Convolutional Neural Networks are often trained on datasets of images. Sometimes the data is so large, a single sample cannot fit in memory. This requires splitting the sample over multiple nodes. The performance is constrained I/O, taking more time to move data than the computations itself.

This work aims to expand the implementation of CNNs in LBANN to reduce the overhead of I/O by loading the data across different nodes using MPI I/O.

**Background**

**Neural Network Parallelism**
- Model Parallelism: Splitting the model across resources
- Sample Parallelism: Splitting the data across resources
- Spatial Parallelism: Splitting the sample across resources
- These different methods can be combined together

**Application and Implementation**

This summer we are working with a large 3D cosmology dataset
- Each sample is 1 GB, 4x512x512x512
- Currently I/O takes more time than the computation

**Figure 1** An example of Spatial Parallelism, [B] Sample Parallelism, without any spatial parallelism.

**Figure 2** Example cosmology data, a simulation of dark matter in the universe, evolved over 3 billion years to a redshift of 0. This will be used as input to the CNN.

**Figure 3** Each process will read in a portion of the sample using MPI I/O.

**Figure 4** Example workflow each IBM Power 9 would read in the appropriate section of data

**Figure 5** https://hpc.llnl.gov/hardware/platforms/lassen

**Figure 6** Preliminary results, 2 processes on each node reading in half of the data, as seen in Fig 1A. The data is of size 512x512x512x4, about 1GB each file.

**Results**

<table>
<thead>
<tr>
<th>Process</th>
<th>Seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Node</td>
<td>2.5</td>
</tr>
<tr>
<td>5 Nodes</td>
<td>2.0</td>
</tr>
<tr>
<td>10 Nodes</td>
<td>1.5</td>
</tr>
<tr>
<td>20 Nodes</td>
<td>1.0</td>
</tr>
<tr>
<td>25 Nodes</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Results are collected by running the program outside of LBANN multiple times and averaging the results. Preliminary results show a promising speed up.

Next steps include:
- Loading more samples
- Integrating into LBANN

**Conclusion**

As datasets and sample sizes continue to grow it is important the tools can scale to match the needs.

- As the results show we can achieve around a 3x speed up by using 5 nodes and MPI I/O, instead of a single node.
- This can be used in the future on datasets such as MRIs.
- This allows for more complex datasets to utilize machine learning.

**Citations**

[1] https://github.com/LLNL/lbann