CUDA Optimized dynamic programming search for automatic speech recognition on a GPU platform

Babedi B. Letswamotse[1], Naison Gasela[1] and Zenzo P. Ncube[2]
Department of Computer Science
North-West University, Mafikeng Campus[1], Private Bag X2046, Mmabatho 2745
Tel: +27 18 3892811, Fax: +27 18 3929775
and Department of Information and Communication Technology
Sol Plaatjie University[2], Private Bag X5008, Kimberley 8300
email: {18040969, Naison.Gasela}@nwu.ac.za[1]; Zenzo.Ncube@spu.ac.za[2]

Abstract- In a typical recognition process, there are substantial parallelization challenges in concurrently assessing thousands of alternative interpretations of a speech utterance to find the most probable interpretation. During this process, input signals are converted into feature vectors, thus decoding these feature vectors to produce relevant output is a computationally expensive task. Many time critical applications are unable to use Automatic Speech Recognition (ASR) due to the heavy latency in processing the speech with a large vocabulary size. This work proposes to optimize the performance of dynamic programming search. Optimizing dynamic programming search requires a certain level of parallelism since search is a parallel process. We find that a better way to optimize speech recognition search is by the use of parallel architectures such as graphic processing units (GPUs). GPUs provide large computational power at a very low expense which positions them as global accelerators. These savings encourage using GPUs as hardware accelerators to support computationally intensive applications.

Index Terms— GPU, Decoding, Dynamic Programming, Automatic Speech Recognition, CUDA

I. INTRODUCTION

Speech is the most effective form of communication for human to human interactions, so people expect the same when it comes to human-machine (computer) interactions. They expect the speech recognition systems in which the computer speaks and recognizes any human language. For these expectations to be met speech recognition has to be put into practice. Speech Recognition is the process of recognizing spoken input and converting it into written text through a speech recognition system.

Speech recognition systems are also used in a wide range of applications such as air traffic control, embedded telecommunication systems, robotics and, computer and video games. Many systems in the real world make use of speech recognition, these kinds of systems assist in so many ways. They assist with online shopping and voice activated passwords, security systems etc. Optimizing the performance of automatic speech recognition systems will help improve the services provided by these systems both commercially, socially and educationally.

Most of the modern speech recognition systems are usually based on statistic models such as Hidden Markov Models (HMMs). According to [1], HMMs are popular due to their simplicity and computational feasibility, and their parameters that can be estimated automatically from a large amount of data.

Speech recognition is divided into three phases: feature extraction, classification and decoding. The classification stage is a collection of segmented words and sub-words into different classes based on some properties [2]. Classification is composed of acoustic models which are files that are generated by taking audio recordings of speech and their transcriptions and then compile them into statistical representations of the words sounds. Each of these statistical representations is assigned a label called a phoneme [3], pronunciation dictionary which is a machine-readable dictionary that contains a collection of words and their transcriptions and the language model which is a probability distribution P(s) over words S that attempts to reveal how frequently a string S occurs as a sentence. Language models are often used for dictation applications. In any speech recognition systems there are two vital metrics to contemplate: the elapsed time between the acquisition of the speech signal and the recognized word, and the accuracy.

In a typical recognition process, there are substantial parallelization challenges in concurrently assessing thousands of alternative interpretations of a speech utterance to find the most probable interpretation. Most time critical applications are unable to use Automatic Speech Recognition (ASR) due to the heavy latency in processing the speech with a large vocabulary size.

In this research the authors focus on the decoding process of speech recognition. The decoding process (which is often referred to as search) in a speech recognizer's operation is to find a sequence of words whose corresponding acoustic and language models best match the input feature vector sequence [1]. Search is a computationally expensive task since it handles irregular graph structures with data parallel operations. There are algorithms that were developed specifically for this task but many search algorithms were developed prior to the existence of parallelism. Dynamic programming has a complexity of O (n^2), which can cause unreasonable demands on both the processing time and system memory. Parallelizing the Dynamic Programming search will help improve the efficiency of the recognition process hence reducing the latency in processing speech for large vocabulary systems.

II. RELATED WORK/BACKGROUND

Rehman et al. [4] implemented a dynamic programming algorithm (Viterbi) on NVidia graphics processing unit using CUDA and concluded that it has been accelerated from 3 to 6 times as compared to the serial execution on

central processing unit.

Hachkar et al. [5] used two algorithms to implement a system of Automatic Recognition of isolated Arabic Digits: Dynamic Time Warping (DTW) and Discrete Hidden Markov Model (DHMM). DTW-based system recognition leads to recognition accuracy of 77%. The better recognition accuracy of about 92% was obtained with DHMM-based system. They found that the recognition performances for the two ASR systems are worse in noisy environment, but the pattern recognition using HMM is better than the pattern using DTW.

Wei and Weisheng [6] attempted on improving the recognition efficiency without compromising the recognition accuracy. They analysed the traditional Viterbi-Beam search algorithm and proposed an improved adaptive Viterbi- Beam search algorithm by analysing the voice activity model of different stages. The method combining Viterbi algorithm with Beam pruning technique is useful to compress the search space, which reduces the computational complexity. The experimental results show that the search space is compressed effectively without affecting the recognition accuracy and an improvement on search efficiency of 35.77% is observed.

III. RESEARCH METHODOLOGY

This work follows three processes which form the overall research methodology. The first being the dynamic programming search method survey, which will be done to look at what recent speech processing improvements have been achieved using this approach particularly on CUDA GPUs. Secondly is the experimental setup, which will be focusing on the actual experimental tools and how they will be setup. Thirdly as proof of concept, performance comparisons between the then CUDA GPU optimized dynamic programming search algorithm and the original algorithm will be done.

HTK assimilated CMU sphinx 4 recognizer will be used for recognition on both the Linux based workstation with Intel core i7-3770 CPU and the GPU based system running on GTX 8800. The Hidden Markov Models will be connected together in a sequence to enable continuous speech. A speech recognition Viterbi search algorithm will be implemented on both systems. The Viterbi search algorithm will be optimised by loop unrolling to improve the optimality of the search process and thus improving the efficiency of the recognition process.

CUDA will be used to implement the optimised version of the speech recognition Viterbi search algorithm on the GPU based system using the already existing language corpus. We will make use of the shared memory only for communication intensive processes due to the memory limitations associated with shared memory. For this implementation we will synchronize the threads to ensure that the parallel threads cooperate in order to yield correct results and avoid deadlocks. We will use a barrier synchronization primitive called _syncthreads ( ) which is provided by CUDA. Performance results of the implementations will also be thoroughly analyzed and evaluated. Tools to be used include the following:

- HTK version 3.4.1
- CMU (Carnegie Mellon University) Sphinx 4
- GTX 8800 GPU
- NVidia CUDA 5.5 Toolkit
- Linux based workstation with Intel core i7-3370 CPU only
- CMU US BDL Arctic 0, 95 Speech corpus

IV. CONCLUSIONS

The aim of this research is to optimize the performance of the Speech Recognition Dynamic Programming search, therefore we are going to implement a dynamic programming based search algorithm (Viterbi) and optimize it using CUDA.

REFERENCES

[1] N. Indurkhya and F. J. Damerau, "An Overview of Modern Speech Recognition, "in *Handbook of natural language processing*. London: CRC Press, 2009, Ch. 15, pp. 339-366.

[2] M. Rahman,F. Khan, and A. Bhuiyan, "Continuous Bangla speech segmentation, classification and feature extraction., "*International Journal of Computer Science Issues, (IJCSI)*, vol. 9, no. 2, pp. 67-75, March (2012).

[3] Acoustic Modelling. Microsoft Research. [Online], http://research.microsoft.com/en-us/projects/acoustic-modeling/ (Accessed: 8 March 2014).

[4] M. K. Rehman, M. U. Sarwar, M. R. Talib, M. S. Mansoor and M.B. Sarwar, "Parallel Implementation of Dynamic Programming Algorithm Using Graphics Processing Unit," International Journal of Computer Science and Management Research, vol. 2, no. 4, pp. 2097-2107, April 2013.

[5] Z.Hachkar, A. Farchi , B.Mounir and J. EL Abbadi "A Comparison of DHMM and DTW for Isolated Digits Recognition System of Arabic Language," International Journal on Computer Science and Engineering ,(IJCSE), vol. 3, no. 3, pp. 1002- 1008, March 2011.

[6] H. Weisheng and L. Wei, "Improved Viterbi Algorithm in Continuous Speech Recognition ," in International Conference on Computer Application and System Modeling (ICCASM 2010), Taiyuan, 2010, pp. v7-207- v7-209.

Babedi Betty Letswamotse received her undergraduate degree (Computer Science and Mathematics) in 2011 and her Honours degree (Computer Science) in 2012 from the North West University (Mafikeng Campus) and is presently studying towards her Master of Science degree in Computer Science at the same institution. Her research interests include GPU general purpose computing and automatic speech recognition.