

Table 1: Query Times on Random Queries to ImageNet Subset.

Parameters	Maximum Query Time (s)	Average Query Time ( $s \times 10^{-6}$ )
Uniform	0.0013	7.37
$\rho = 1, T = 3000, \beta = 0.68$	0.00046	13.2

Table 2: Success Probability on Random Queries to ImageNet Subset.

Parameters	Bottom 10th Percentile Succ. Prob.	Average Success Prob.
Uniform	0.275	0.621
$\rho = 1, T = 3000, \beta = 0.68$	0.576	0.772

Table 3: Query Time for Random Queries to MNIST Subset.

Parameters	Maximum Query Time (s)	Average Query Time ( $s \times 10^{-5}$ )
Uniform	0.0012	4.03
$\rho = 1, T = 3000, \beta = 0.68$	0.00048	1.10
$\rho = 0.83, T = 3000, \beta = 0.68$	0.0011	1.29
$\rho = 0.67, T = 1600, \beta = 0.88$	0.037	2.30
$\rho = 0.25, T = 1600, \beta = 0.88$	0.0033	2.15
$\rho = 0.1, T = 1600, \beta = 0.88$	0.0027	3.06

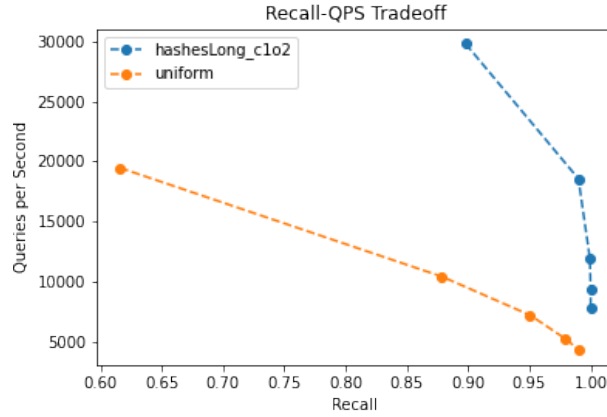


Figure 1: Recall/Query-per-Second tradeoff for a subset of the MNIST dataset

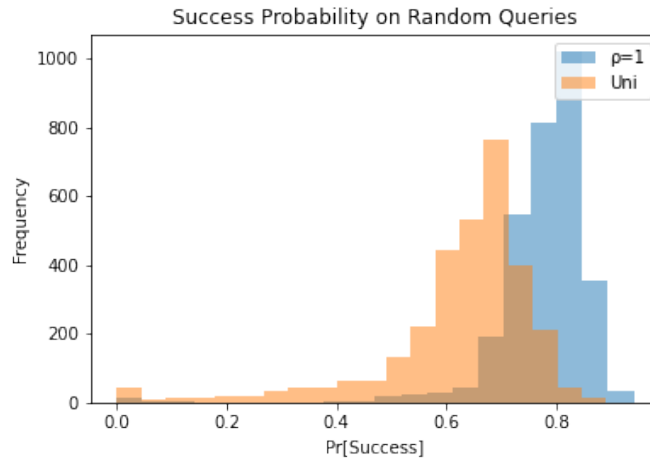


Figure 2: Success Probability for Random Queries to the ImageNet Dataset

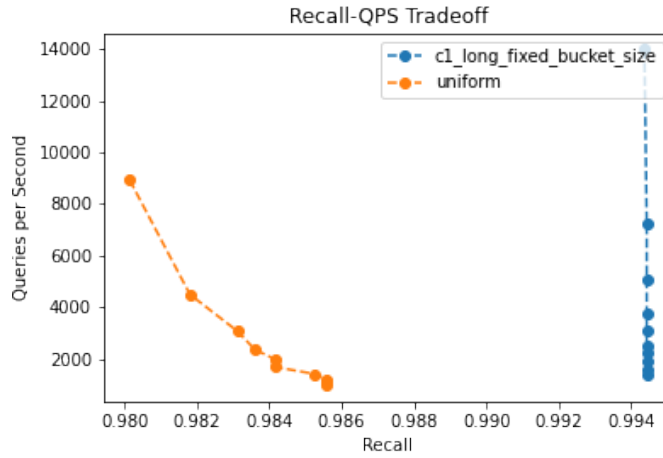


Figure 3: Recall/Query-per-Second tradeoff for a subset of the ImageNet dataset

## **Experiment Descriptions:**

In this document, we present a series of experiments extending those in the original submission. For one extension, we performed the same experiments as in the original paper on a subset of the ImageNet dataset. In particular, we extracted the first 624 images of the 3x8x8 validation subset from <https://image-net.org/index.php>, flattened the image to 1x192, and binarized the dataset with a threshold (i.e. all pixel values below a threshold of 16 were set to 0, and the complement is set to 1). We then ran our algorithm with stopping condition  $C=5$ , radius  $r=5$ , and the specified parameters in the tables above. The success probabilities are presented in Table 2 and Figure 2

On querying, we measured the time until the near neighbor was returned for a given query/NN pair using the “time” library for Python. We recorded the maximum and average query times in Tables 1-3 above for ImageNet and MNIST.

Further, we constructed Recall/Query-rate tradeoff curves for both datasets (as in <http://ann-benchmarks.com/index.html#datasets>). Recall was measured as the fraction of near neighbors that were recovered for a given query with a fixed number of trees. Queries-per-second was calculated by taking the inverse of query time. These data are an average over 1500 and 3000 random queries for MNIST and ImageNet, respectively. Because the recall was too high to resolve differences among tree types for our settings, we increased  $r$  to 10 at query time for the ImageNet dataset, and  $r$  to 15 for the MNIST dataset.