White Paper

PTC ImageJ Plugin – Analyzing and Plotting the Photon Transfer Curve

January 2014

Authors

Dr. Daniel Van Blerkom

Chief Technology Officer & Co-Founder, Forza Silicon

Introduction

This document is a user manual for an ImageJ plugin developed by Forza Silicon to quickly analyze and plot the Photon Transfer Curve of a set of images.

- **ImageJ** is a freeware image viewing and analysis software package distributed by the NIH. It can be downloaded here: http://rsbweb.nih.gov/ij/. ImageJ is Java based, open source, runs on many platforms, and can be easily extended to perform additional analysis beyond what is included in the base package. These extensions are called "plugins".
- The **Photon Transfer Curve**, or PTC, is a useful tool for analyzing the performance of image sensors. This plugin computes the PTC, and gives a graphical plot from which the conversion gain and read noise can be fit to the PTC curve.

Inspiration for this plugin came from a course taught by Dr. Albert Theuwissen on CMOS image sensor characterization. (http://www.harvestimaging.com/)

Description

The PTC curve is a plot of temporal noise versus signal level, on a log-log scale. At high enough signal levels, the temporal noise is dominated by the photon shot noise of the optical signal, and so it increases as the square root of the signal – hence giving a slope of 1/2 in the log-log plot. At lower signal levels, the photon shot noise is below the read noise of the image sensor. The read noise is typically a signal independent noise associated with the readout signal chain from the pixel through the ADC - thus it has a slope of 0 in the log-log plot.

By fitting lines into these two regimes of the PTC, two important parameters of an image sensor can be obtained: conversion gain (in DN/e-) and read noise (in e-). The conversion gain, in digital codes (DN) per electron, describes the gain of the pixel and readout signal chain in converting collected photoelectrons to ADC output codes. This value can be obtained from the slope = 1/2 line. The read noise in DN, which is the Y-intercept of the slope = 0 line, can then be input referred using the conversion gain to give the read noise equivalent in electrons

The full-well of the pixel in electrons is easily found once the conversion gain is known, by finding the maximum signal in DN and dividing by the conversion gain. Since the linearity of the response often reduces as the pixel approaches full saturation, the definition of the usable full-well depends on the application's tolerance to non-linearity. Dividing the full-well in electrons by the read noise in electrons gives the dynamic range of the sensor.

Many other characteristics of interest can be obtained from the PTC curve. The interested reader is referred to the following references:

1) James Janesick, "Photon Transfer", SPIE Press, 2007.

2) http://harvestimaging.com/blog/

Installation

The PTC ImageJ plugin is distributed as a "jar" file. Simply copy the PTC_.jar file to the "plugins" directory in the ImageJ application directory tree, and then restart ImageJ, to load the plugin. If the plugin has been loaded successfully, there will be an item called "Forza" in the ImageJ plugins pulldown menu. Under "Forza" will be the "Photon Transfer Curve" plugin selection.

Image Preparation

The PTC plugin operates on a "stack" of 16-bit grayscale images. A "stack" in ImageJ is a collection of images of identical size and type; each one is referred to as a "slice" in the stack. ImageJ displays a stack in one window with a selector bar at the bottom of the window to select the image slice from the stack. The PTC plugin expects at least three slices in the stack it operates on – one dark frame and at least two light frames.

The first image slice in the stack must be a dark frame. This frame is subtracted from the following light frames to remove Fixed Pattern Noise (FPN) and any black level offset, when determining the signal level at each pixel.

The next frames are the light frames. To produce the best PTC results, the recommended scene is one that contains the full range of pixel output values in a smooth ramp. To be able to resolve the read noise floor, the scene should contain plenty of black, and almost black, pixels. To determine the pixel full-well, the scene should also have saturated pixels in the brightest region. The light frame images should all be of the exact same scene; typically, they would be taken one right after the other.

An example of an ideal gray image is shown in Figure 1, together with the profile plot along the horizontal yellow line shown in the image. This image has plenty of dark pixels, and also reaches pixel saturation at the right edge.

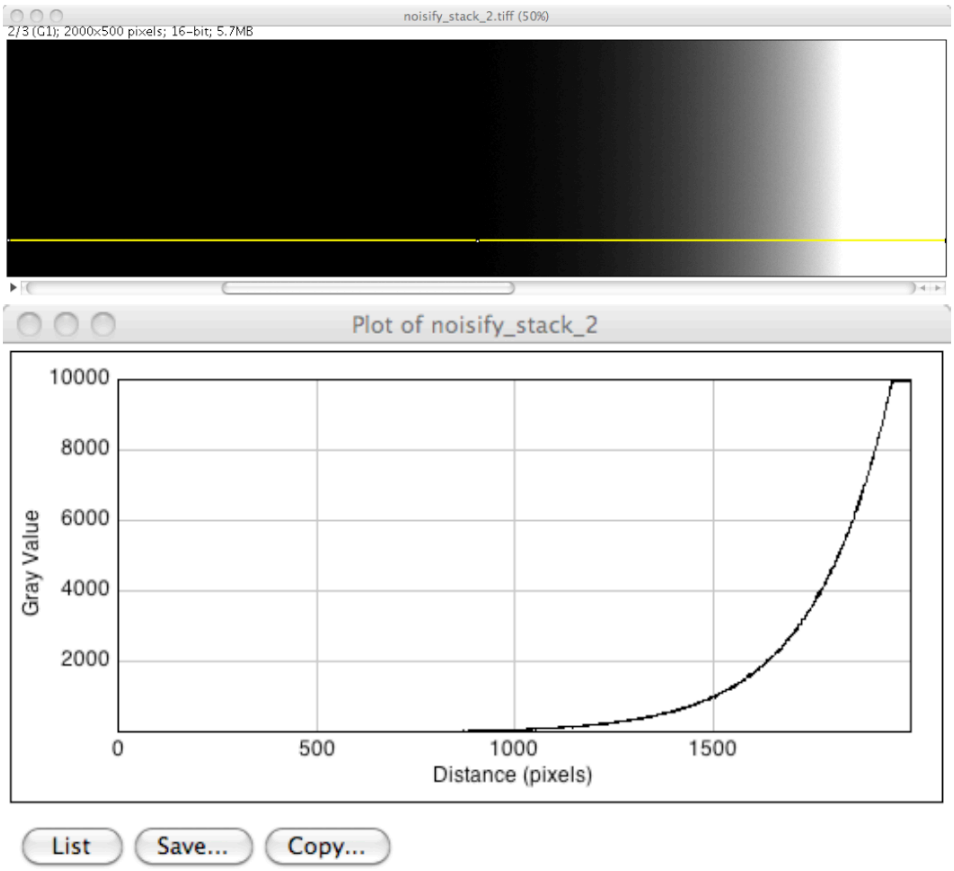

Figure 1 - Example gray level image, and profile plot of the image

The PTC plugin uses the light frames to estimate the noise at each pixel, and then bins the noise measurements based on the signal level at each pixel. The more noise samples per signal level bin, the more accurate the estimate of the noise will be for that signal level; hence the need for a gradual ramp of signal levels. The plugin also allows the user to decide the number of bins used, to trade-off the number of points plotted versus the accuracy of the noise estimation for each signal level.

Running the Plugin

Once the image stack has been created, as described on the previous page, the plugin is run by selecting “Plugins~>Forza~>Photon Transfer Curve” from the ImageJ panel.

The plugin will then prompt for the number of points to use per decade, as shown in Figure 2. This sets the signal level bin size, which determines how many pixels will be averaged to estimate the noise for that signal level range. A higher number of points per decade will give more resolution to the PTC plot, at the cost of accuracy of the noise estimation for each point. A smaller number of points per decade will increase the accuracy of the noise estimate, but will result in a coarser PTC graph.

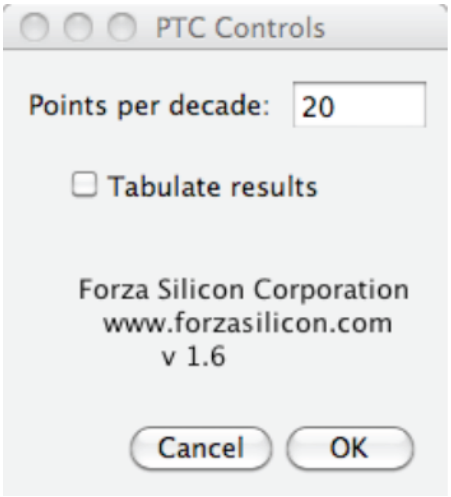

Figure 2 - PTC plugin control panel

The plugin will then analyze the images and produce the PTC graph. If “Tabulate results” was checked, a list of X-Y values that represent the points on the graph will open in another window. As described below, the PTC graph allows the user to fit lines to the read noise and shot noise regimes of the data, and gives a readout of the conversion gain and read noise in electrons. The PTC values from the tabulated results can also be copied and pasted into an Excel worksheet for further analysis.

Analyzing the PTC Results

The graph produced by the PTC plugin is shown in Figure 3.

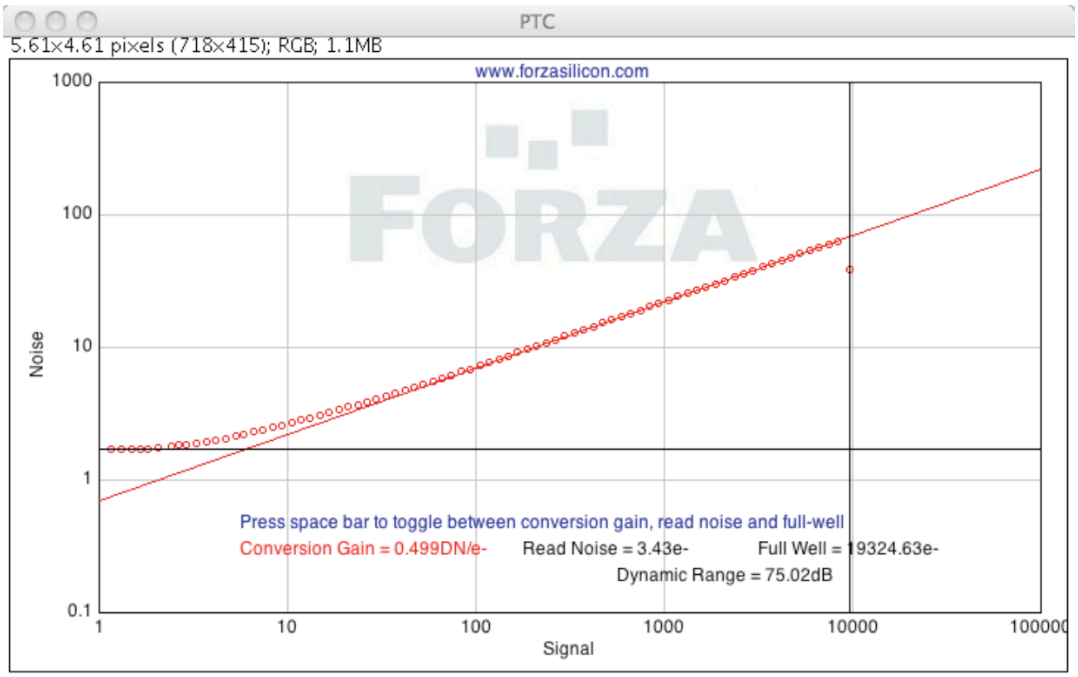

Figure 3 - Example PTC plugin output graph

The x-axis of the graph is the signal level in DN, on a log scale; and the y-axis is the noise level in DN, also on a log scale. Red circles mark the values calculated from the input images. Overlaid on the graph are three adjustable lines. When fitted to the plot points, the horizontal line finds the read noise, the vertical line finds the full-well, and the slope = ½ line finds the conversion gain. One line is adjusted at a time – the selected line is colored red, while the others are black. Clicking on the plot changes the selected line's position. Pressing the space bar allows the user to cycle through and adjust the three lines.

Once the lines have been fit to the plot, the values for the conversion gain, read noise, full-well, and dynamic range of the sensor can be read from the bottom part of the plot.

Summary

We hope you find this plugin useful. Internally, we've found the PTC curve to be a useful tool to:

- cross-check and validate other characterization results
- give quick, initial estimates of the pixel performance
- highlight non-linearity problems

For more information on the ImageJ Plugin or to provide improvements and enhancements, please email info@forzasilicon.com.

©2014 Forza Silicon, Corp. All rights reserved.