УДК 004.272.26

EVALUATION OF MEMORY ACCESS METHODS FOR TRAINING CONVOLUTIONAL NEURAL NETWORKS ON GPU CUDA

Альдарф Алаа (Национальный исследовательский университет ИТМО)
Шакер Алаа (Национальный исследовательский университет ИТМО)
Научный руководитель – проф. ФПИиКТ, Бессмертный И.А.
(Национальный исследовательский университет ИТМО)

This work discusses the possible time gain that can be achieved by comparing and analyzing three different implementations of data transfer in GPU CUDA to speed up the convolutional neural networks learning time.

Введение. Convolutional Neural networks (CNN) are widely used in many domains such as Handwriting recognition, Object identification, Data classification, Pattern recognition and Natural Language processing. CNN training requires a large amount of computation and data, training with large datasets can take a very long time. The huge number of floating-point operations and the relatively low data transfer rate at each training stage make this task well suited for general-purpose GPU computing using parallel programming models such as NVIDIA CUDA.

Data transfer and memory access are the main parts to speed up the neural network training on the GPU. There are many methods in the CUDA Framework to optimize memory performance, and choosing the right method that matches the application area will result in better performance. This work evaluated the memory access methods for training convolutional neural networks on GPU CUDA.

Основная часть. This work studies the effect of memory access techniques on the training time of the convolutional neural networks by performing the following steps:

1) Implementing the standard synchronous CNN training using the CUDA Framework.

2) Implementing the asynchronous copy-to-memory technique. Asynchronous copy enables overlap of data transfers with computations. On all CUDA-enabled devices, it is possible to overlap CPU computation, data transfers and GPU computations.

3) Implementing the zero-copy technique. Zero copy is a feature in the CUDA Toolkit. It enables GPU threads to directly access host memory.

4) Evaluating the results by comparing the training time and the accuracy of the neural network.

Выводы. The results of this work show that the accuracy values of the different implementations are close to each other and can reach up to 90%. For the training time, the asynchronous copy-to-memory implementation can get an 18% time acceleration over the standard implementation, while the zero-copy implementation slows down the execution and increases the training time by 280%.

This improvement in training speed using the asynchronous copy-to-memory is due to the fact that this implementation allows the program to overlap data transfer and computation. Although, the reduction in training speed using the zero-copy because the GPU threads must read data directly from the host memory. Thus, implementing the neural network training on a GPU will be more efficient using asynchronous copy-to-memory than zero-copy or the standard synchronous implementation, whereas zero-copy can be useful when the GPU has insufficient memory.

Альдарф Алаа (автор)	Подпись
Шакер Алаа (автор)	Подпись
Бессмертный И.А. (научный руководитель)	Подпись