CUDA-Accelerated k-Nearest Neighbors Performance Benchmarking with Distance Metric Extension and WebGPU Visualization

May 2025

Presented by

Enlai Yii

Apply data-parallelism and CUDA techniques to benchmark and optimize the k-Nearest Neighbors (k-NN) algorithm at scale.

- Extend support for multiple distance metrics
- Visualize k-NN behavior using WebGPU
- Compare CPU vs GPU performance

CUDA Variants

Knn_cuda_global
Naive GPU version

Knn_cuda_shared
(attempted) Optimized shared-memory variant

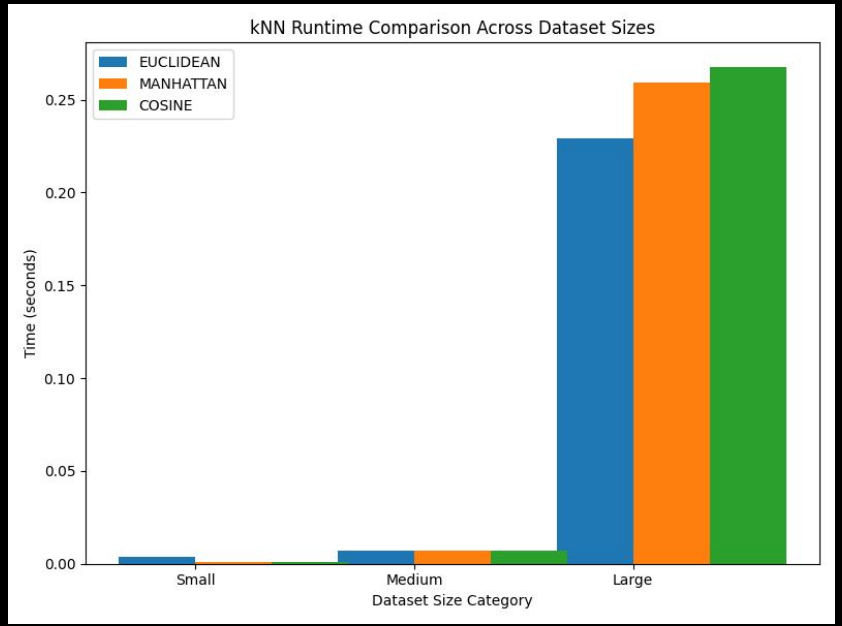
Knn_cuda_texture
Optimized memory-bound version

Knn_c
Baseline CPU Implementation

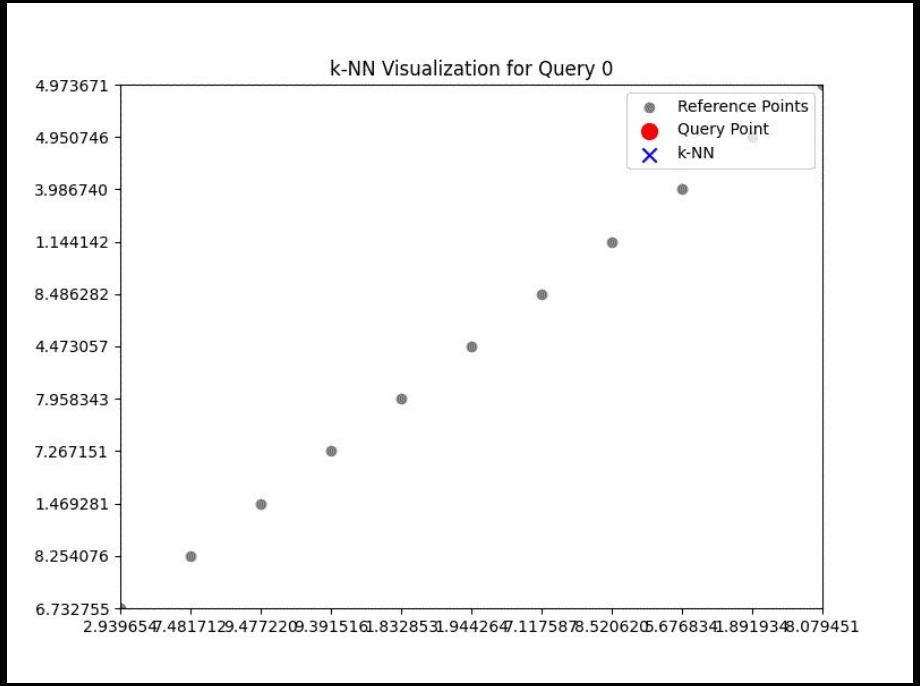
Knn_cudlas
Matrix-based variant using cuBLAS

Performance Results

- Up to 17,000× speedup for small datasets (cublas vs CPU)
- Texture memory reduced memory latency for repeated reads
- Cosine distance was the most expensive due to normalization

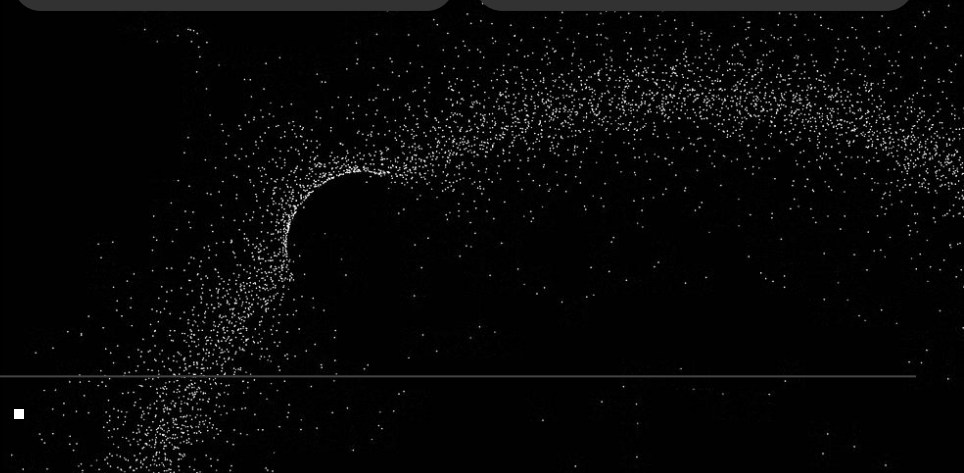


Real-Time WebGPU Visualization



Key Takeways

1. CUDA massively outperforms CPU for large-scale k-NN
2. Each implementation shows trade-offs in complexity and performance
3. Metric extensions and visualizations improved model transparency
4. Night profiling helped identify bottlenecks and validate optimizations



Thank you

This project implemented a full GPU benchmarking pipeline—combining core CUDA concepts, optimization strategies, and modern web graphics to deliver both high performance and high interpretability