Improve OCR Accuracy on Color Documents

Use Image Detergent™ to Clean Up Color Document Images Prior to OCR for Improved Results

Abstract

This white paper confirms that industry-standard practices to clean color document images can be improved to produce higher OCR accuracy. Image Detergent™ from Accusoft Pegasus improves OCR accuracy by 5-10% more than a standard Smoothing filter. This white paper leads the reader through the testing that proves it.

Standard smoothing algorithms provide a good way to reduce background noise and improve the appearance of scanned documents. However, they are also highly destructive to text and other data commonly found on a document image. The Image Detergent filter within the ScanFix® Xpress software development kit (SDK) from Accusoft Pegasus works on a different principle than other smoothing filters, and is intended specifically for use on color document images. This paper explores the impact of the Image Detergent smoothing filter on color document images containing various text and background colors. The two items quantitatively measured were OCR accuracy and cleaned up file size. OCR accuracy was measurably improved using Image Detergent, and file sizes for both lossless and lossy compression methods were significantly smaller for the images after processing with Image Detergent.

Image Detergent is only available within ScanFix Xpress from Accusoft Pegasus. A trial version can be downloaded here: http://www.accusoft.com/scanfix.htm.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

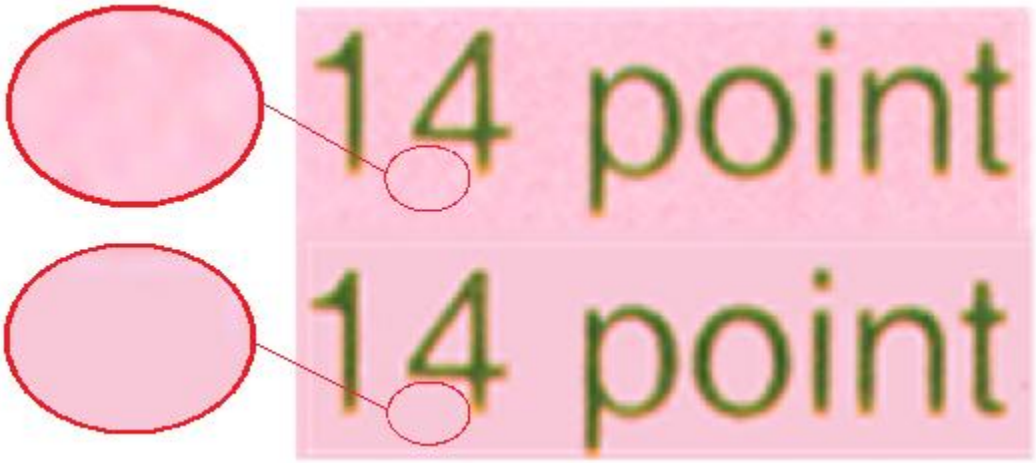

Figure 1 Before (top) and after (bottom) clips from an image cleaned with Image Detergent

Introduction

Noise is a common problem in all areas of digital signal processing, and the realm of document imaging is no exception. Noise in images typically shows up as specks or variations in color where none is desirable. An example of this can be seen above; the pink background in the top half of the image consists of many different shades of pink. This variation contains no useful information, and could be replaced by a solid color, as in the bottom half. Noise comes from many sources, including physical sources such as variations in paper color, or dirt and lint on the scanner or paper, as well as software sources, such as lossy compression. Lossy compression, such as used by the JPEG compression algorithm, produces very high degrees of compression, but at the cost of changing the pixels in the image—hence the "lossy" description. One of the common side effects of the JPEG compression is specks that surround sharp transitions in color.

Smoothing filters are a common way to remove noise from color images, by blending adjacent pixels together to dilute the impact of small changes in the colors of a region. This technique is known in signal processing as a low pass filter, because it allows low frequency data, which changes gradually over time, to pass through, and blocks high frequency data, which changes quickly. In image processing, a smoothing filter, like most other filters, is a type of morphological filter, meaning it changes the morphology or shape of the image. The morphological filters, in addition to the desirable behavior of reducing noise, also have the side effect of blurring and softening sharp edges, as shown below.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

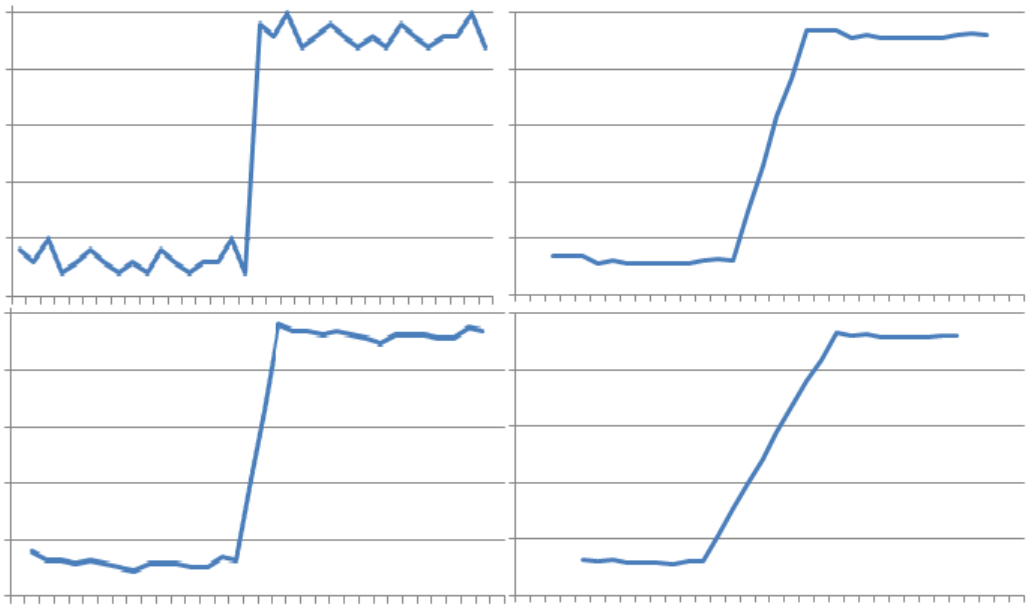

Figure 2 A noisy signal (upper left) smoothed with progressively more aggressive smoothing filters (lower left, top right, lower right) shows less and less noise, but more and more flattening of the sharp edge in the original signal.

This may not be a concern in images such as photographs, which have little high frequency data, but in document images, which are full of sharp edges and thin lines—both of which contain many high frequencies—a low pass filter can be very destructive. The Image Detergent filter is designed to address exactly this situation.

Rather than working morphologically, by blending pixels found in the same region in the image, Image Detergent works in the colorspace to provide smoothing. A colorspace is best envisioned as a cube, with an X, Y, and Z axis. In the most common colorspace, RGB, the axes correspond to the percentage of red, green, and blue that are mixed together to form any color in the colorspace. Image Detergent works in the RGB colorspace, taking colors that are in close proximity in the colorspace, and pulling them together. Since the location of the pixels in the image is not changed, no blurring of lines or edges occurs, but the regions of colors cleaned by Image Detergent end up smooth and noise-free.

In this paper, the impact of Image Detergent on various document processing operations will be explored. The baseline data set consists of a number of documents, the "raw" images, which have been captured on a color scanner and stored in an uncompressed format. These raw images will be filtered and compressed using different image processing methods, and the resulting modified images will be compared to determine the impacts of these image processing methods on optical character recognition and file size using common lossy and lossless compression algorithms.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

The Hypothesis for OCR Recognition

Image Detergent will improve OCR recognition, and reduce the image size of compressed images.

The Approach

Binarize original images, apply OCR and measure recognition accuracy. Next, apply Image Detergent to those same original images, then binarize them, apply OCR, and measure recognition accuracy. Compare against initial results. Take the same approach on image file size.

The Data Set

The data set consists of a number of documents printed on a color laser printer on colored paper. Each document has the same text repeated in differing font faces using 14 and 12 point text.

Paper color	Font face	Ink color
White	Arial	Black
Ivory	Times New Roman	Red
Blue	Courier	Green
Pink		Blue
Yellow		

The text consisted of the following, reproduced for all desired combinations:

> 14 point Color scanning gives you more information. Now you can use it.
> The quick brown fox jumped over the lazy hen. Now is the time for all
> good men to come to the aid of their country.
>
> ABCDEFGHIJKLMNOPQRSTUVWXYZ
> abcdefghijklmnopqrstuvwxyz
> 1234567890
> ~!@#$%^&*()_+

The documents were scanned using an older Fujitsu® 4750C color scanner to simulate a common production environment. The documents were scanned at 150, 200, and 300 dpi, but the 200 and 300 dpi images were re-sampled to 150 dpi for this test. The images were stored in a lossless TIFF format. The total number of scanned pages used in the

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

analysis was 180. The scanner default settings automatically adjusted the brightness on the Ivory and White paper, resulting in a background color whose average was very near pure white. The blue, pink, and yellow papers yielded colors of approximately 0xb0f0f8, 0xf8c8d8, and 0xf8f8c8.

The resolution of 150 dpi was determined to be the borderline resolution for reliable OCR of 12 point and larger text, and the analysis focused on the 12 and 14 point text. By focusing on the borderline cases, changes in OCR accuracy are readily determined. The image shown in Figure 1 of this white paper consists of green ink on pink paper, and it provided a good illustration of the difficulties encountered when recognizing characters from a color image.

Cleanup method

The method used to generate the data for this analysis used three software development toolkits from Accusoft Pegasus. Each toolkit is focused on a different imaging field, and all the toolkits are designed to easily work together. The ImagXpress toolkit is a general purpose photo and document imaging SDK, and it is used for opening and decompressing the images. ImagXpress supports a wide range of file formats (see www.accusoft.com/imagxpressformats.htm), including popular document imaging formats such as TIFF and JBIG2, and color formats such as JPEG and JPEG 2000. SmartZone™, part of the FormSuite™ SDK, is a zonal OCR tool with an easy to use programmer interface (www.accusoft.com/formsuite.htm). The third toolkit used in this test is ScanFix Xpress, the scanned image cleanup toolkit.

Each image was loaded into memory using ImagXpress and subjected to two different processes. The first process was a simple binarization, using the ScanFix Binarize() method, with the minimum and maximum thresholds set to 128 and 255 respectively. The image was then processed using SmartZone, and the resulting text stored for comparison.

The second process used the ScanFix Image Detergent method to clean up the image background color. The Image Detergent object was set up to clean four colors, corresponding to pure white, and the average background colors for the blue, pink, and yellow page as estimated using an industry standard utility program. This method was chosen for simplicity, and because it allowed a greater control over the radius than the AutoImage Detergent() method. The radius used for each color was around half of the distance from each color to its nearest neighbor, to prevent overlapping cleanup regions.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

Here is a diagram of the cleanup operation, showing the effected regions of the spectrum:

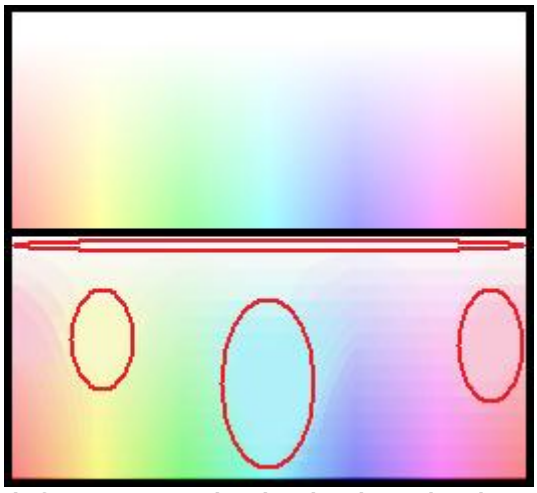

Figure 2 Before and after spectrum showing the cleaned regions of the colorspace.

After the Image Detergent operation, the image was binarized with the same settings as the first process.

Test Results:

By providing a cleaner input, the binarization process produced a better quality output, which increased the recognition rate when the images were read using SmartZone.

OCR impact

Figure 3 Binarizing alone (left) compared to Image Detergent followed by binarization (right). The thin areas of the characters proved difficult for the binarization process, but a cleaned image resulted in fewer broken characters.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

The OCR impact of the Image Detergent process was calculated by comparing the output from each process, and counting the number of characters correctly matched, and subtracting one for characters that were dropped, added, or misread. Overall, recognition rates were 2% higher for 14 point and 5% higher for 12 point text with the Image Detergent operation than without it. While the 2% increase seems small, it translated into a 63% increase in the number of blocks of 14 point text read with 95% or greater accuracy. For 12 point text, it corresponded to a 21% increase in blocks read with over 80% accuracy. The reason for this change was the reduction in noise, and relative increase in contrast between the background and foreground colors.

Comparison with Standard Smoothing Algorithm

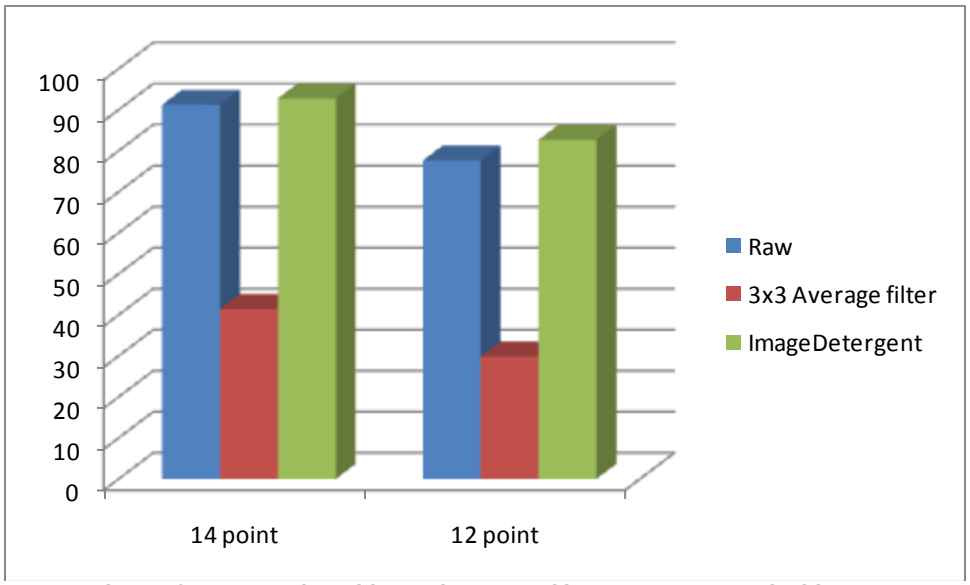

Figure 4 Comparison of OCR results with raw image and images processed with a 3x3 Average filter and the Image Detergent filter.

Compared to other smoothing algorithms however, the results are dramatically different. Replacing the Image Detergent filter with a standard smoothing filter (a 3x3 pixel average filter) also reduced the background noise, but at great cost to the machine readability of the characters in the documents. Readability dropped from an averate of 93% with Image Detergent to 41% with the average filter. The reason for this is that while smoothing does even out the background color, it also significantly deteriorates the sharpness of the characters. This decrease in sharpness leads to dramatically poorer performance when the image is binarized, as shown in Figures 4 and 5.

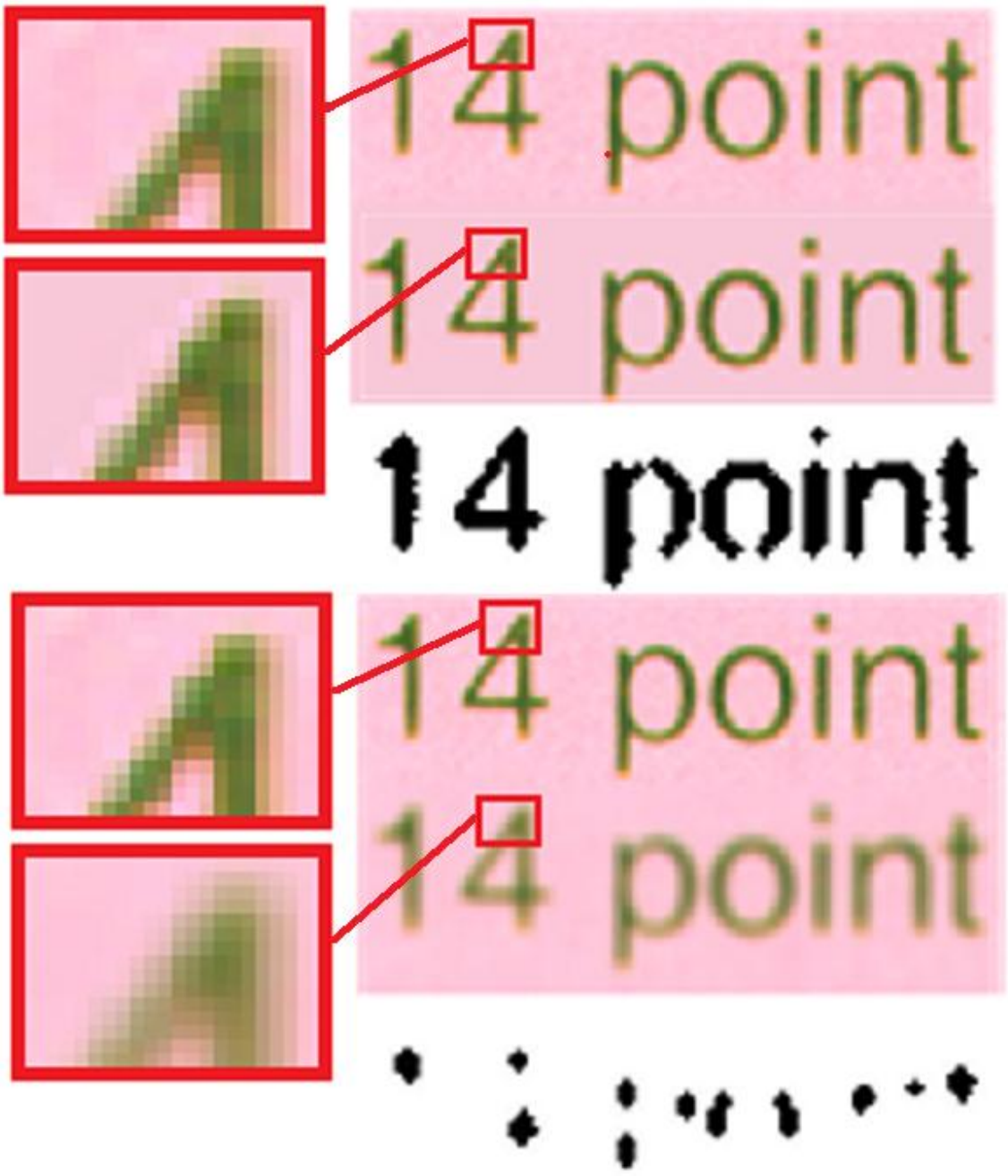

Figure 5 Image Detergent (top) compared to a 3 x 3 pixel average filter (bottom). Both clean the background, but the blurred text is far more difficult to binarize, as it has greatly reduced contrast with the background.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

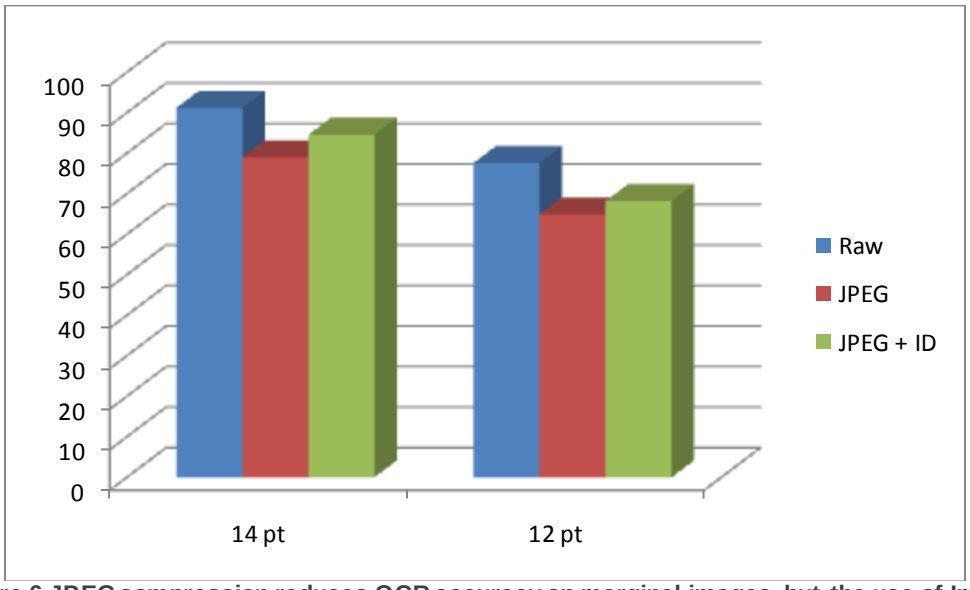

Figure 6 JPEG compression reduces OCR accuracy on marginal images, but the use of Image Detergent can regain a significant percentage of the lost accuracy.

As shown in Figure 6, when the test images were compressed as 50% quality JPEG images, the OCR quality went down for both the images with and without Image Detergent, as would be expected for such marginal images. However, performance on the non-Image Detergent images declined more; recognition dropped 12% on 14 and 13% on 12 point text when JPEG compression was used. When Image Detergent was applied to the JPEG images after decompression and before OCR, it significantly reduced the negative impact of the JPEG artifacts, reducing the impact of the JPEG degradation by as much as 70%, dropping those percentages to 7% and 9% respectively. Figure 7 shows the artifacts produced by JPEG's compression algorithm, and how Image Detergent removes many of those artifacts from the image.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705
accusoft.com

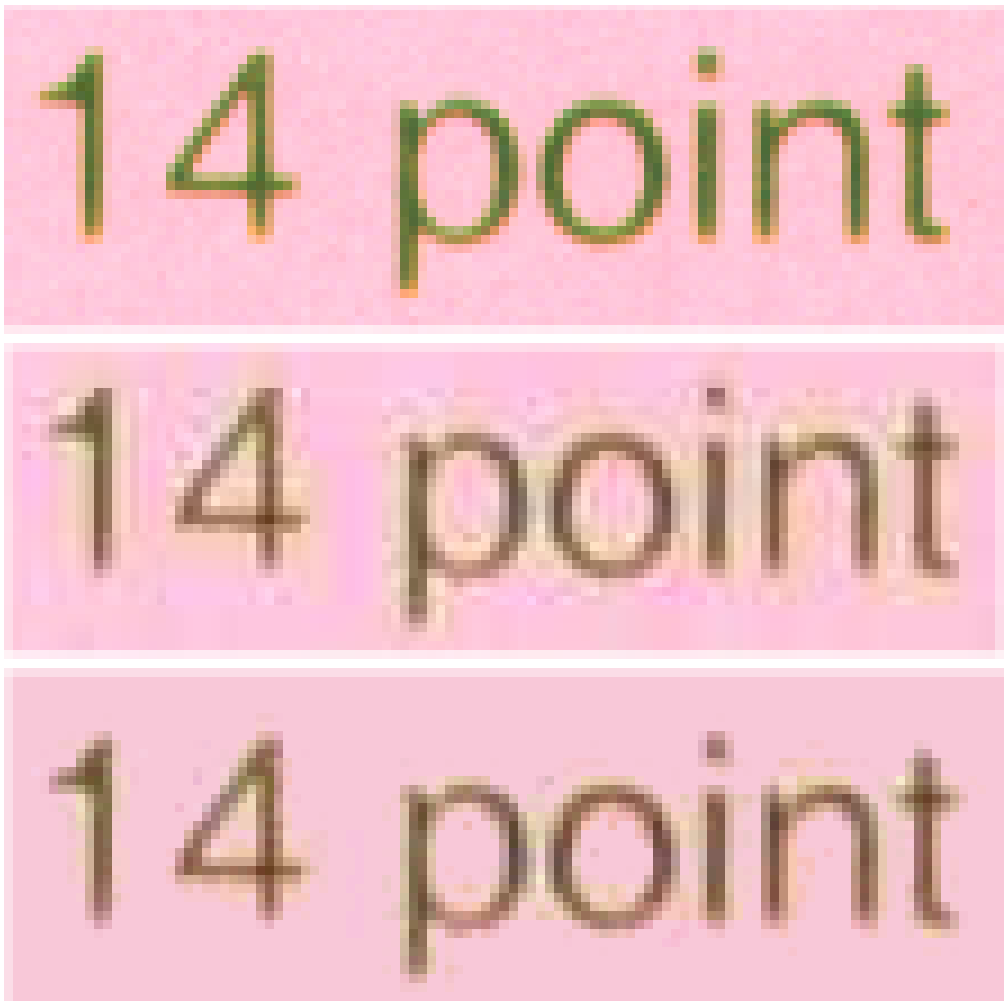

Figure 7 Original (top), 50% quality JPEG (middle), and cleaned JPEG (bottom). The JPEG compression actually cleans the background some, as the lossy compression algorithm discards the high frequency data, but it also creates noise in the form of compression artifacts, which appear as specks and brightness variance around the characters. Image Detergent helps clean up those artifacts.

The Hypothesis for Reducing Output File Size

Noise, while not desirable, is still data, and can significantly impact the file size in compressed image formats. Removing the noise from the image before it is compressed can reduce file sizes, in some cases very dramatically.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

The Approach

For the file size comparison, each image was cleaned with Image Detergent using the same settings that were used for OCR testing, and the results were stored in a lossless format. The original and cleaned images were then compressed using two formats, a lossy and a lossless format. The lossy format used was JPEG, using a 50% quality level, comparable to the Independent JPEG Group's reference implementation. The lossless format used was TIFF with LZW compression.

The Data Set

The images used were the same as were used for the OCR testing. Each image was compressed using the lossy and lossless formats, and the file sizes of the compressed original and cleaned images were compared to calculate the difference. The differences are shown in Figure 8, as a percentage of the original file size. A value of 90 indicates the compressed Image Detergent file was 90% the size of the compressed original file.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

Test Results

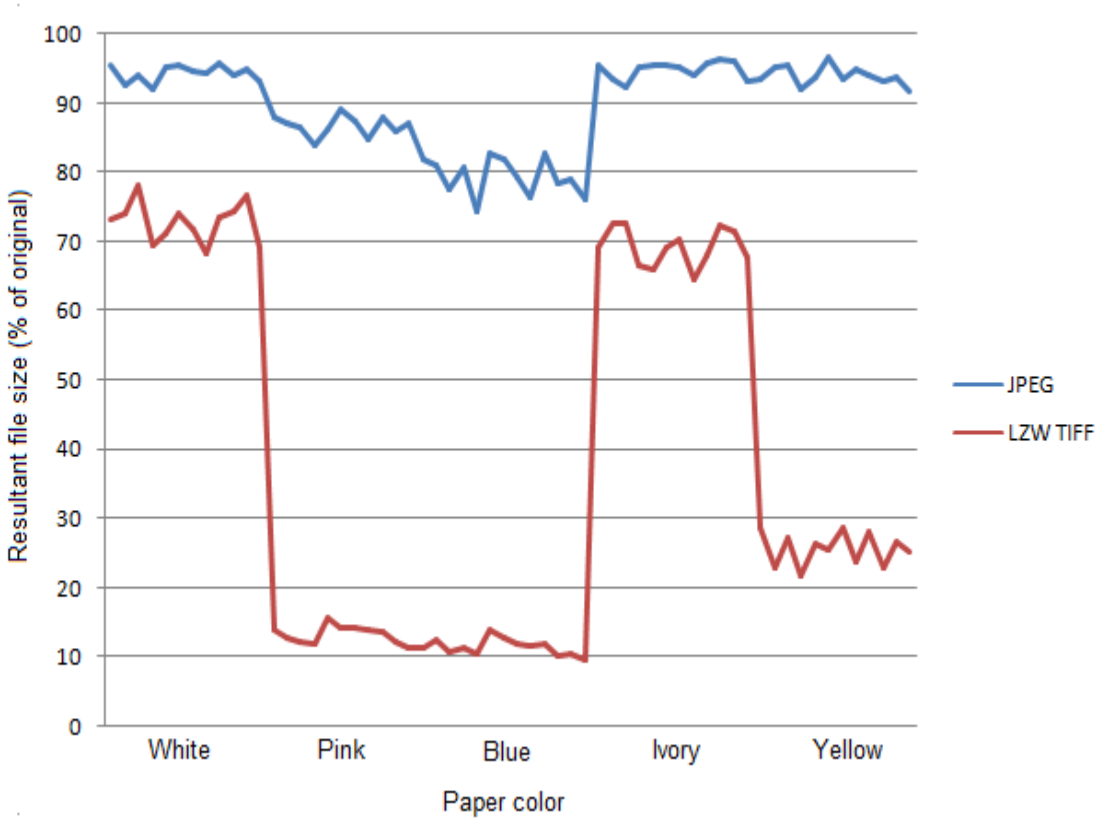

Figure 4 Percentage file size reduction with Image Detergent. In all cases, the compressed file size was reduced. The most dramatic changes were in the Pink and Blue colored pages, which had the most noise.

When comparing the sizes of the original and the cleaned images using JPEG compression at a 50% quality level, the cleaned images generated file sizes 10% smaller on average. With a lossless compression, such as LZW compressed TIFF, the change is far more dramatic, with an average reduction of 62% (though the LZW files are still significantly larger than the JPEG files).

Reductions in compressed file sizes were most dramatic on images with strong color backgrounds, which contained the most noise in the original images. With those images compression improvements of 70% to 90% were achieved using LZW. Blue showed the greatest improvement in the JPEG tests, with 20% reductions in compressed file sizes.

Conclusions

Traditional color smoothing filters are best kept to traditional color images. For document images, a different approach is needed, and the ScanFix Xpress v6 toolkit provides the tools you need. Image Detergent is just one of the many features provided to help deal with problems unique to color documents. Other features include color deskew and crop, color dropout technologies such as Color Drop and Virtual Bulb, Color Detect for sorting and segmenting images, automatic brightness and contrast adjustment, as well as a wide range of traditional color image filters for tasks such as smoothing and sharpening.

This paper has shown that using Accusoft Pegasus' ScanFix Xpress v6 SDK to apply Image Detergent to color images results in smaller files sizes for image archival, and higher accuracy of OCR recognition than standard imaging industry practices of smoothing filters.

Find a full list of ScanFix Xpress SDK image enhancement features and download a demo application or trial version of the ScanFix Xpress SDK here.

Sample code you could use to test with your own sample images is provided in the next section. Download the SmartZone and ScanFix Xpress SDKs from www.accusoft.com to execute the sample code.

Please contact us at sales@accusoft.com or support@accusoft.com for more information.

Source Code Sample

This section demonstrates how to use ScanFix in conjunction with ImagXpress and SmartZone to load, clean, and OCR a document image. This code was used to generate the data used in this whitepaper.

```
/*
 *      This is a simple C# command line appliation that accepts an image filename
 *      as a command line argument, and generates a cleaned color image, a binarized
 *      image, and a text file containing the ASCII results of the OCR operation.
 */

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            // Here, instnaces of the three components used in this sample are created
            PegasusImaging.WinForms.SmartZone2.SmartZone smartzone =
```


4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

```
   new PegasusImaging.WinForms.SmartZone2.SmartZone();
Accusoft.ScanFixXpressSdk.ScanFix scanfix =
          new Accusoft.ScanFixXpressSdk.ScanFix();
 Accusoft.ImagXpressSdk.ImagXpress imgX =
          new Accusoft.ImagXpressSdk.ImagXpress();

/*
 * Here you will need to set up the component licenses, if you have them;
 * if not, then the program will run, but will display dialog boxes asking
 * you to register the products.
*/

// create ImagXpress image object, so that we can use it to load and save
// the image data
 Accusoft.ImagXpressSdk.ImageX image;

// load the file into the image object
image = Accusoft.ImagXpressSdk.ImageX.FromFile(imgX, args[0]);

// load image into ScanFix
scanfix.FromHdib(image.ToHdib(true));

// set up the Image Detergent options structure, so we can specify the colors
// we wish to clean; in the test set, these colors are white, pink, blue,
// and yellow
Accusoft.ScanFixXpressSdk.ImageDetergentOptions idoptions =
          new Accusoft.ScanFixXpressSdk.ImageDetergentOptions();

// add blue
idoptions.ColorRadiusList.Add(new Accusoft.ScanFixXpressSdk.ColorRadius(
          System.Drawing.Color.FromArgb(0x00b0f0f8), 50, false));
// add pink
idoptions.ColorRadiusList.Add(new Accusoft.ScanFixXpressSdk.ColorRadius(
          System.Drawing.Color.FromArgb(0x00f8c8d8), 37, false));
// add yellow
idoptions.ColorRadiusList.Add(new Accusoft.ScanFixXpressSdk.ColorRadius(
          System.Drawing.Color.FromArgb(0x00f8f8c8), 25, false));
// add white
idoptions.ColorRadiusList.Add(new Accusoft.ScanFixXpressSdk.ColorRadius(
          System.Drawing.Color.FromArgb(0x00ffffff), 20, false));

// clean the image with the given ImageDetergent settings
scanfix.ImageDetergent(idoptions);

// this gets access to the image in ScanFix, without giving up control
image = Accusoft.ImagXpressSdk.ImageX.FromHdib(imgX,
          scanfix.ToHdib(false));

// save the cleaned image so we can examine it
image.Save(args[0] + "_cleaned");

// set up simple thresholding
Accusoft.ScanFixXpressSdk.BinarizeOptions binopts =
          new Accusoft.ScanFixXpressSdk.BinarizeOptions();
binopts.HighThreshold = 255;
binopts.LowThreshold = 128;

// binarize the image
scanfix.Binarize(binopts);

// this gets access to the image in ScanFix, without giving up control
```


4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705 accusoft.com

```
                image = Accusoft.ImagXpressSdk.ImageX.FromHdib(imgX,
                          scanfix.ToHdib(false));

                // save the thresholded image so we can examine it
                image.Save(args[0] + "_thresholded");

                // set up OCR recognition zone to the full page, letter size at 150 dpi
                System.Drawing.Rectangle rect = new System.Drawing.Rectangle(0, 0,
                          (int)(8.5 * 150), 11 * 150);
                smartzone.Reader.Area = rect;

                // recognize the zone, and get the results
                PegasusImaging.WinForms.SmartZone2.TextBlockResult blockresult;
                blockresult = smartzone.Reader.AnalyzeField(scanfix.ToHdib(true));

                // save the output to a text file for analysis
                if (blockresult.NumberTextLines > 0)
                {
                    // save the page text to a file
                    System.Console.Out.WriteLine(blockresult.Text);
                    System.IO.TextWriter outfile = new System.IO.StreamWriter(args[0]
                          + ".txt");
                    outfile.WriteLine(blockresult.Text);
                    outfile.Close();
                }
                else
                {
                    // note that no text was found for the given image
                    System.Console.Out.WriteLine("No text was recognized");
                    System.IO.TextWriter outfile = new System.IO.StreamWriter(args[0]
                          + ".txt");
                    outfile.WriteLine("No text recognized");
                    outfile.Close();
                }

            }
        }
    }
```

About the Author

Scot Alexander, Software Engineer, Accusoft Pegasus

Scot joined Accusoft Pegasus (Pegasus Imaging) with the acquisition of TMSSequoia in December 2004. As an important member of the team since 1994, Scot has contributed to several high performance document imaging product lines from Accusoft Pegasus, including ScanFix Xpress, ScanFix Application, and Prizm Viewer. He began working on color image processing in 1999, has a patent pending on a comb detection algorithm, and is the inventor of the Virtual Bulb, Image Detergent, and Color Drop technologies within ScanFix Xpress. In his spare time, he enjoys science fiction, Sluggy Freelance, digital image processing, and algorithmic art. Scot earned a Bachelor of Science in Computing and Information Science from Oklahoma State University.

About Accusoft Pegasus

Founded in 1991 under the corporate name Pegasus Imaging, and headquartered in Tampa, Florida, Accusoft Pegasus is the largest source for imaging software development kits (SDKs) and image viewers. Imaging technology solutions include barcode, compression, DICOM, editing, forms processing, OCR, PDF, scanning, video, and viewing. Technology is delivered for Microsoft .NET, ActiveX, Silverlight, AJAX, ASP.NET, Windows Workflow, and Java environments. Multiple 32-bit and 64-bit platforms are supported, including Windows, Windows Mobile, Linux, Solaris x86, Solaris SPARC, Mac OS X, and IBM AIX. Visit www.accusoft.com for more information.

4001 n. riverside drive | tampa, fl 33603 | p. 813.875.7575 | f. 813.875.7705

accusoft.com