# Fall 2023: ME759 Final Project Proposal

**Quick intro:**


- Name: Varan Shukla
- Email: varan.shukla@wisc.edu
- Home department: Computer Sciences
- Status: MS Student
- Name of your teammate (if applicable): Harsh Sahu(hsahu@wisc.edu)


**Project Title**: Implementing Convolutional Neural Networks in Parallel

**Link to git repo for project**: https://git.doit.wisc.edu/VARAN.SHUKLA/hpc-finalproject/

**Problem statement**:

Parallelised implementation of a single forward pass for a custom Convolutional Neural Network using CUDA. The forward pass implementation will involve parallelising the following operations: Convolution Layer, Pooling Layer, Padding layer, Dense Layer and Activation Layer.

**Motivation/Rationale**:

Being actively involved in Deep Learning research, we believe that optimizing deep learning models' runtime is one of the fundamental tasks. Parallelizing operations in a CNN's forward pass can significantly reduce the inference runtime. This can be achieved by distributing the workload across multiple processors or dedicated devices like GPUs, making it feasible to train and use bulky CNN models with huge datasets. Existing frameworks and libraries already provide blazing fast solutions where most of the computations are done on the GPU with the help of CUDA library. With the help of this project, we believe that we can solidify the conceptual understanding of writing efficient kernels from the algorithmic point of view. At the same time, this project gives us an opportunity to learn the underlying concepts of deep learning and implementation of an end-to-end framework consisting of convolution, pooling, dense neural network layers etc.

**Explain how you contemplate going about it**:

We'll be using CUDA API to parallelise the compute tasks on GPU. We'll be programming in C++ and will probably be using the cuBLAS library, although we plan on writing our own kernel functions from scratch which will help us to parallelise our operations and have a more fine-grained control over its parallelization. We intend to create small functional units for each of the steps i.e. convolution, pooling, activation etc. and create a streamline job to process the input on the trained model and produce the output. Given a trained model we'll benchmark the time taken to process the final output using our parallelised implementation and will compare it against non-parallel solutions and other state-of-the-art solutions. Although we are not certain on the model and parameters we'll be using, we might pick a well-known and trained model such as one for the MNIST image classification.

**ME759 aspects the proposed work draws on**:

- Usage of CUDA APIs to parallelize compute heavy tasks on GPU
- Usage of libraries like cuBLAS and thrust
- fine-grained control on thread, block and grid level operations in GPU
- Timing the results and comparing against non-parallel solutions

**Deliverables**: We expect to deliver a working forward pass parallelised implementation code with a technical report by 12/13/2023. The repo would contain the steps to reproduce our results, while the report will analyze our learnings and compare the results we'll obtain.

**How you will demonstrate what you accomplished**: We will compare runtime of our parallel implementation against a standard library using the same CNN architecture using CPU as well as GPU. We'll try to emphasize the benefits of our parallelization by comparing against non-parallel solutions.