

## HOWTO: How to remove lines to help OCR

Many documents, especially forms, have boxes around areas of text. If there are too many, they can interfere with how OCR engines find text. The problem is worse if you are doing zonal OCR on an area that is surrounded and separated by form lines.

The DotImage OCR Module combined with the Advanced Document Cleanup from DotImage Document imaging gives you an easy way to remove these lines.

This function adds an event handler to an OcrEngine that is called right before the image is sent to be recognized.

### C#

```
private void AddLineRemovalOnSendOff(OcrEngine eng) { eng.ImageSendOff += new  
OcrImagePreprocessingEventHandler(OnImgSendOff); }
```

In the handler, you can do any image processing you want to the incoming image and it will not affect the original -- just the image that the OCR engine sees. Here's how you do a simple line removal.

### C#

```
void OnImgSendOff(object sender, OcrImagePreprocessingEventArgs e) { AtalaImage img =  
(AtalaImage)e.ImageIn.Clone(); // LineRemoval requires a 1 bit image if (img.PixelFormat !=  
PixelFormat.Pixel1bppIndexed) { img =  
img.GetChangedPixelFormat(PixelFormat.Pixel1bppIndexed); } // LineRemovalCommand has  
properties that let you control the // line length and other factors // You can set those  
properties to control which lines are removed img = new  
LineRemovalCommand().Apply(img).Image; // Setting e.ImageOut will cause the OCR Module // to  
use this image instead of the one // provided from the original Image Source e.ImageOut =  
img; }
```

Original Article:

Q10296 - HOWTO: How to remove lines to help OCR

Atalasoft Knowledge Base

<https://www.atalasoft.com/kb2/KB/50179/HOWTO-How-to-remove-lines-to-help-OC...>