Changing the Degrees of Freedom for a particle filter tracking algorithm based on prior knowledge

Koen Snijder - K.Snijder-1@student.tudelft.nl

Supervisor: Kees Kroep

Responsible Professor: Dr. Ranga Rao Venkatesha Prasad

1. Background

Tactile Internet (TI) aims to revolutionize the online interaction scene by providing a new form of interaction through touch by using haptic feedback over a network.

However, If the **latency is more than 1 ms**, the user might experience effects like motion sickness, or the operations might be disrupted [1].

To circumvent the delay, a solution has been found by interaction with **local simulations** of the **physical reality.** See Figure 1.

Since these simulations will have to simulate a **dynamic environment**, it is vital to create a tracking system that can keep track of all the moving objects. The tracking is done using a **particle filter** algorithm.

A particle filter **predicts** the movement of an object in **6 Degrees of Freedom (DoF)**, but this is **computationally expensive.**

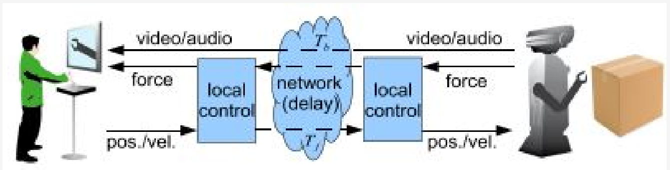

Figure 1: Interaction with local simulations [2].

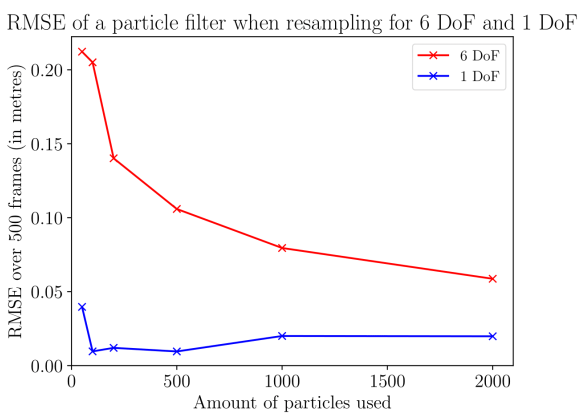

Figure 2: RMSE of a particle filter while tracking in 6 DoF and 1 DoF.

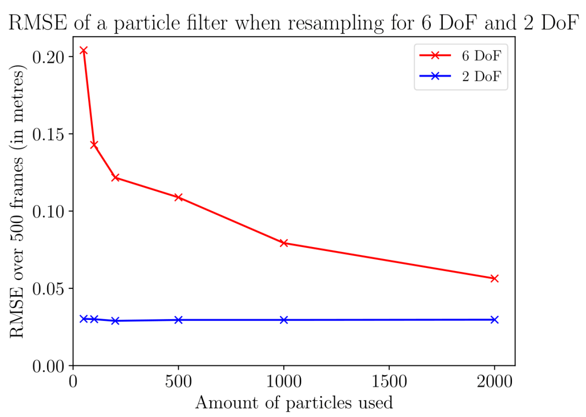

Figure 3: RMSE of a particle filter while tracking in 6 DoF and 2 DoF.

2. Research Questions

- How to tune a particle filter based on prior knowledge of an object's movement?
- What is the advantage of a tuned particle filter of a standard particle filter?

3. Methodology

- Read literature to understand a particle filter mathematically.
- Defined 3 experiments to perform:
  - Translation over one axis,
  - Translation over two axes,
  - Translation over two axes with rotation.
- Find out how to tune the particle filter and change the algorithm.
- Change Degrees of Freedom (DoF) of particle filter based on the specific experiment.
- Compare performance of standard particle filter with 6 DoF to performance of tuned particle filter with less DoF.

4. Results

The Root Mean Square Error (RMSE) from tuning a particle filter for 1 DoF and 2 DoF in comparison with 6 DoF can be seen in **Figure 2** and **Figure 3.**

The average execution times for this can be seen in **Table 1** for 1 DoF and **Table 2** for 2 DoF.

5. Future Work

- Perform experiment with translation and rotation.
- For the results, take an average of multiple runs.
- Fix bugs in tracking code:
  - Algorithm predicts axis with an offset,
  - Rotations are not predicted accurately.
- Upgrade tracking algorithm with GPU acceleration.

6. Conclusion

Tracking with a tuned particle filter is advantageous as it gains a massive speedup by at least a factor of **36** for the most similar tracking accuracy, where the tuned filter is approximately **1.5** times more accurate than the standard particle filter.

Amount of Particles	Average execution time per frame for 6 DoF (ms)	Average execution time per frame for 1 DoF (ms)
50	182.94	134.23
100	351.08	258.64
200	695.46	352.03
500	1705.79	352.87
1000	3199.71	388.32
2000	4885.41	413.99

Table 1: Average execution time per frame for each amount of particles for tracking in 1 DoF.

Amount of Particles	Average execution time per frame for 6 DoF (ms)	Average execution time per frame for 2 DoF (ms)
50	132.58	133.93
100	272.50	259.46
200	487.33	503.87
500	1165.57	1191.07
1000	2613.64	1963.42
2000	4963.51	1820.59

Table 2: Average execution time per frame for each amount of particles for tracking in 2 DoF.

References

[1]. Daniel Van Den Berg, Rebecca Glans, Dorian De Koning, Fernando A. Kuipers, Jochem Lugtenburg, Kurian Polachan, Prabhakar T. Venkata, Chandramani Singh, Belma Turkovic, and Bryan Van Wijk. Challenges in haptic communications over the tactile internet. IEEEAccess, 5:23502–23518, 2017

[2] .Xiao Xu, Burak Cizmeci, Clemens Schuwerk, and Eckehard Steinbach. Model-mediated teleoperation: Toward stable and transparent teleoperation systems. IEEE Access, 4:425–449, 2016.