DOI: https://doi.org/10.30837/csitic52021232891

Using OpenMP Directives to Accelerate OCR with Tesseract OCR

Olesia Barkovska [1]

Ihor Ryzhov[2]

[1,2]Kharkiv National University of Radio Electronics, 14 Nauky Ave,
Kharkiv UA-61166, Ukraine

Abstract. *This paper is devoted the methods of speed-up optical character recognition which is used for transformation of the scanned image to the edited text format. The example of application of these methods are the systems of the automated search of fragment of text in the catalogues of electronic libraries, where as an entrance format both the entered text and vocal query or scanned fragment of the text document can be used. The paper shows that the quality of the original image, as well as the applied image preprocessing algorithms, has the greatest influence on the quality of text recognition. Today the task of text recognition is implemented in many libraries. An example is the Tesseract OCR, considered in the work. It is shown that the joint use of the standard parallel programming library OpenMP, which is built into all modern C and C ++ compilers, reduces the time of processing up to 33% compared to the sequential implementation.*

Keywords: *OCR, Tesseract, multithreading, optical character recognition, parallelism.*

I. Introduction and Problem Statement

Optical Character Recognition, or OCR, is one of the major topics in computer vision technology. It is widely used in various applications, such as a digital libraries, automatic banking systems, and mailing services. Tesseract OCR Engine, which we evaluate in this paper, is one of renowned OCR programs. The library has been expanded in various languages besides English by adding additional training models [1]. OCR process including Tesseract is known to be very compute intensive because of the computation involving image and mathematical processing to obtain higher recognition accuracy. While there has been a significant improvement in the recognition accuracy of Tesseract OCR, parallelization has not been extensively studied. Therefore, implementation and adaptation of the systems of optical recognition of text for a decision on the systems with CMY, is actual task.

The purpose of the work is research of influence of the use of directives of OPENMP on the acceleration of optical character recognition by the library of Tesseract OCR.

II. Problem Solution and Results

Tesseract OCR performs character recognition on an image in several stages (Figure 1). The first one is searching for words and strings [3]. It finds text strings by analyzing the page layout and the estimated text size. If the text string is bent or skewed, Tesseract OCR can recognize the character by basic fit [2].

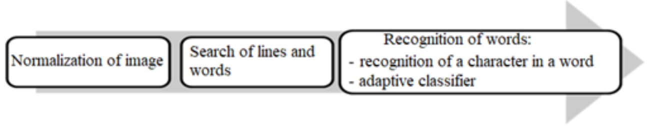
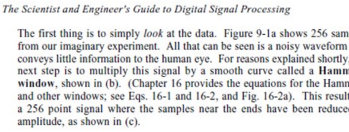


Figure 1. Stages of text recognition

It consists of two steps: one is the recognition of characters in a word, and the other is a process of improving accuracy, called "adaptive classifier" [2]. The adaptive classification stage is an iterative procedure where the number of iterations depends on the number of recognized words. The OpenMP API allows users to use multiple threads to run multiple iterative processes concurrently and use shared memory that these threads can access. A scanned fragment of text, consisting of 867 words, was used as a test image (Figure 2).



The Scientist and Engineer's Guide to Digital Signal Processing

The first thing is to simply *look* at the data. Figure 9-1a shows 256 samples from our imaginary experiment. All that can be seen is a noisy waveform that conveys little information to the human eye. For reasons explained shortly, the next step is to multiply this signal by a smooth curve called a **Hamming window**, shown in (b). (Chapter 16 provides the equations for the Hamming and other windows; see Eqs. 16-1 and 16-2, and Fig. 16-2a). This results in a 256 point signal where the samples near the ends have been reduced in amplitude, as shown in (c).

Figure 2. Fragment of the text image

The results obtained with the use of *pragma omp parallel* for directives, as well as without them, are shown in Table 1. To determine the resulting acceleration, the ratio of the time of the serial to parallel implementation was used.

Table 1. Results of parallel processing

Number of threads	Elapsed time execution, milliseconds	Speedup	Accuracy, %
1	~6000	-	94
2	~5400	10%	94
4	~4000	33%	94

Analysis of the obtained results shows that the accuracy of character recognition when using the Tesseract OCR library was 94%. Application of parallelization in the most computationally loaded loops using OpenMP led to a reduction in the total time by up to 33% for 4 OpenMP threads.

III. Conclusions

It was investigated the effect of using OpenMP directives on accelerating optical character recognition using the Tesseract OCR library. The results showed that the overall performance was improved by 33% compared to the sequential processing version. There are also ways to speed up target kernels using other parallelization mechanisms such as CUDA and OpenCL.

References

[1] L. Vincent, “Announcing Tesseract OCR.” [Online]. Available: http://googlecode.blogspot.com/2006/08/announcing-tesseract-ocr.html

[2] R. Smith, “An Overview of the Tesseract OCR Engine,” in Proceedings of the Ninth International Conference on Document Analysis and Recognition - Volume 02, ser. ICDAR ’07, 2007, pp. 629–633.

[3] Olesia Barkovska, Oleg Mikhal , Daria Pyvovarova , Oleksii Liashenko , Vladyslav Diachenko and Maxim Volk, Local Concurrency in Text Block Search Tasks, International Journal of Emerging Trends in Engineering Research. - Volume 8. No. 3, March 2020. – P.690-694.