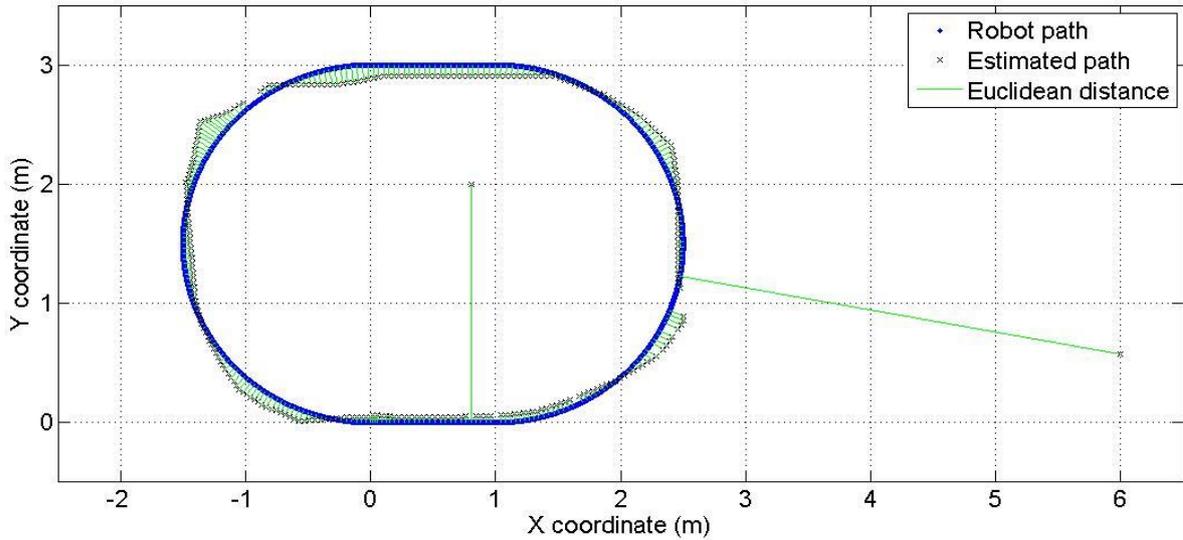


Fig. 1.- Results with the robot in motion.

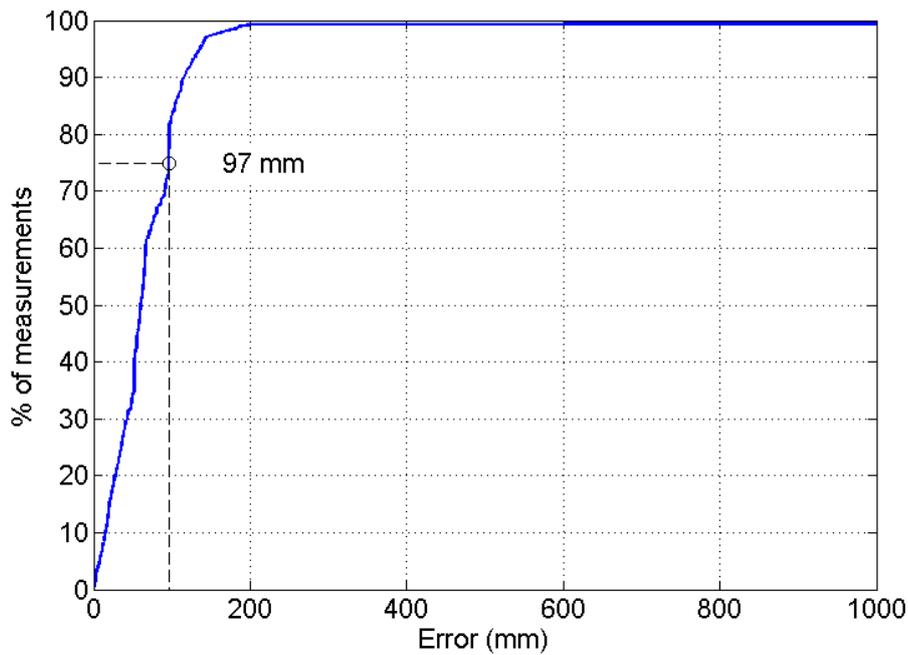


Fig. 2.- CDF of Euclidian distance errors between estimated positions and the ground truth.

NOTE 1: Remember that at the beginning of each participant’s turn, competitors will have 30 minutes for the system set-up (notice, that the positioning systems of the competitors can only be installed onboard the robot-without interaction with the mobile robot systems- or in the provided aluminum profiles).

NOTE 2: Three positions will be provided during the set-up stage so the participants can use them to calibrate their sensorial system. One of the given positions will be the (0,0) coordinates of the system, that coincides with the initial position of the robot. The other two positions are not necessary in the trajectory of the robot. One example of the positions that could be provided for calibration purposes is: P1(0,0)mm, P2(2000,0)mm and P3(0,2000)mm.