INFO: Typical cause of an inverted black and white TIFF image

Each pixel in a Black and White image is represented by a single bit. A 1-bit TIFF image has no palette, and can only be black and white. An "on" bit and an "off" bit can represent either black or white depending on the photometric tag (ID262) of the TIFF image. The typical photometric orientation of a black and white TIFF image is 0 = White and 1 = Black, which corresponds to a photometric tag value of 0. The opposite, 0 = Black, 1 = White corresponds to a photometric tag value of 1. Some image viewers ignore the photometric tag completely always assuming 0 = White, most notably the Windows XP Image Viewer. To make matters even worse, many TWAIN drivers create images with 0 = Black and 1 = White. The misconception is that there is a bug in our product, when in fact it's a bug in the Windows Image Viewer!

The following code will correct the image and palette so this will no longer be a problem. The code below requires a license of DotImage.

[C#]

```
f (image.PixelFormat == PixelFormat.Pixel1bppIndexed) { // See if the 0 palette entry is less than the 1 palette entry. Color p0 = image.Palette.GetEntry(0); Color p1 = image.Palette.GetEntry(1); if ((p0.R << 16 + p0.G << 8 + p0.B) < (p1.R << 16 + p1.G << 8 + p1.B)) { // Swap the palette entries. image.Palette.SetEntry(0, p1); image.Palette.SetEntry(1, p0); // Invert the image data. Atalasoft.Imaging.ImageProcessing.Channels.InvertCommand inv = new Atalasoft.Imaging.ImageProcessing.Channels.InvertCommand(); inv.Apply(image); } }
```

[VB.NET]

```
f image.PixelFormat = PixelFormat.Pixel1bppIndexed Then ' See if the 0 palette entry is less than the 1 palette entry. Dim p0 As Color = image.Palette.GetEntry(0) Dim p1 As Color = image.Palette.GetEntry(1) If (p0.R << 16 + p0.G << 8 + p0.B) < (p1.R << 16 + p1.G << 8 + p1.B) Then ' Swap the palette entries. image.Palette.SetEntry(0, p1) image.Palette.SetEntry(1, p0) ' Invert the image data. Dim inv As Atalasoft.Imaging.ImageProcessing.Channels.InvertCommand = _

New Atalasoft.Imaging.ImageProcessing.Channels.InvertCommand() inv.Apply(image) End If End If
```

Original Article:

Q10069 - INFO: Typical cause of an inverted black and white TIFF image

INFO: Typical cause of an inverted black and white TIFF image

Atalasoft Knowledge Base
https://www.atalasoft.com/kb2/KB/50349/INFO-Typical-cause-of-an-inverted-bl...