

How to read text from noisy image with PDF extractor SDK in C# and ByteScout PDF Suite

Step-by-step tutorial on how to read text from noisy image with PDF extractor SDK in C#

Source code documentation samples give simple and easy method to install a needed feature into your application. Want to read text from noisy image with PDF extractor SDK in your C# app? ByteScout PDF Suite is designed for it. ByteScout PDF Suite is the bundle that provides six different SDK libraries to work with PDF from generating rich PDF reports to extracting data from PDF documents and converting them to HTML. This bundle includes PDF (Generator) SDK, PDF Renderer SDK, PDF Extractor SDK, PDF to HTML SDK, PDF Viewer SDK and PDF Generator SDK for Javascript.

Want to quickly learn? This fast application programming interfaces of ByteScout PDF Suite for C# plus the guidelines and the code below will help you quickly learn how to read text from noisy image with PDF extractor SDK. Just copy and paste the code into your C# application's code and follow the instructions. Applying C# application mostly includes various stages of the software development so even if the functionality works please test it with your data and the production environment.

ByteScout PDF Suite free trial version is available on our website. C# and other programming languages are supported.

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout PDF Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout PDF Suite](#)

[Get Free API key for Web API](#)

[visit www.Bytescout.com](http://www.Bytescout.com)

Source Code Files:

```

using ByteScout.PDFExtractor;
using System;

namespace ReadTextFromNoisyImage
{
    class Program
    {
        static void Main(string[] args)
        {
            try
            {
                //Read all text from noisy image file
                using (TextExtractor extractor = new TextExtractor())
                {
                    // Load noisy image document
                    extractor.LoadDocumentFromFile("sample.png");

                    // Set the font repairing OCR mode
                    extractor.OCRMode = OCRMode.TextFromImagesAndVectorsAndRepairedFont;

                    // Set the location of OCR language data files
                    extractor.OCRLanguageDataFolder = @"c:\Program Files\ByteScout PDF";

                    // Set OCR language
                    extractor.OCRLanguage = "eng"; // "eng" for english, "deu" for German
                                                    // Find more language files at https://www.bytescout.com/ocr-languages/

                    // Set document rendering resolution
                    extractor.OCRResolution = 300;

                    // You can also apply various preprocessing filters
                    // to improve the recognition on low-quality scans.

                    Console.WriteLine("Please wait while PDF Extractor processes the document.");

                    // Automatically deskew skewed scans
                    extractor.OCRImagePreprocessingFilters.AddDeskew();

                    // Remove vertical or horizontal lines (sometimes helps to avoid OCR errors)
                    //extractor.OCRImagePreprocessingFilters.AddVerticalLinesRemover();
                    //extractor.OCRImagePreprocessingFilters.AddHorizontalLinesRemover();

                    // Repair broken letters
                    extractor.OCRImagePreprocessingFilters.AddDilate();

                    // Remove noise
                    extractor.OCRImagePreprocessingFilters.AddMedian();

                    // Apply Gamma Correction
                    extractor.OCRImagePreprocessingFilters.AddGammaCorrection();

                    // Add Contrast
                    extractor.OCRImagePreprocessingFilters.AddContrast(20);
                }
            }
        }
    }
}

```

```
        // (!) You can use new OCRAnalyser class to find an optimal set of
        // filters for your specific document.
        // See "OCR Analyser" example.

        //Read all text
        string allText = extractor.GetText();

        Console.Clear();
        Console.WriteLine("Extracted Text: \n\n" + allText);
    }
}
catch (Exception ex)
{
    Console.Clear();
    Console.WriteLine("Exception: " + ex.Message);
}

Console.ReadLine();
}
}
```

VIDEO

<https://www.youtube.com/watch?v=NEwNs2b9YN8>

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout PDF Suite Home Page](#)
[Explore ByteScout PDF Suite Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout PDF Suite Online Training](#)

ON-DEMAND REST WEB API

[Get Your API Key](#)
[Explore Web API Docs](#)
[Explore Web API Samples](#)

[visit www.ByteScout.com](http://www.ByteScout.com)

[visit www.PDF.co](http://www.PDF.co)

www.bytescout.com