ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 6, Issue 2, February 2019

Practical Use of OCR Library - Tesseract

Mirzayev Mirodil, Malikova Kamila

Professor, Tashkent University of information technologies of Fergana Branch
3rd year student, Tashkent University of information technologies of Fergana Branch

ABSTRACT: This article gives general information about Optical character recognition (OCR) technique. One can find list of different libraries which solve the problem of OCR prepared using programming languages. As an example the library called Tesseract is used to provide more information about OCR. This article also describes the use of the Tesseract OCR library and shows an example of its practical use in the application for the Windows operating system family.

KEYWORDS: OCR; Optical Character Recognition; Tesseract; C#; command line; Visual Studio.

I.INTRODUCTION

Modern technologies help us in many areas of life: from the most complex operations to simple reminders in mobile phones. Today, quite popular programs that make it possible to extract data from files of different formats. For example, the transformation of an audio file into text or text into an audio file, extracting text from an image, and the like.

Based on the last example, obtaining text from an image is called OCR technology (Optical Character Recognition) - a popular trend in programming. There are a huge number of programs that implement OCR. But at the same time, practically each of these programs uses one set of libraries:

- Tesseract OCR
- Apache Tika
- OpenCV
- IronOCR.

One of the most popular libraries is the “Tesseract” library. A feature of the Tesseract library is that it is considered the Open Source Library and is written in C++. Despite this, there are its counterparts in other programming languages and for other platforms. The equivalent is “Tessnet” in C # (C-Sharp).

GitHub, a popular developer community, has a huge number of projects using the Tesseract library, both open source and closed source. There is a small console program that rather accurately performs an OCR operation [1].

First of all, in order for the program to work, you need to go through the command line to the folder with this project:

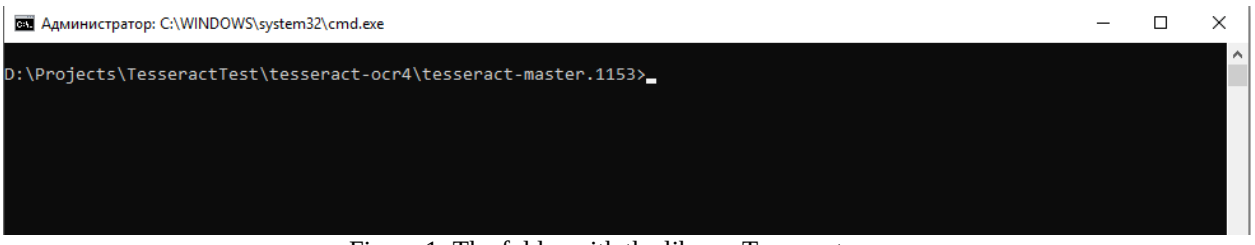

Figure 1: The folder with the library Tesseract

Next you need to start the program by typing in the command line the name of the program - tesseract.exe:

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 6, Issue 2, February 2019

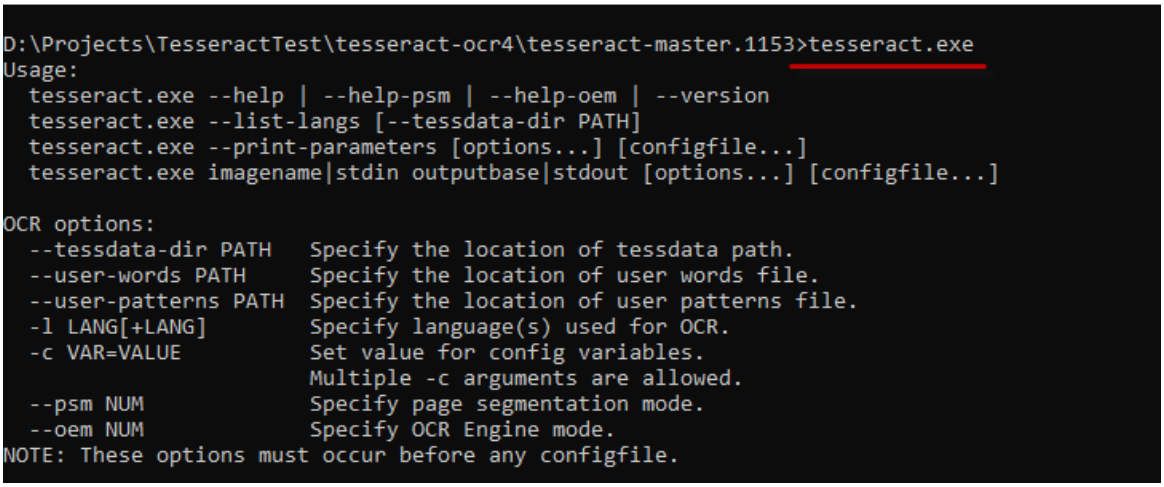

Figure 2. Run the tesseract.exe program

After starting the instructions for use. Here in the Usage section it is shown how to use the program:

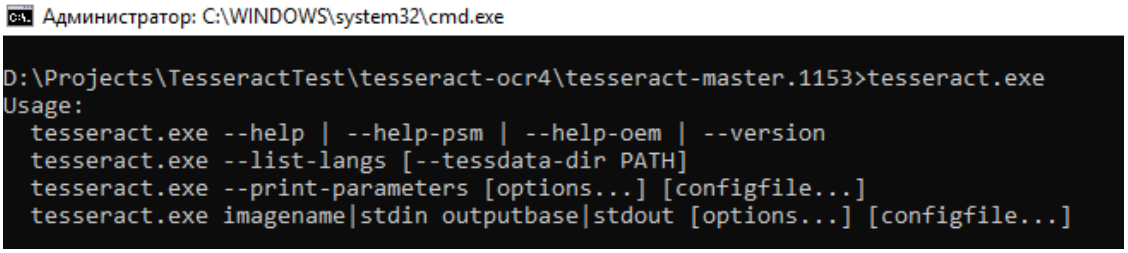

Figure 3: Using the tesseract.exe program

The tesseract.exe --help command returns a list of commands and their use.

The command tesseract.exe --list-langs returns the list of language packs located in the tessdata folder. Language packs can be downloaded at tesseract.net.

The command tesseract.exe --print-paramentrs returns a list of parameters that can be used.

The last command, tesseract.exe imagename | stdinoutputbase | stdout, extracts text from images. Here, in the place of imagename | stdin, you must enter the full path to the image, the extension must be an extension of the image (* .jpg, * .png, * .jpeg, * .bmp, etc.). In place of outputbase | stdout, the full path to the file where you want to save the text file. Most often such files are with * .txt extension.

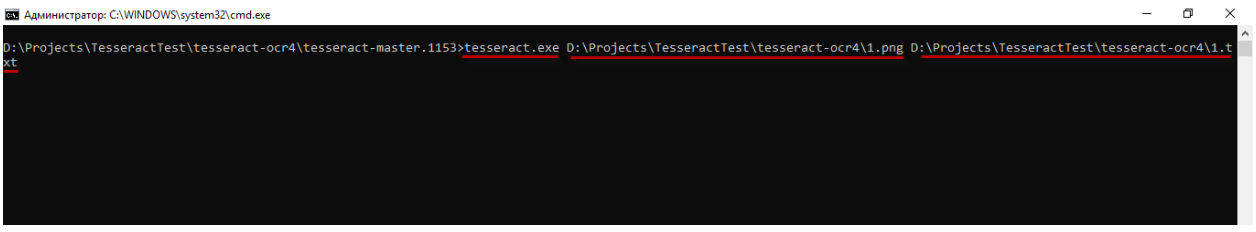

Figure 4: Running the command "tesseract.exe imagename | stdinoutputbase | stdout"

Since this library was written in C ++, it cannot be used in projects written in other programming languages. In this case, you can use the program tesseract.exe as an embedded module.

To do this, you can write a function that opens the command line, goes to the directory with the file tesseract.exe, and executes the command:

tesseract.exe imagename | stdinoutputbase | stdout

As input parameters, you need to specify the path to the tesseract.exe file, the path to the image itself, language packs necessary for the implementation of OCR. The output value is a String.

In the body of the method, the tesseract.exe process should be called via the command line.

ISSN: 2350-0328
International Journal of AdvancedResearch in Science, Engineering and Technology
Vol. 6, Issue 2, February 2019

The source code for this ParseText method in object oriented language C # looks like this:

```
privatestaticstringParseText(stringtesseractPath, stringimageFilePath, paramsstring[] lang)
{
string output = string.Empty;
vartempOutputFile = Directory.GetCurrentDirectory() + @"\tesseract-master.1153" + @"\1.txt";
vartempImageFile = imageFilePath;
MessageBox.Show("tempImageFile: " + tempImageFile + "\ntempOutputFile: " + tempOutputFile);
try
        {
ProcessStartInfo info = newProcessStartInfo();
info.WorkingDirectory = tesseractPath;
MessageBox.Show(info.WorkingDirectory);
info.WindowStyle = ProcessWindowStyle.Normal;
info.UseShellExecute = false;
info.FileName = "cmd.exe";
info.Arguments =
"/c tesseract.exe " +
// Image file.
tempImageFile + " " +
// Output file (tesseract add '.txt' at the end)
tempOutputFile +
// Languages.
" -l " + string.Join("+", lang);
MessageBox.Show(info.WorkingDirectory);
// Start tesseract.
          Process process = Process.Start(info);
process.WaitForExit();
if (process.ExitCode == 0)
          {
// Exit code: success.
             output = File.ReadAllText(tempOutputFile + ".txt");
          }
else
          {
thrownewException("Error. Tesseract stopped with an error code = " + process.ExitCode);
          }
        }
finally
        {
File.Delete(tempImageFile);
File.Delete(tempOutputFile + ".txt");
}
returnoutput;
              }
```

[2][3][4]

Having understood the Tesseract algorithm, you can write your own OCR library or a whole program that transforms an image into text.

REFERENCES

1. https://github.com/doxakis/How-to-use-tesseract-ocr-4.0-with-csharp
2. https://stackoverflow.com/questions/21086649/execute-command-line-from-a-specific-folder
3. https://stackoverflow.com/questions/1469764/run-command-prompt-commands
4. https://stackoverrun.com/ru/q/1183579