

Neelansh Kaabra (nkaabra)
Doreen Valmyr (dvalmyr)

Project Milestone Report

Work Completed So Far:

As of November 29th, we have created our sequential KD Tree. When running the program, a user can specify which datasets they would like to build the KD Tree for as well as the number of features that they want to use. For example, each point in a dataset may contain 6 features, but we can specify that we only want to utilize the first 4 of those features. We also output helpful error messages if the compilation of the program fails. In addition to this sequential version of the KD Tree construction, we parallelized building the KDTree. We have achieved approximately a 4x speedup compared to the sequential implementation, but hope to increase this speedup further. This marks a significant milestone in enhancing the efficiency of the KDTree construction process.

We have also created a sequential KNN search algorithm that uses the KD tree to find the 'k' nearest neighbors of a specified target point.

Lastly, we have worked on lock free insertions for the KD Tree. This was developed to support multiple concurrent insertions using C++'s `<atomic>` library and the "compare and exchange" instruction. This addition, although initially unplanned, is a vital enhancement for ensuring the KD Tree's efficacy in parallel environments. As a result, we completed this stretch goal simultaneously while creating the KNN search algorithm.

Current Status and Next Steps:

The parallel KNN search model is currently in development, with the initial sequential structure in place. The immediate focus is on validating the accuracy of the search algorithm. Ensuring both correctness and performance efficiency remains a priority.

Upcoming Weeks: The plan is to finalize the Parallel KNN Search model and then move towards comprehensive benchmarking and testing. This phase will involve rigorous simulations to validate the correctness of all project components and to evaluate their performance against predefined standards.

Revised Goals and Deliverables:

- Finalization of Parallel KNN Search Model: This is an immediate priority. Efforts will be concentrated on debugging and optimizing the model.
- Benchmarking and Testing: Once the KNN model is finalized, extensive testing and performance analysis will be conducted on all parts of the project. This includes verifying the correctness of the implementations and assessing their performance improvements.
- Preparation for Poster Session: The set of goals for the poster session will involve showcasing the completed project, highlighting the key achievements in parallelizing KDTree construction and search functionalities, and demonstrating the performance benefits.

Potential Adjustments:

While initial goals are on track, the focus on implementing a lock-free KDTree may require adjustments in the timeline for the remaining tasks, especially the Parallel KNN Model search.

The "nice to haves" in the project may need to be re-evaluated based on the progress in the coming weeks, with a focus on ensuring that core deliverables are met without compromise.

Overall, the project is well-aligned with its objectives, with significant progress made in key areas. The next few weeks will be crucial for fine-tuning and validating the implementations, ensuring both their correctness and efficiency.

Updated Schedule

| Task | Dates | Assignee | Status |
|--------------------------------------|---------------------|---------------------|-------------|
| Project Proposal | Nov 7 - Nov 15 | Doreen and Neelansh | Completed |
| Sequential KD Tree algorithm | Nov 19 - 23 | Doreen and Neelansh | Completed |
| Parallelize KD tree Construction | Nov 23 - Nov 26 | Neelansh | In Progress |
| Sequential KNN search | Nov 27 - Dec 3 | Doreen | Completed |
| Lock Free KD Tree Insertions | Nov 27 - Dec 3 | Neelansh | Completed |
| Finish Milestone Report | Dec 3 - Dec 4 | Doreen and Neelansh | Completed |
| Parallel KNN Search | Dec 5 - Dec 9 | Doreen | Not Started |
| Performance Benchmarking for KD Tree | Dec 6 - Dec 10 | Doreen and Neelansh | Not Started |
| Performance Benchmarking for KNN | Dec 6 - Dec 10 | Doreen and Neelansh | Not Started |
| Poster Session Prep | Dec 14 - Dec 15 | Doreen and Neelansh | Not Started |
| Finish Final Report | December 7 - Dec 14 | Doreen and Neelansh | Not Started |

Poster Session Presentation:

1. Extensive Graphs and Analysis Reports:
 - a. Our presentation will include detailed graphs and analytical reports.
 - b. These will illustrate the performance benefits of parallel implementation versus the sequential KDTree.
 - c. We will also highlight the advantages of using a lock-free data structure as opposed to a mutex-based locked structure.
2. Comparative Measurements:
 - a. The graphs will compare various aspects such as:
 - i. Different implementations over varying KDTree dimensions.
 - ii. Performance with varying data point counts.
 - iii. Efficiency using different numbers of threads.

Concerns and Challenges:

Parallel KNN Search: The primary challenge lies in finalizing the parallel KNN Search. This involves dedicated coding and debugging sessions to ensure correctness and efficiency.

Parallel KD Tree: We would like to further parallelize the building of our KD Tree which will require more thought about the specific OpenMP directives which can be used in our implementation.

Benchmarking and Performance Validation: A significant concern is the final benchmarking phase. It's crucial to demonstrate that the parallel implementations are decently faster than the sequential ones to validate the success of your project.