IARIA Congress 2024

A Comparative Analysis of CPU and GPU-Based Cloud Platforms for CNN Binary Classification

PRESENTER: DR. FAN WU

AUTHORS: TAIEBA TASNIM, DR. MOHAMMAD RAHMAN, DR. FAN WU

COMPUTER SCIENCE DEPARTMENT,

TUSKEGEE UNIVERSITY

DATE: 2ND JULY,2024

Dr. Fan Wu

fwu@tuskegee.edu

Working Experience

- Head, Computer Science Department, Tuskegee University, Tuskegee, AL
- Professor, Computer Science Department, Tuskegee University, Tuskegee, AL
- Director, Center of Information Assurance Education (CIAE), Tuskegee, AL
- Director, Tuskegee University Office of Undergraduate Research (TUOUR), Tuskegee, AL

Research Areas

Mobile Security, Information Assurance, Data Science, Machine Learning, Mobile Graphics, Mobile Computing, Computer Graphics, Bioinformatics, Biostatistics, High Performance Computing with GPGPU Technology, and Robotics

Outlines

- Introduction
- Literature Review
- Methodology
- Results
- Conclusion and Future work
- Acknowledgement
- References

Introduction

Convolutional Neural Networks (CNNs):

- Specialized for image classification, detection, and segmentation.
- Extract features using pooling layers for accurate predictions.
- Inspired by the human visual cortex.
- Trained on extensive datasets with backpropagation and human-configured parameters.

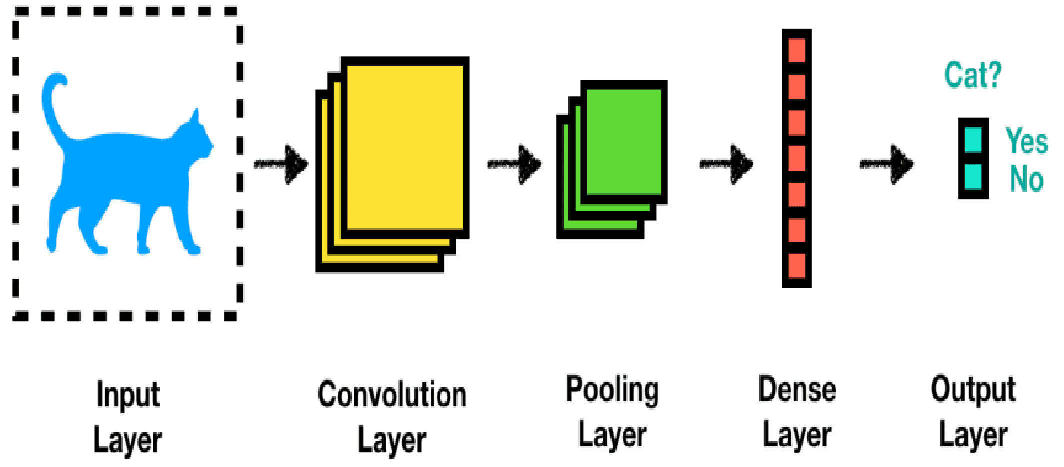


Image adapted from Ng, Andrew. "AI for Everyone." Coursera.

Goals and Objectives of our paper

The goal of our paper is to:

- Evaluate CNN binary classification performance on CPU and GPU cloud platforms.
- Provide comprehensive benchmarking analysis of computational efficiency.

Objectives of our study are:

- Comparative analysis of CPU vs. GPU performance for CNNs.
- Methodological insights into implementing CNNs on different architectures.
- Empirical data from extensive experiments on benchmarking datasets.
- Practical guidelines for deploying CNN models on CPU and GPU platforms.

Literature Review

Importance of CNNs:

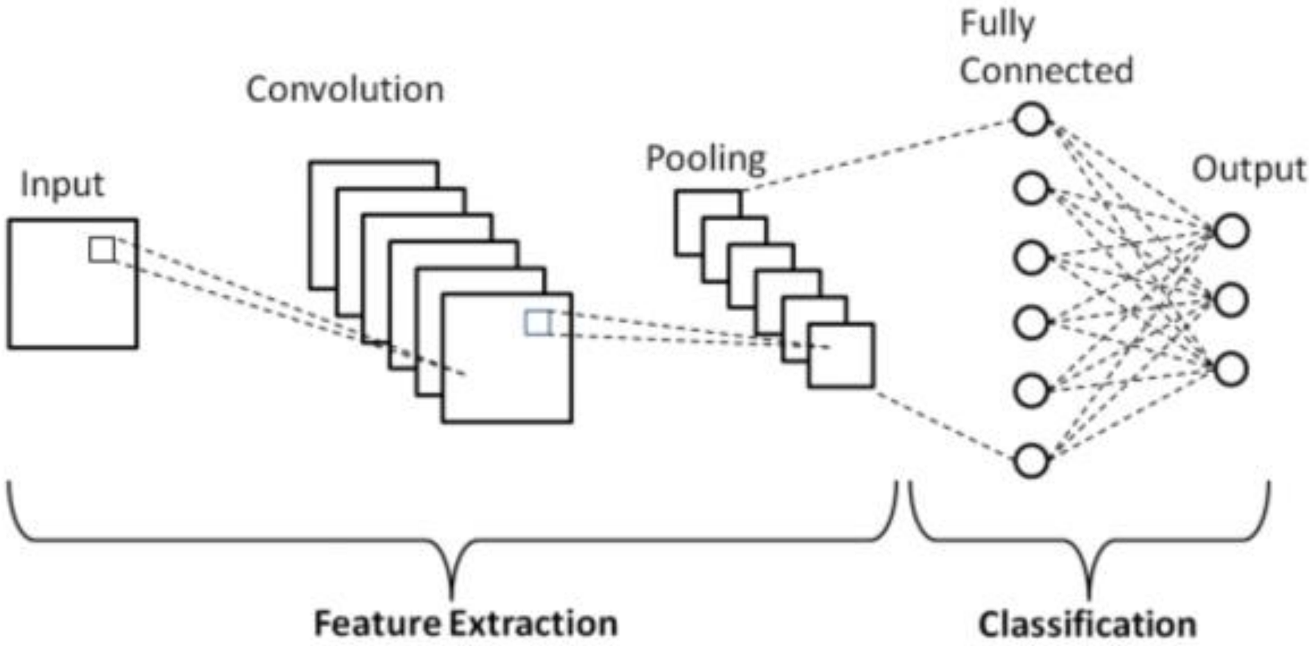
- GPUs outperform CPUs by 2 to 24 times in CNN tasks due to parallel processing capabilities (Strigl et al., 2010; Cengil et al., 2017).
- CPUs have sequential processing limitations (Strigl et al., 2010).

Performance Factors:

- Power efficiency and cost are critical in hardware selection (Süzen, 2020).
- CPUs in embedded systems achieve 65% of a PC's GPU performance with only 2.6% of the power (Oh et al., 2017).

Benchmarking Studies:

- Machine learning models predict CNN execution time, power, and memory usage to aid hardware selection (Bouzidi et al., 2022).



Methodology

1. CNN Architecture:

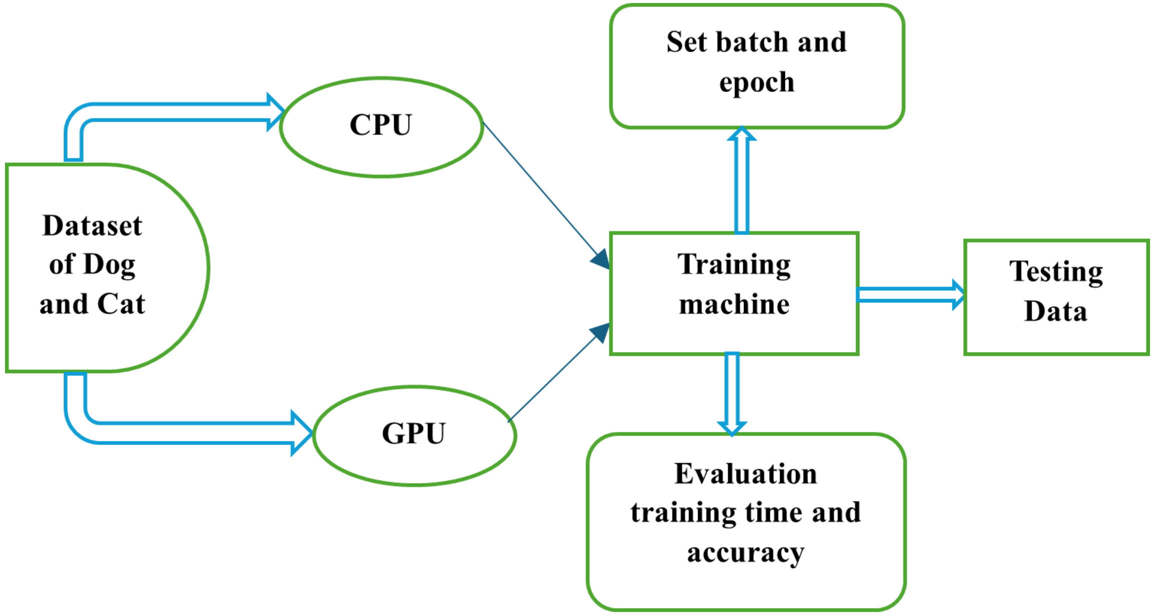
- The architecture includes input layers, convolutional and pooling layers, and fully connected layers.
- The goal is to recognize and interpret intricate patterns in the dataset, consisting of high-quality images of dogs and cats.

Image adapted from Phung, V. H. and E. J. Rhee (2018). Journal of information and communication convergence engineering **16**(3): 173-178.

Methodology

2. Data Acquisition

- Utilized comprehensive datasets from Kaggle and Google.
- Included a diverse collection of high-quality images of various dog and cat breeds.
- These datasets are essential for training and evaluating the CNN models effectively.



Methodology

3. Experimental setup:

- Trained CNN model using Google Colab with Keras support on Google Cloud's CPUs and GPUs.
- Achieved high training speeds and used network pruning without losing accuracy.
- Made minor code adjustments to improve GPU performance and ensure consistency.
- Imported data from Google Drive, transitioned from CPU to GPU training, and optimized tensor operations and memory management for faster, accurate results.

Evaluation Metrics

1. **True Positive Rate (TPR):**

$$TPR = \frac{TP}{TP + FN}$$

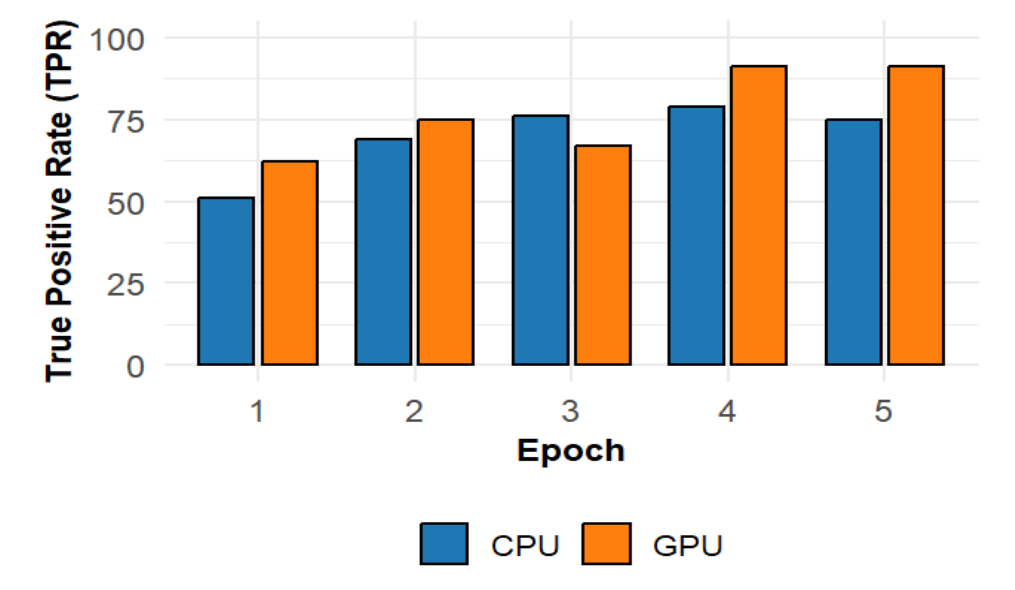
- Measures the proportion of correctly identified positive instances.
- Essential for evaluating the model's accuracy in scenarios with class imbalances.

2. Training Time:

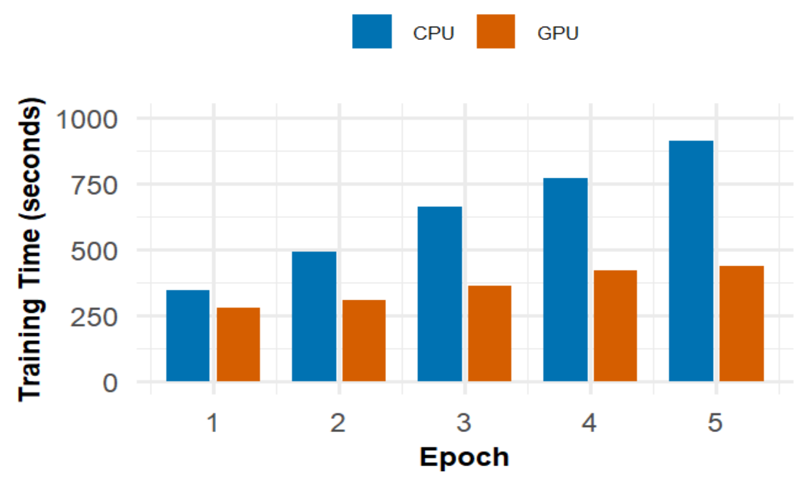
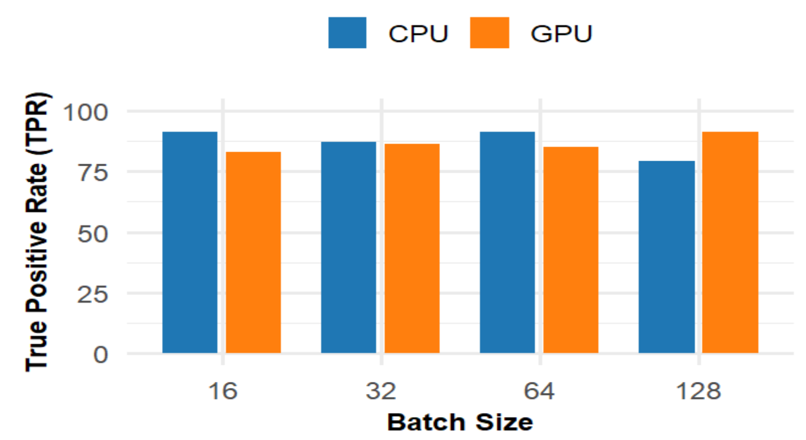
- Monitored to assess model efficiency across different hardware setups.
- Highlights trade-offs between accuracy and speed.

Results

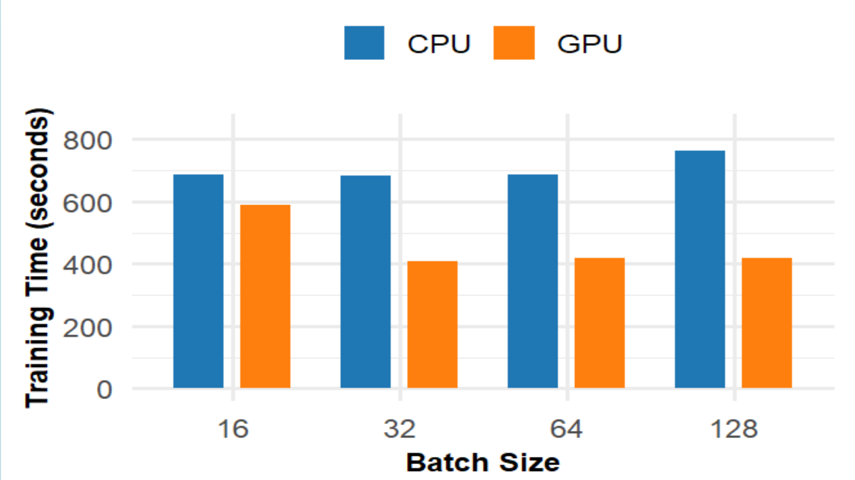
- Trained model on an 8000-image dataset of dogs and cats.
- Batch sizes (16, 32, 64, 128) and Epochs (1 to 5).
- Used 1000-image dataset for comparative analysis.

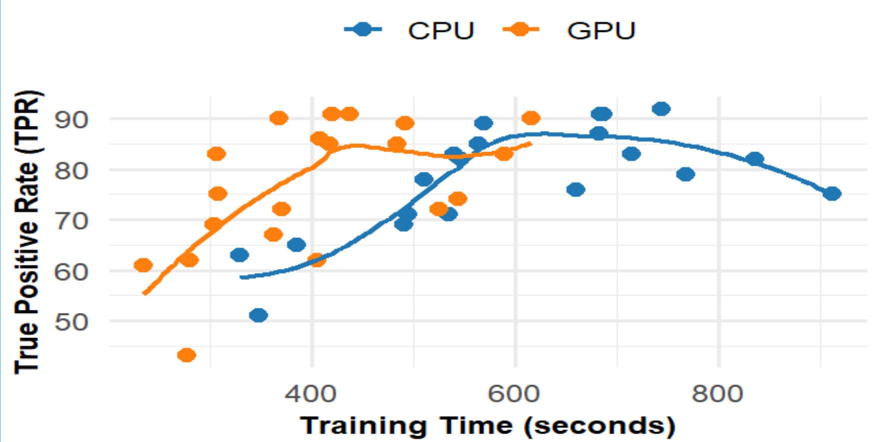
<table>
<tr><th>Figure</th><th>Observations</th></tr>
<tr><td>
TPR by Epoch</td><td>CPU has a better TPR than GPU in the early epoch.
GPU shows consistent improvement, especially in later epochs.
GPU achieves peak TPR in the fourth epoch.
Overall, GPUs demonstrate superior performance in extensive training iterations compared to CPUs.</td></tr>
</table>

Results

Figure	Observations
 Training Time by Epoch	• GPUs have consistently lower training times than CPUs initially. • CPU times rise significantly; GPU times remain stable and faster. • GPUs reduce training time for each epoch, proving superior efficiency.
 TPR by Batch Size	• Both CPU and GPU perform well across batch sizes; CPUs slightly outperform GPUs at smaller sizes. • TPR remains high and consistent for both, showing their effectiveness in binary classification tasks.

Results

<table>
<tr><th>Figure</th><th>Observations</th></tr>
<tr><td>

Training Time by Epoch</td><td>GPUs have consistently shorter training times across all batch sizes compared to CPUs.
GPU training times remain stable as batch sizes increase, while CPU times rise.
GPUs offer a clear advantage in training time efficiency over CPUs.</td></tr>
<tr><td>

TPR by Batch Size</td><td>GPUs reach high TPR with shorter training times compared to CPUs.
GPUs achieve peak TPR faster than CPUs.
CPUs show a slight TPR decline beyond 600 seconds, while GPUs maintain stable high performance.</td></tr>
</table>

Conclusion & Future work

Conclusion

- GPUs consistently outperformed CPUs in training efficiency and execution speed.
- GPUs achieved higher or comparable True Positive Rates (TPR) with marked performance consistency.
- GPUs showed superior efficiency, especially with increased batch sizes, compared to CPUs.
- GPUs offer substantial advantages in speed and accuracy for extensive CNN tasks.

Future Work

- Include more hardware models, such as NVIDIA's Tesla and RTX series.
- Better understand CPU and GPU performance differences to select optimal hardware for CNN tasks.
- Conduct cost and performance analysis to improve hardware selection and

Acknowledgement

The work is partially supported by the National Science Foundation (NSF) under NSF Awards #2019561, #2234911, #2209637, and #2100134. The opinions, findings, and recommendations in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References:

[1] S. Verma et al., "Demystifying the MLPerf Training Benchmark Suite," in 2020 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), 2020, pp. 24-33.
[2] D. Strigl, K. Kofler, and S. Podlipnig, "Performance and Scalability of GPU-Based Convolutional Neural Networks," in 2010 18th Euromicro Conference on Parallel, Distributed and Network-based Processing, 2010, pp. 317-324.
[3] E. Buber and B. Diri, "Performance Analysis and CPU vs GPU Comparison for Deep Learning," in 2018 6th International Conference on Control Engineering & Information Technology (CEIT), 2018, pp. 1-6.
[4] E. Cengil, A. Çinar, and Z. Güler, "A GPU-based convolutional neural network approach for image classification," in 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), 2017, pp. 1-6.
[5] M. U. Yaseen, A. Anjum, O. Rana, and R. Hill, "Cloud-based scalable object detection and classification in video streams," Future Generation Computer Systems, vol. 80, pp. 286-298, 2018/03/01/ 2018.
[6] A. A. Süzen, B. Duman, and B. Şen, "Benchmark Analysis of Jetson TX2, Jetson Nano and Raspberry PI using Deep-CNN," in 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), 2020, pp. 1-5.
[7] S. Oh, M. Kim, D. Kim, M. Jeong, and M. Lee, "Investigation on performance and energy efficiency of CNN-based object detection on embedded device," in 2017 4th International Conference on Computer Applications and Information Processing Technology (CAIPT), 2017, pp. 1-4.
[8] M. Blott et al., "Evaluation of Optimized CNNs on FPGA and non-FPGA based Accelerators using a Novel Benchmarking Approach," Proceedings of the 2020 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, pp. 317-317, 2020.
[9] A. Saha, M. Rahman, and F. Wu, "Evaluating LSTM Time Series Prediction Performance on Benchmark CPUs and GPUs in Cloud Environments," 2024, pp. 321-322.
[10] H. Bouzidi, H. Ouarnoughi, S. Niar, and A. A. E. Cadi, "Performance Modeling of Computer Vision-based CNN on Edge GPUs," ACM Transactions on Embedded Computing Systems, vol. 21, no. 5, pp. 1-33, 2022.
[11] Google Colaboratory (2024). https://colab.research.google.com. Last Accessed 10 Mar 2024.
[12] Keras (2024). https://keras.io. Last Accessed 10 Mar 2024.
[13] V. Sharma, G. K. Gupta, and M. Gupta, "Performance Benchmarking of GPU and TPU on Google Colaboratory for Convolutional Neural Network," in Applications of Artificial Intelligence in Engineering, Singapore, 2021, pp. 639-646: Springer Singapore.
[14] V. H. Phung and E. J. Rhee, "A deep learning approach for classification of cloud image patches on small datasets," Journal of information and communication convergence engineering, vol. 16, no. 3, pp. 173-178, 2018.

Thank You