Open Source .NET Libraries for Image Processing, Recognition and Computer Vision

Štefan Toth
Department of Software Technologies, Faculty of Management Science and Informatics, University of Žilina, Univerzitná 8215/1, 010 26 Žilina, Slovakia
stefan.toth@fri.uniza.sk

Abstract. This article describes and compares two .NET libraries for image processing and computer vision. The first library, Emgu CV is a wrapper to OpenCV, which is very old but highly optimized and the most used library all over the world. The second one AForge.NET is written in pure C#. It has few features which are useful and the Emgu CV does not implement them. Both libraries have pros and cons which will be described and compared. In addition, we will briefly show our developed application which uses both libraries.

1 Introduction

Nowadays the image processing and the computer vision have become significant areas in the computer science. As developed the areas, there were created many image processing libraries which are written in the C or C++ programming language. However, in the C# language there were not developed many libraries. It is due the fact C# is a modern language created and first released in the beginning of the 21st century and also due to managed code which was not so appropriate for the time expansive image processing and computer vision algorithms. Today, the C# language is more and more popular, computers are high-performance machines and the managed code is executed more quickly than in the past (even in some cases there is managed code faster than native code). In addition, the C# and .NET have many advantages such as rapid application development, object-oriented programming language with type-safe code, automatic garbage collector and many other advanced features. Therefore using of C# for writing of image processing algorithms can be proper now.

We would like to describe two most important and well maintained open source libraries in the .NET environment. The first library, Emgu CV, is a .NET wrapper to the OpenCV library and the second one, AForge.NET, is a pure .NET solution.
2 Libraries

2.1 Emgu CV

*Emgu CV* is a multi-platform .NET library which is a wrapper to OpenCV image processing library. The OpenCV is written in the C and C++ language, while the Emgu CV is written in C#. Advantage of wrapping in Emgu CV is that it wraps or maps functions to methods and so these methods are grouped into classes logically, which make work with them simple and better, since working with object-oriented programming is more intuitive. Another advantage is the .NET IL code that means we can use library from .NET compatible languages such as C#, Visual Basic, Visual C++, IronPython etc. So we can use many advanced features of .NET as well as Emgu CV uses, i.e. generic classes and operations, serialization, operators and so on.

*OpenCV* (Open Source Computer Vision) is one of the most used libraries for the image processing, recognition and computer vision. It is used by such major companies like IBM, Microsoft, Intel, Sony, Siemens or Google. In academic area, it is used by universities and research centers like Stanford, MIT, Cambridge and others. [1]

The first release of the OpenCV library was in 1999 by Intel Research, but the official release version 1.0 was in 2006. Now the current release version is 2.3.1 from August 2011. The library is still in active progress. Now it has more than 2500 optimized algorithms written in C or C++ and runs under Windows, Linux and Mac OS. [2] It was designed for computational efficiency and with strong focus on real-time applications. [1] The OpenCV contains over 500 functions in many areas of computer vision.

![OpenCV Overview](image.png)

Fig. 1. Overview of OpenCV possibilities [2]

OpenCV is released under a BSD license, which means that the library is free for both open source and commercial use. Emgu CV uses a dual license model. The first license is under GPL v3, which allows developing application only in open source, so
it cannot be used in any commercial software. For commercial use it is needed to buy a license, which costs from $199 USD for a single developer.

The Emgu CV contains these six DLLs:

- **Emgu.CV.dll** – the core of the Emgu CV library. It contains wrapped classes to image processing or the class `CvInvoke`, by which we can directly invoke an OpenCV function.
- **Emgu.CV.UI.dll** – contains control for user interface like `ImageBox` for showing images, `HistogramBox` for showing histograms and `MatrixBox`.
- **Emgu.CV.GPU.dll** – includes a few implemented algorithms for executing on GPU (graphics processing unit) of a graphics card supported CUDA technology by Nvidia. Performance comparison between CPU and GPU is described in the article Pedestrian detection in CSharp [3].
- **Emgu.CV.ML.dll** – machine learning library. It contains such classifiers as artificial neural network, boost classifier, decision tree, expectation maximization, random tree, K-nearest classifier, Normal Bayes classifier and Support Vector Machine.
- **Emgu.CV.OCR.dll** – it is a new library in the Emgu CV and instead of OpenCV it uses tesseract-ocr library to optical character recognition.
- **Emgu.Util.dll** – collects several utilities (classes and interfaces) by Emgu projects.

---

**Fig. 2.** Architecture of Emgu CV (without Emgu.CV.OCR.dll). It has two layers of wrapper. The first layer (basic layer) contains function, structure and enumeration mapping which directly reflects those in OpenCV. The second layer contains classes that mix in advantages the .NET world. [4]
2.2 AForge.NET

AForge.NET is a .NET framework released under LGPL v3 (starting from version 2.0.0, all previous releases were published under GPL v3), which means we can use the library for both open source and proprietary software. This framework is designed for developers and researchers in the fields of computer vision and artificial intelligence for image processing, neural networks, genetic algorithms, fuzzy logic, machine learning, robotics, etc. [5]. The first version of the framework was released in 2006 and since that year it is in constant progress. Now it consists of 20 DLLs. We describe several of them:

- AForge.dll – contains core classes and structures, which can be used independently for various purposes. For example Point, DoublePoint, Range or DoubleRange structures and very useful Parallel class. The Parallel class is similar to the Parallel class in the .NET 4.0, but it has only For() method. Actually, AForge.NET is compiled to .NET 2.0 due to better multi-platform compatibility, so author of the library created own implementation of the Parallel class. Author has made some experiments and in [6] is shown that his implementation is faster than the Microsoft solution.

- AForge.Imaging.dll – library with many image processing routines and filters like image representation, filters and transformation.

- AForge.Vision.dll – classes for motion detection and processing in video streams, like blob counting or two frames difference detector.


- AForge.Neuro.dll – contains classes for neural networks representation and computations, mainly class for multi-layer neural network with activation functions and for distance network used by Kohonen self-organizing maps or elastic net.

- AForge.Robotics.*.dll (Lego, Surveyor, TeRK) – classes to control robotics various robots.

- AForge.Video.DirectShow.dll – allows to access video sources using DirectShow interface. For example for playing video or for capture devices like webcams.

- AForge.Video.VFW.dll – classes, which allow reading and writing of AVI files using old Video for Windows interface.

- AForge.Video.Kinect.dll – allows the access to video and depth data of Microsoft Xbox Kinect device.
3 Comparison of libraries

Summary description as stated above as well as other information of both libraries is given in the Table 1. Some of all the information about these libraries were obtained by our experiences.

Table 1. Summary of differences between the Emgu CV and the AForge.NET

<table>
<thead>
<tr>
<th>Features</th>
<th>Emgu CV [4] (cur. version 2.3.0.1416)</th>
<th>AForge.NET [5] (cur. version 2.2.2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>First release</td>
<td>2008 (OpenCV 1999)</td>
<td>2006</td>
</tr>
<tr>
<td>Code</td>
<td>written in C#, requires 32 or 64 bit OpenCV library</td>
<td>written in pure C#, some DLL requires external library</td>
</tr>
<tr>
<td>Dependencies (external libraries)</td>
<td>OpenCV, Tesseract, ZedGraph</td>
<td>Ice, TeRK, libfreenect, GhostAPI, ffmpeg</td>
</tr>
<tr>
<td>DLL count</td>
<td>6 + 19 externals</td>
<td>20 + 11 externals</td>
</tr>
<tr>
<td>Running on 32 &amp;&amp; 64 bit OS</td>
<td>Yes, but depends on 32 bit or 64 bit version of OpenCV libraries</td>
<td>Yes</td>
</tr>
<tr>
<td>Cross platform - compiled in Mono</td>
<td>Yes</td>
<td>Yes, but not all DLLs (without some dependencies to Windows libraries)</td>
</tr>
<tr>
<td>Minimal version of .NET platform</td>
<td>.NET 3.5</td>
<td>.NET 2.0</td>
</tr>
<tr>
<td>Image Processing</td>
<td>Yes (more algorithms)</td>
<td>Yes</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>Yes (more algorithms)</td>
<td>Yes (only neural networks)</td>
</tr>
<tr>
<td>Robotics</td>
<td>No</td>
<td>Yes (Lego, Surveyor, TeRK, Kinect)</td>
</tr>
<tr>
<td>GPU Processing</td>
<td>Yes (only for a few algorithms)</td>
<td>No</td>
</tr>
<tr>
<td>Public classes count</td>
<td>96 files / 261 files</td>
<td>255 files / 446 files</td>
</tr>
<tr>
<td>Lines count</td>
<td>6141 + external libraries</td>
<td>19660 + external libraries</td>
</tr>
<tr>
<td>Documentation and samples</td>
<td>Yes (good state)</td>
<td>Yes (very good state - detailed documentation and more published articles)</td>
</tr>
<tr>
<td>License</td>
<td>Dual - GPL v3 or commercial</td>
<td>LGPL v3</td>
</tr>
<tr>
<td>Actively maintained</td>
<td>Yes (author Canning + contributors)</td>
<td>Yes (author Andrew Kirillov + contributors)</td>
</tr>
</tbody>
</table>
As we can see in the Table 1., AForge.NET has 3 times more code lines than Emgu CV and also it has two times more number of public classes than Emgu CV. These results do not reflect real numbers, because Emgu CV is a wrapper to the big OpenCV library, whereas AForge.NET is written almost pure in C# code.

It is hard to compare, which library is better. AForge.NET has better support for video playing and recording. It has support to robotics area and provides extensive settings to neural network algorithm. However the Emgu CV or rather OpenCV is better, because it is old library which has more sophisticated and optimized algorithms for the image processing or machine learning. Despite the fact, both libraries have some pros and cons and they complement each other perfectly.

4 Demo application using libraries

We would like to present our developed software for traffic sign recognition briefly. It uses both of libraries. The Emgu CV is used as a main library and by using this library the core of traffic sign recognition (TSR) library for image process, detection and recognize a traffic signs on the road was created. We used the AForge.NET library to grab a video frame from a video source like a video file or a camera. More information about our developed application for collecting data and TSR library can be found in [7] and [8].

Fig. 3. Main screen of the application for the traffic sign recognition, in left, and screen of settings and testing of TSR, in right. Images are acquired by AForge.NET library and detection and recognition of traffic signs is made by Emgu CV library.

5 Conclusion

We described two open source libraries used for image processing, recognition and computer vision applications. Both libraries are created in the .NET and by object-oriented programming paradigm. They can be used to develop application for recogni-
tion and machine learning problems in the .NET. These developed libraries have very good documented code and there are many examples of using various algorithms. So they are useful for solving many problems and for learning in these fields.

References