Supplementary Results

To show the capability of OpenLPT, we aim at comparing results of OpenLPT at a high image density up to 0.125 ppp with those published in the original STB paper by Schanz et al. (2016). In particular, two figures, Fig.S1 and Fig.S2, are shown to illustrate the percentage of ghost particles ($F_{g(tot)}$) and the position uncertainty (Δ_p) as functions of time (in frames). These two quantities are key to evaluate the convergence time of the code at a high image density. Although we rarely conduct experiments at this extremely-high concentration (>0.1 ppp), it has been proven crucial to test the robustness of the code.

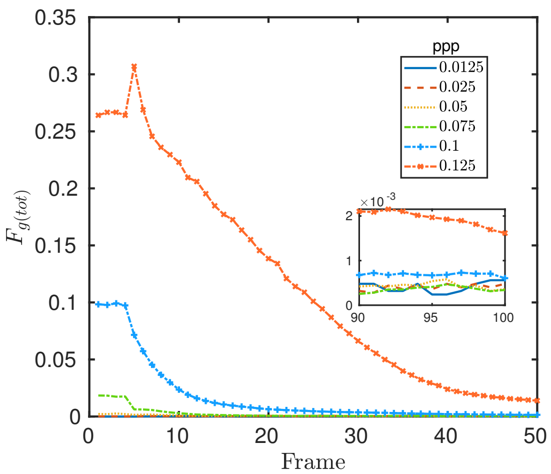

Fig. S1: Total number of reconstructed ghost particles (tracked and untracked) relative to the number of true particles.

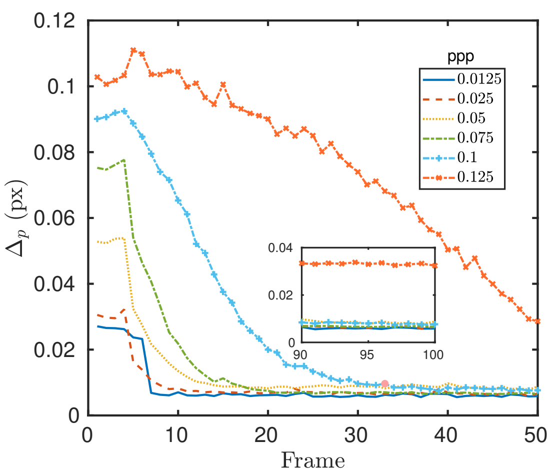

Fig. S2: 3D positional error in pixel, averaged over all tracked particles relative to the true particle position.

In figure S1, compared with figure 5 in Schanz et al. (2016), the percentage of ghost particles starts at 0.3 for the particle image density at 0.125 ppp. Compared with $F_{g(tot)} = 4.4$ in Schanz et al. (2016), the percentage of ghost particles drops by a factor of thirteen, which shows a significant improvement of reducing the number ghost particles during IPR. In addition, the converged value of $F_{g(tot)}$ is also about half of that in Schanz et al. (2016). Other than these two improvements, we found two differences between our results and figure 5 in (Schanz et al., 2016). First, the convergence time for OpenLPT is longer than that in the original STB. Second, $F_{g(tot)}$ decreases even in the first four frames in figure 5 of Schanz et al. (2016), whereas it remains constant in our case. The exact reason for these differences is unclear, which might be attributed to different implementations of the 2D particle identification and the velocity field calculations.

In addition, as shown in figure S2, the converged position error of particles from OpenLPT is larger than the value reported in Schanz et al. (2016). This difference can be attributed to the difference in the window size used. The window size determines an area that the particle image projection and residual image calculation were performed. In our implementation, we choose to use a window size that is double the particle size, whereas it was chosen as the same size in Schanz et al. (2016).