

barcode image preprocessing filters in VB.NET with ByteScout BarCode Reader SDK

Make barcode image preprocessing filters in VB.NET

:

Tutorial on how to do barcode image preprocessing filters in VB.NET

Every ByteScout tool contains example VB.NET source codes that you can find here or in the folder with installed ByteScout product. ByteScout BarCode Reader SDK helps with barcode image preprocessing filters in VB.NET. ByteScout BarCode Reader SDK is the SDK for reading of barcodes from PDF, images and live camera or video. Almost every common type like Code 39, Code 128, GS1, UPC, QR Code, Datamatrix, PDF417 and many others are supported. Supports noisy and defective images and docs. Includes optional documents splitter and merger for pdf and tiff based on found barcodess. Batch mode is supported for superior performance using multiple threads. Decoded values are easily exported to JSON, CSV, XML and to custom format.

VB.NET code snippet like this for ByteScout BarCode Reader SDK works best when you need to quickly implement barcode image preprocessing filters in your VB.NET application. To do barcode image preprocessing filters in your VB.NET project or application you may simply copy & paste the code and then run your app! Code testing will allow the function to be tested and work properly with your data.

On our website you may get trial version of ByteScout BarCode Reader SDK for free. Source code samples are included to help you with your VB.NET application.

ImagePreprocessingFilters.VS2005.vbproj

```
<?xml version="1.0" encoding="utf-8"?>
<Project DefaultTargets="Build" xmlns="http://schemas.microsoft.com/developer/msbuild/2003"
  <PropertyGroup>
    <Configuration Condition="'$(Configuration)' == '' ">Debug</Configuration>
    <Platform Condition="'$(Platform)' == '' ">AnyCPU</Platform>
    <ProductVersion>8.0.50727</ProductVersion>
    <SchemaVersion>2.0</SchemaVersion>
    <ProjectGuid>{75D64BA4-2691-4BBA-A89D-B575EE05AB63}</ProjectGuid>
    <OutputType>Exe</OutputType>
    <StartupObject>ImagePreprocessingFilters.Module1</StartupObject>
    <RootNamespace>ImagePreprocessingFilters</RootNamespace>
    <AssemblyName>ImagePreprocessingFilters</AssemblyName>
    <MyType>Console</MyType>
  </PropertyGroup>
  <PropertyGroup Condition="'$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
```

```

<DebugSymbols>>true</DebugSymbols>
<DebugType>full</DebugType>
<DefineDebug>>true</DefineDebug>
<DefineTrace>>true</DefineTrace>
<OutputPath>bin\Debug\</OutputPath>
<DocumentationFile>ImagePreprocessingFilters.xml</DocumentationFile>
<NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
</PropertyGroup>
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
  <DebugType>pdbonly</DebugType>
  <DefineDebug>>false</DefineDebug>
  <DefineTrace>>true</DefineTrace>
  <Optimize>>true</Optimize>
  <OutputPath>bin\Release\</OutputPath>
  <DocumentationFile>ImagePreprocessingFilters.xml</DocumentationFile>
  <NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
</PropertyGroup>
<ItemGroup>
  <Reference Include="Bytescout.BarCodeReader, Version=8.20.0.1340, Culture=neutral,
    <SpecificVersion>False</SpecificVersion>
    <HintPath>C:\Program Files\Bytescout BarCode Reader SDK\net2.00\Bytescout.BarCode
  </Reference>
  <Reference Include="System" />
  <Reference Include="System.Data" />
  <Reference Include="System.Deployment" />
  <Reference Include="System.Drawing" />
  <Reference Include="System.Xml" />
</ItemGroup>
<ItemGroup>
  <Import Include="Microsoft.VisualBasic" />
  <Import Include="System" />
  <Import Include="System.Collections" />
  <Import Include="System.Collections.Generic" />
  <Import Include="System.Data" />
  <Import Include="System.Diagnostics" />
</ItemGroup>
<ItemGroup>
  <Compile Include="Module1.vb" />
</ItemGroup>
<ItemGroup>
  <Content Include="barcode-with-large-margins.png">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
  <Content Include="low-contrast-barcode.png">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
  <Content Include="noisy-barcode.png">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
  <Content Include="too-dense-barcode.png">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
</ItemGroup>
<Import Project="$(MSBuildBinPath)\Microsoft.VisualBasic.targets" />
<!-- To modify your build process, add your task inside one of the targets below and
  Other similar extension points exist, see Microsoft.Common.targets.
<Target Name="BeforeBuild">
</Target>
<Target Name="AfterBuild">
</Target>

```

```
-->
</Project>
```

ImagePreprocessingFilters.VS2008.vbproj

```
<?xml version="1.0" encoding="utf-8"?>
<Project DefaultTargets="Build" xmlns="http://schemas.microsoft.com/developer/msbuild/2003"
  <PropertyGroup>
    <Configuration Condition="'$(Configuration)' == ''">Debug</Configuration>
    <Platform Condition="'$(Platform)' == ''">AnyCPU</Platform>
    <ProductVersion>9.0.30729</ProductVersion>
    <SchemaVersion>2.0</SchemaVersion>
    <ProjectGuid>{75D64BA4-2691-4BBA-A89D-B575EE05AB63}</ProjectGuid>
    <OutputType>Exe</OutputType>
    <StartupObject>ImagePreprocessingFilters.Module1</StartupObject>
    <RootNamespace>ImagePreprocessingFilters</RootNamespace>
    <AssemblyName>ImagePreprocessingFilters</AssemblyName>
    <MyType>Console</MyType>
    <OldToolsVersion>2.0</OldToolsVersion>
    <TargetFrameworkVersion>v3.5</TargetFrameworkVersion>
  </PropertyGroup>
  <PropertyGroup Condition="'$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
    <DebugSymbols>>true</DebugSymbols>
    <DebugType>full</DebugType>
    <DefineDebug>>true</DefineDebug>
    <DefineTrace>>true</DefineTrace>
    <OutputPath>bin\Debug\</OutputPath>
    <DocumentationFile>ImagePreprocessingFilters.xml</DocumentationFile>
    <NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
  </PropertyGroup>
  <PropertyGroup Condition="'$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
    <DebugType>pdbonly</DebugType>
    <DefineDebug>>false</DefineDebug>
    <DefineTrace>>true</DefineTrace>
    <Optimize>>true</Optimize>
    <OutputPath>bin\Release\</OutputPath>
    <DocumentationFile>ImagePreprocessingFilters.xml</DocumentationFile>
    <NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
  </PropertyGroup>
  <ItemGroup>
    <Reference Include="Bytescout.BarCodeReader, Version=8.20.0.1340, Culture=neutral,
      <SpecificVersion>False</SpecificVersion>
      <HintPath>C:\Program Files\Bytescout BarCode Reader SDK\net3.50\Bytescout.BarCode
    </Reference>
    <Reference Include="System" />
    <Reference Include="System.Data" />
    <Reference Include="System.Deployment" />
    <Reference Include="System.Drawing" />
    <Reference Include="System.Xml" />
  </ItemGroup>
  <ItemGroup>
```

```

<Import Include="Microsoft.VisualBasic" />
<Import Include="System" />
<Import Include="System.Collections" />
<Import Include="System.Collections.Generic" />
<Import Include="System.Data" />
<Import Include="System.Diagnostics" />
</ItemGroup>
<ItemGroup>
  <Compile Include="Module1.vb" />
</ItemGroup>
<ItemGroup>
  <Content Include="barcode-with-large-margins.png">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
  <Content Include="low-contrast-barcode.png">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
  <Content Include="noisy-barcode.png">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
  <Content Include="too-dense-barcode.png">
    <CopyToOutputDirectory>Always</CopyToOutputDirectory>
  </Content>
</ItemGroup>
<Import Project="$(MSBuildToolsPath)\Microsoft.VisualBasic.targets" />
<!-- To modify your build process, add your task inside one of the targets below and
      Other similar extension points exist, see Microsoft.Common.targets.
-->
<Target Name="BeforeBuild">
</Target>
<Target Name="AfterBuild">
</Target>
-->
</Project>

```

ImagePreprocessingFilters.VS2010.vbproj

```

<?xml version="1.0" encoding="utf-8"?>
<Project DefaultTargets="Build" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <PropertyGroup>
    <Configuration Condition="'$(Configuration)' == ''">Debug</Configuration>
    <Platform Condition="'$(Platform)' == ''">AnyCPU</Platform>
    <ProductVersion>
</ProductVersion>
    <SchemaVersion>2.0</SchemaVersion>
    <ProjectGuid>{75D64BA4-2691-4BBA-A89D-B575EE05AB63}</ProjectGuid>
    <OutputType>Exe</OutputType>
    <StartupObject>ImagePreprocessingFilters.Module1</StartupObject>
    <RootNamespace>ImagePreprocessingFilters</RootNamespace>
    <AssemblyName>ImagePreprocessingFilters</AssemblyName>
    <MyType>Console</MyType>
    <OldToolsVersion>3.5</OldToolsVersion>

```

```

    <TargetFrameworkVersion>v4.0</TargetFrameworkVersion>
</PropertyGroup>
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Debug|AnyCPU' ">
    <DebugSymbols>>true</DebugSymbols>
    <DebugType>full</DebugType>
    <DefineDebug>>true</DefineDebug>
    <DefineTrace>>true</DefineTrace>
    <OutputPath>bin\Debug\</OutputPath>
    <DocumentationFile>ImagePreprocessingFilters.xml</DocumentationFile>
    <NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
</PropertyGroup>
<PropertyGroup Condition=" '$(Configuration)|$(Platform)' == 'Release|AnyCPU' ">
    <DebugType>pdbonly</DebugType>
    <DefineDebug>>false</DefineDebug>
    <DefineTrace>>true</DefineTrace>
    <Optimize>>true</Optimize>
    <OutputPath>bin\Release\</OutputPath>
    <DocumentationFile>ImagePreprocessingFilters.xml</DocumentationFile>
    <NoWarn>42016,41999,42017,42018,42019,42032,42036,42020,42021,42022</NoWarn>
</PropertyGroup>
<ItemGroup>
    <Reference Include="Bytescout.BarCodeReader, Version=8.20.0.1340, Culture=neutral,
        <SpecificVersion>False</SpecificVersion>
        <HintPath>C:\Program Files\Bytescout Bar Code Reader SDK\net4.00\Bytescout.BarCode
    </Reference>
    <Reference Include="System" />
    <Reference Include="System.Data" />
    <Reference Include="System.Deployment" />
    <Reference Include="System.Drawing" />
    <Reference Include="System.Xml" />
</ItemGroup>
<ItemGroup>
    <Import Include="Microsoft.VisualBasic" />
    <Import Include="System" />
    <Import Include="System.Collections" />
    <Import Include="System.Collections.Generic" />
    <Import Include="System.Data" />
    <Import Include="System.Diagnostics" />
</ItemGroup>
<ItemGroup>
    <Compile Include="Module1.vb" />
</ItemGroup>
<ItemGroup>
    <Content Include="barcode-with-large-margins.png">
        <CopyToOutputDirectory>Always</CopyToOutputDirectory>
    </Content>
    <Content Include="low-contrast-barcode.png">
        <CopyToOutputDirectory>Always</CopyToOutputDirectory>
    </Content>
    <Content Include="noisy-barcode.png">
        <CopyToOutputDirectory>Always</CopyToOutputDirectory>
    </Content>
    <Content Include="too-dense-barcode.png">
        <CopyToOutputDirectory>Always</CopyToOutputDirectory>
    </Content>
</ItemGroup>
<Import Project="$(MSBuildToolsPath)\Microsoft.VisualBasic.targets" />
<!-- To modify your build process, add your task inside one of the targets below and
    Other similar extension points exist, see Microsoft.Common.targets.
<Target Name="BeforeBuild">

```

```
</Target>
<Target Name="AfterBuild">
</Target>
-->
</Project>
```

Module1.vb

```
Imports Bytescout.BarCodeReader

Module Module1
    ''' <summary>
    ''' This exmample demonstrates the use of image filters to improve the decoding or s
    ''' </summary>
    Sub Main()

        Using reader As New Reader("demo", "demo")

            ' Set barcode type to find
            reader.BarcodeTypesToFind.Code128 = True

            ' WORKING WITH LOW CONTRAST BARCODE IMAGES

            ' Add the contrast adjustment for the low contrast image
            reader.ImagePreprocessingFilters.AddContrast(40)

            Console.WriteLine("Image {0}", "low-contrast-barcode.png")

            Dim barcodes As FoundBarcode() = reader.ReadFrom("low-contrast-barcode.png")

            If barcodes.Length = 0 Then
                Console.WriteLine("No barcode found!")
            Else
                For Each barcode As FoundBarcode In barcodes
                    Console.WriteLine("Found barcode {0} with value '{1}'", barcode.Type, barcode.Value)
                Next
            End If

            reader.ImagePreprocessingFilters.Clear()
            Console.WriteLine()

            ' WORKING WITH NOISY BARCODE IMAGES

            ' Add the median filter to lower the noise
            reader.ImagePreprocessingFilters.AddMedian()

            Console.WriteLine("Image {0}", "noisy-barcode.png")

            barcodes = reader.ReadFrom("noisy-barcode.png")
```

```

If barcodes.Length = 0 Then
    Console.WriteLine("No barcode found!")
Else
    For Each barcode As FoundBarcode In barcodes
        Console.WriteLine("Found barcode {0} with value '{1}'", barcode.Type, barcode.Value)
    Next
End If

reader.ImagePreprocessingFilters.Clear()
Console.WriteLine()

' WORKING WITH DENSE AND ILLEGIBLE BARCODES

' Add the scale filter to enlarge the barcode to make gaps between bars more visible
reader.ImagePreprocessingFilters.AddScale(2) ' enlarge twice

Console.WriteLine("Image {0}", "too-dense-barcode.png")

barcodes = reader.ReadFrom("too-dense-barcode.png")

If barcodes.Length = 0 Then
    Console.WriteLine("No barcode found!")
Else
    For Each barcode As FoundBarcode In barcodes
        Console.WriteLine("Found barcode {0} with value '{1}'", barcode.Type, barcode.Value)
    Next
End If

reader.ImagePreprocessingFilters.Clear()
Console.WriteLine()

' REMOVE EMPTY MARGINS FROM IMAGE TO SPEED UP THE PROCESSING

' Add the crop filter to cut off empty margins from the image.
' This will not improve the recognition quality but may speed up the process
' if you enabled multiple barcode types to search.
reader.ImagePreprocessingFilters.AddCropDark()

Console.WriteLine("Image {0}", "barcode-with-large-margins.png")

barcodes = reader.ReadFrom("barcode-with-large-margins.png")

If barcodes.Length = 0 Then
    Console.WriteLine("No barcode found!")
Else
    For Each barcode As FoundBarcode In barcodes
        Console.WriteLine("Found barcode {0} with value '{1}'", barcode.Type, barcode.Value)
    Next
End If

reader.ImagePreprocessingFilters.Clear()
Console.WriteLine()

End Using

Console.WriteLine("Press any key to exit..")
Console.ReadKey()

```

End Sub

End Module

FOR MORE INFORMATION AND FREE TRIAL:

[Download Free Trial SDK \(on-premise version\)](#)

[Read more about ByteScout BarCode Reader SDK](#)

[Explore documentation](#)

[Visit www.ByteScout.com](#)

or

[Get Your Free API Key for www.PDF.co Web API](#)

www.bytescout.com