OpenFLANN Speed Comparison

Hyungtae Lim (shapelim@kaist.ac.kr)

April 16, 2021

Table 1: `nearestKSearch()` speed test: # of input cloud is **10,000**.

K (# Nearest points)	Algorithm	`setInputCloud()` [s]	`KNN()` [ms]	Total [s]
1	PCL	0.00121	0.00283	0.00122
	Nano	**0.00106**	**0.00095**	**0.00106**
	Pico	0.00116	0.00133	0.00117
10	PCL	0.00121	0.00366	0.00122
	Nano	**0.00106**	**0.00189**	**0.00107**
	Pico	0.00116	0.00325	0.00117
500	PCL	0.00121	0.05975	0.00127
	Nano	**0.00106**	0.12101	**0.00118**
	Pico	0.00116	**0.05944**	0.00122
1,000	PCL	0.00123	**0.10606**	0.00134
	Nano	**0.00106**	0.38239	0.00145
	Pico	0.00116	0.10745	**0.00127**
5,000	PCL	0.00123	**0.46643**	0.00169
	Nano	**0.00106**	6.07235	0.00713
	Pico	0.00116	0.48517	**0.00165**

Table 2: `nearestKSearch()` speed test: # of input cloud is **30,000**.
(30,000 is similar to the number of cloud points that are acquired by Velodyne-16.)

K (# Nearest points)	Algorithm	`setInputCloud()` [s]	`KNN()` [ms]	Total [s]
1	PCL	0.00435	0.00318	0.00436
	Nano	**0.00371**	**0.00105**	**0.00371**
	Pico	0.00387	0.00147	0.00387
10	PCL	0.00436	0.00418	0.00436
	Nano	**0.00372**	**0.00209**	**0.00372**
	Pico	0.00388	0.00347	0.00388
500	PCL	0.00436	**0.0606**	0.00442
	Nano	**0.00372**	0.12197	**0.00384**
	Pico	0.00387	0.06198	0.00393
1,000	PCL	0.00437	**0.1276**	0.00449
	Nano	**0.00372**	0.4623	0.00418
	Pico	0.00388	0.12812	**0.00401**
5,000	PCL	0.00437	**0.55483**	0.00492
	Nano	**0.00372**	7.48272	0.01121
	Pico	0.00388	0.57286	**0.00445**

Table 3: `nearestKSearch()` speed test: # of input cloud is **100,000**.
(100,000 is similar to the number of cloud points that are acquired by Velodyne-64.)

K (# Nearest points)	Algorithm	`setInputCloud()` [s]	`KNN()` [ms]	Total [s]
1	PCL	0.01758	0.00881	0.01759
	Nano	0.01548	**0.00188**	0.01549
	Pico	**0.01536**	0.0023	**0.01536**
10	PCL	0.01683	0.00869	0.01684
	Nano	0.01484	**0.00282**	0.01485
	Pico	**0.01466**	0.00444	**0.01466**
500	PCL	0.01717	0.07967	0.01725
	Nano	0.01513	0.15086	0.01528
	Pico	**0.015**	**0.07698**	**0.01508**
1,000	PCL	0.01683	0.13835	0.01697
	Nano	0.01488	0.48827	0.01536
	Pico	**0.01462**	**0.13789**	**0.01475**
5,000	PCL	0.01684	**0.67635**	0.01751
	Nano	0.01487	9.92816	0.02479
	Pico	**0.01463**	0.69405	**0.01532**

Table 4: `nearestKSearch()` speed test: # of input cloud is **200,000**.

K (# Nearest points)	Algorithm	`setInputCloud()` [s]	`KNN()` [ms]	Total [s]
1	PCL	0.0375	0.00923	0.03751
	Nano	0.03389	**0.00192**	0.03389
	Pico	**0.03152**	0.00238	**0.03152**
10	PCL	0.03718	0.0101	0.03719
	Nano	0.03355	**0.00337**	0.03356
	Pico	**0.03117**	0.00498	**0.03117**
500	PCL	0.03758	0.08011	0.03766
	Nano	0.03383	0.1457	0.03397
	Pico	**0.03147**	**0.07795**	**0.03155**
1,000	PCL	0.03733	**0.15182**	0.03748
	Nano	0.03369	0.51572	0.03421
	Pico	**0.03129**	0.15219	**0.03144**
5,000	PCL	0.03736	**0.72204**	0.03808
	Nano	0.03367	10.06227	0.04373
	Pico	**0.0313**	0.76547	**0.03207**

Table 5: `radiusSearch()` speed test: # of input cloud is **10,000**.

Radius [meter]	Algorithm	`setInputCloud()` [s]	`radiusSearch()` [ms]	Total [s]
1.0	PCL	0.00126	0.00383	0.00127
	Nano	**0.00106**	**0.00092**	**0.00106**
	Pico	0.00121	0.00094	0.00121
10.0	PCL	0.00123	0.00617	0.00123
	Nano	**0.00104**	**0.00105**	**0.00104**
	Pico	0.00118	0.00502	0.00118
20.0	PCL	0.00127	0.01959	0.00129
	Nano	**0.00106**	**0.00146**	**0.00106**
	Pico	0.00121	0.01893	0.00123
40.0	PCL	0.0013	0.09533	0.00139
	Nano	**0.00107**	**0.00224**	**0.00108**
	Pico	0.00123	0.09183	0.00132
80.0	PCL	0.00123	0.37804	0.00161
	Nano	**0.00104**	**0.00289**	**0.00104**
	Pico	0.00118	0.33893	0.00152

Table 6: `radiusSearch()` speed test: # of input cloud is **30,000**.

Radius [meter]	Algorithm	`setInputCloud()` [s]	`radiusSearch()` [ms]	Total [s]
1.0	PCL	0.00444	0.00602	0.00445
	Nano	**0.0037**	0.00128	**0.0037**
	Pico	0.00402	**0.00126**	0.00402
10.0	PCL	0.00447	0.01518	0.00449
	Nano	**0.00371**	**0.00213**	**0.00371**
	Pico	0.00405	0.0122	0.00406
20.0	PCL	0.00435	0.04918	0.0044
	Nano	**0.00364**	**0.00208**	**0.00364**
	Pico	0.00393	0.04979	0.00398
40.0	PCL	0.00436	0.26981	0.00463
	Nano	**0.00361**	**0.00316**	**0.00362**
	Pico	0.0039	0.26318	0.00416
80.0	PCL	0.00438	1.17727	0.00556
	Nano	**0.00362**	**0.00617**	**0.00363**
	Pico	0.0039	1.1047	0.005

Table 7: `radiusSearch()` speed test: # of input cloud is **100,000**.

Radius [meter]	Algorithm	`setInputCloud()` [s]	`radiusSearch()` [ms]	Total [s]
1.0	PCL	0.0174	0.00922	0.01741
	Nano	**0.01488**	0.00204	**0.01489**
	Pico	0.01525	**0.00191**	0.01525
10.0	PCL	0.01683	0.03263	0.01686
	Nano	**0.01441**	**0.00309**	**0.01441**
	Pico	0.01466	0.02945	0.01468
20.0	PCL	0.01682	0.16717	0.01698
	Nano	**0.01441**	**0.00488**	**0.01442**
	Pico	0.01465	0.16144	0.01482
40.0	PCL	0.01684	0.99892	0.01784
	Nano	**0.0144**	**0.00925**	**0.01441**
	Pico	0.01466	0.94366	0.0156
80.0	PCL	0.01679	4.45371	0.02124
	Nano	**0.0144**	**0.02098**	**0.01442**
	Pico	0.01466	4.21055	0.01887

Table 8: `radiusSearch()` speed test: # of input cloud is **200,000**.

Radius [meter]	Algorithm	`setInputCloud()` [s]	`radiusSearch()` [ms]	Total [s]
1.0	PCL	0.03795	0.01066	0.03796
	Nano	0.03294	**0.00237**	0.03294
	Pico	**0.03205**	0.00243	**0.03205**
10.0	PCL	0.03808	0.05865	0.03814
	Nano	0.03313	**0.00513**	0.03313
	Pico	**0.03209**	0.05942	**0.03215**
20.0	PCL	0.03911	0.33827	0.03945
	Nano	0.03373	**0.00906**	0.03374
	Pico	**0.03272**	0.36185	**0.03308**
40.0	PCL	0.03959	2.14774	0.04174
	Nano	0.03394	**0.02237**	**0.03396**
	Pico	**0.03349**	2.46971	0.03596
80.0	PCL	0.04018	9.76639	0.04995
	Nano	0.03446	**0.05849**	**0.03452**
	Pico	**0.03386**	10.69112	0.04455