Teledyne DALSA • 880 Rue McCaffrey • St-Laurent, Québec, H4T 2C7 • Canada
http://www.teledynedalsa.com/Genie-Nano

G3-AN0006 Genie Nano-Polarized Demo Application Note

Genie Nano: Polarized Sensors Software Demo

For Nano models with P/N: G3-GM14-M2450

Overview

The first Genie Nano model with a quad-polarizer filter (M2450-Polarized) uses the Sony Pregius 5.1 MP IMX250MZR monochrome sensor. These types of sensors incorporate a nanowire polarizer filter that allows detection of both the angle and amount of polarized light. Polarized filtering can reduce the effects of reflections and glare from multiple directions and reveal otherwise undetectable features in the target scene. This application note describes the implementation of a polarization demo with several different algorithms for polarization imaging.

The Genie Nano Polarized software demo is available for download from the Teledyne DALSA ftp server as solutions for Microsoft Visual Studio 2010/12/15/17/19:

ftp://ftp.dalsa.com/Private/p_apps_eng/Demos/GigeCameraDemo_Polarized.zip

Executables are also provided for 64-bit versions using ANSI or Unicode character sets (compiled with Sapera LT 8.41).

Prerequisites

The following table lists the recommended Genie Nano firmware and software for this camera model.

Genie Nano Firmware Design	Software SDK
Standard	Sapera LT 8.31 or higher

Sapera LT SDK (full version), the image acquisition and control SDK for Teledyne DALSA cameras is available for download from the Teledyne DALSA website:

http://teledynedalsa.com/imaging/support/downloads/sdks/

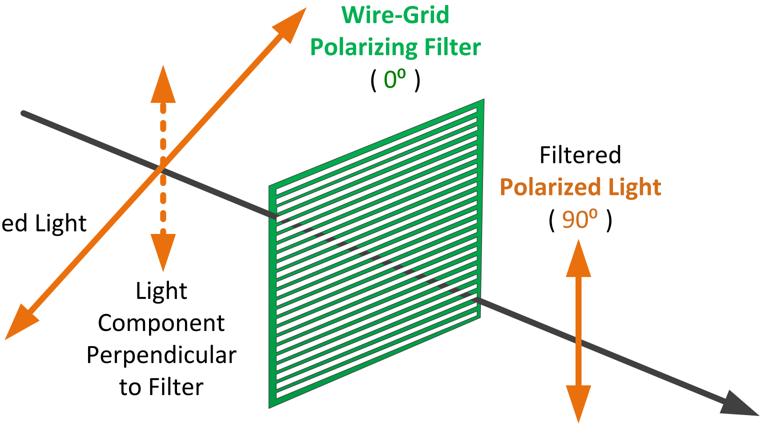
If the required version is not available, contact your Teledyne DALSA representative.

Sapera LT includes the CamExpert application, which provides a graphical user interface to access camera features for configuration and setup.

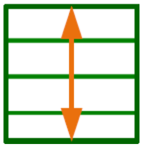
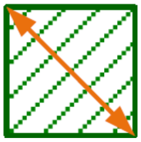
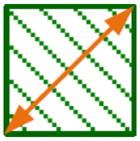
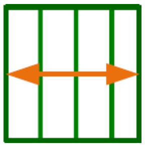
Polarization Filter Theory of Operation

Polarization direction is defined as the electrical field direction. Light, with its electrical field oscillating perpendicular to the nanowire grid, passes through the filter while that in the parallel direction is rejected (reflected/absorbed by the nanowire filter).

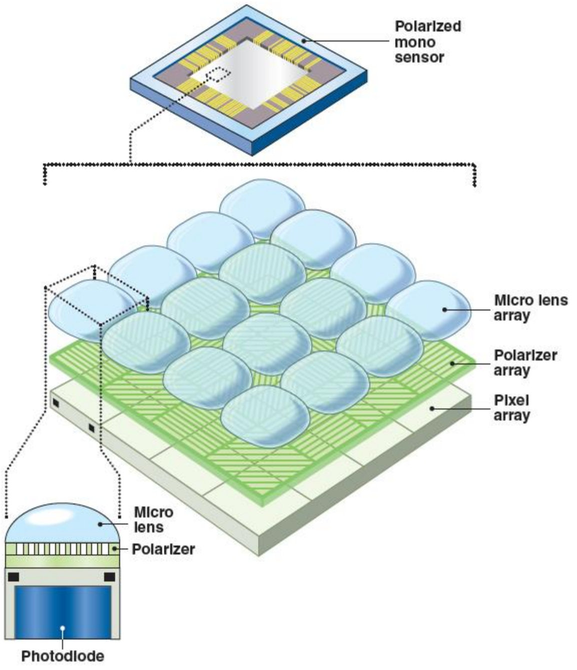
For polarized light, only the portion of the light vector perpendicular to the angle of the nanowire filter grid passes.

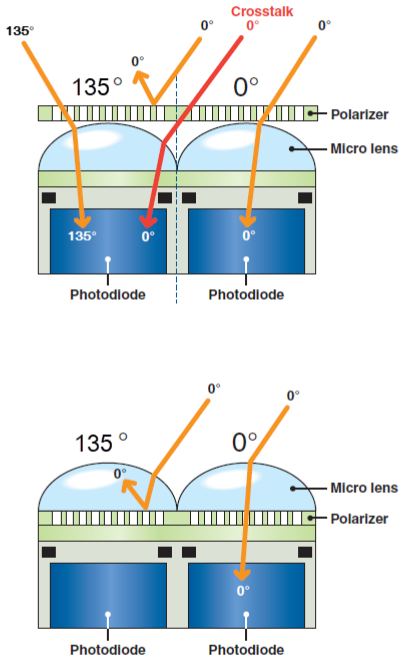


For example, with a wire-grid **polarizer filter** (shown in **green**) at 90° the maximum transmission is for **polarized light** (shown in **orange**) at an angle of 0 °.

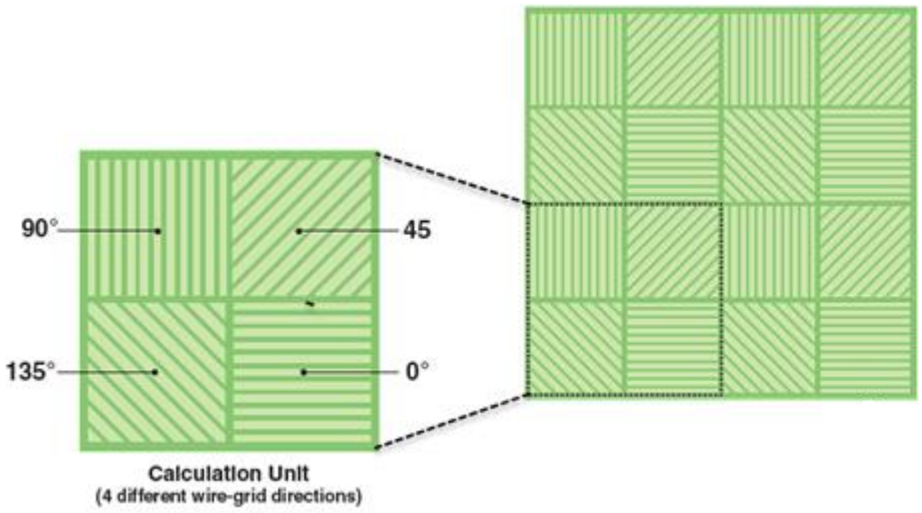
Nanowire Polarizer Filter	Filter Angle (blocks light at this angle)	Polarized Light Angle (passes through filter)
	**0°**	**90°**
	**45°**	**135°**
	**135°**	**45°**
	**90°**	**0°**

The polarizer filter is placed directly on the sensor’s pixel array, beneath the micro lens array. This design, compared to polarizer filters on top of the micro lens array, reduces the possibility of light at a polarized angle being misdirected into an adjacent pixel (crosstalk) and incorrectly detected at the wrong angle.

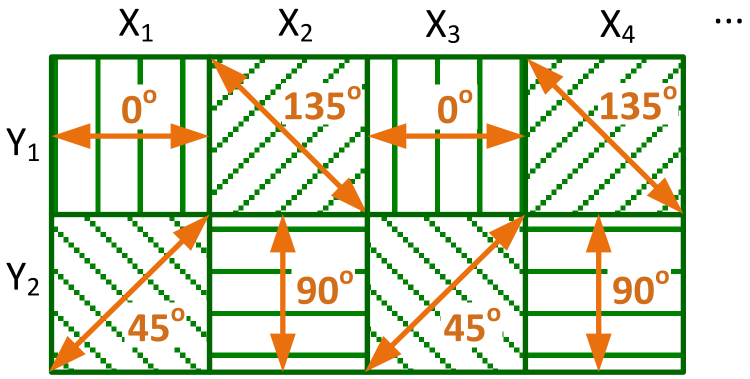




The Genie Nano’s polarizer filter on the camera sensor is a 2x2 pattern, with each pixel having a nanowire polarizer filter with a different angle (90º, 45º, 135º, 0º).



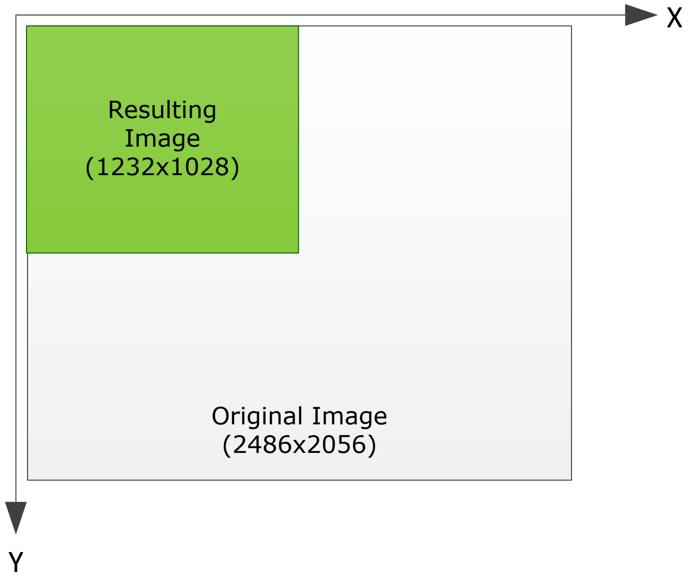
The angles of the **polarized light** that passes through the **polarizer filter** of the camera is arranged in 2x2 pixel blocks as follows:



That is, the first line output is an alternating sequence of pixels with **polarized light angles** of **0°** , **135°**; the following line **45°** ,**90°**.

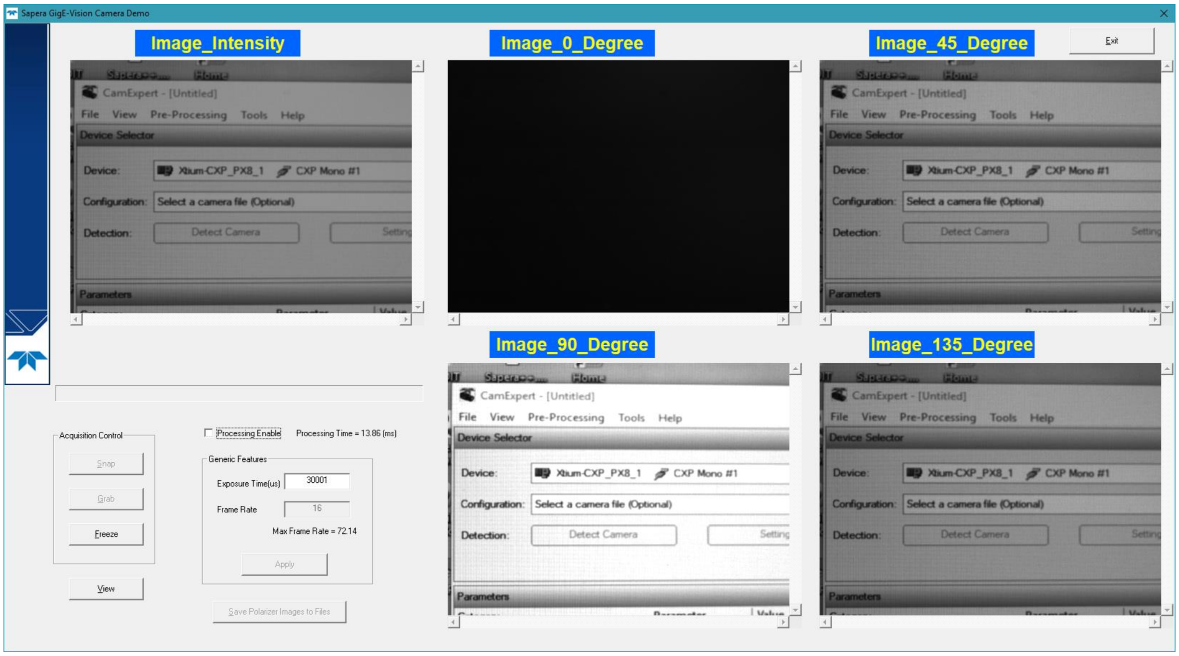
Given the proportion of light available through these four filters, any angle of polarized light can be calculated. Any given state of polarization can be composed by two linearly polarized waves in perpendicular directions. The state of polarization is determined by the relative ampltitude and difference in phase between the two component waves.

Calculations on the 2x2 filter blocks result in a single pixel for each polarizer filter angle, therefore the resulting image is one fourth the original image resolution. For example, with an original image of 2464x2056, the resulting image is 1232x1028 (original buffer width/2 and orignal buffer height/2) for a single polarizing angle.



Polarization Demo User Interface

The demo user interface can display either raw images for the image intensity sum and each of the 4 polarizer filter angles, or when the Processing Enable checkbox is active, the result of the 4 processing algorithms.



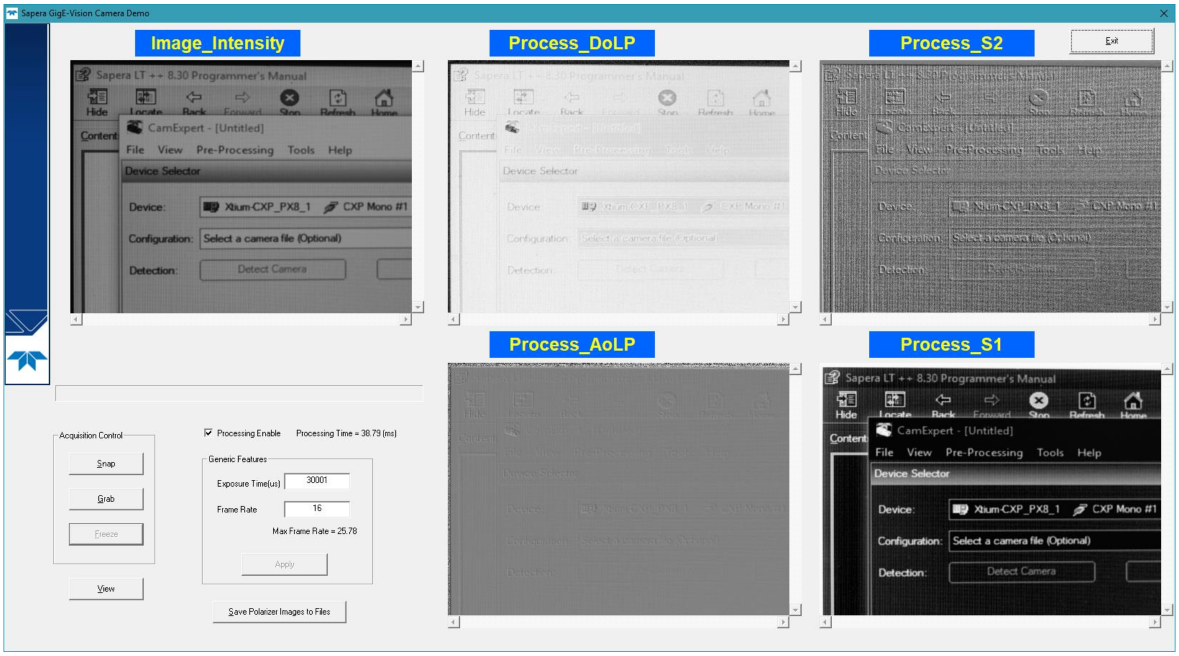
☐ Processing Enable Processing Time = 13.86 (ms)

Exposure Time(us) 30001

Frame Rate 16

Max Frame Rate = 72.14

Raw Polorization Images



☑ Processing Enable Processing Time = 38.79 (ms)

Exposure Time(us) 30001

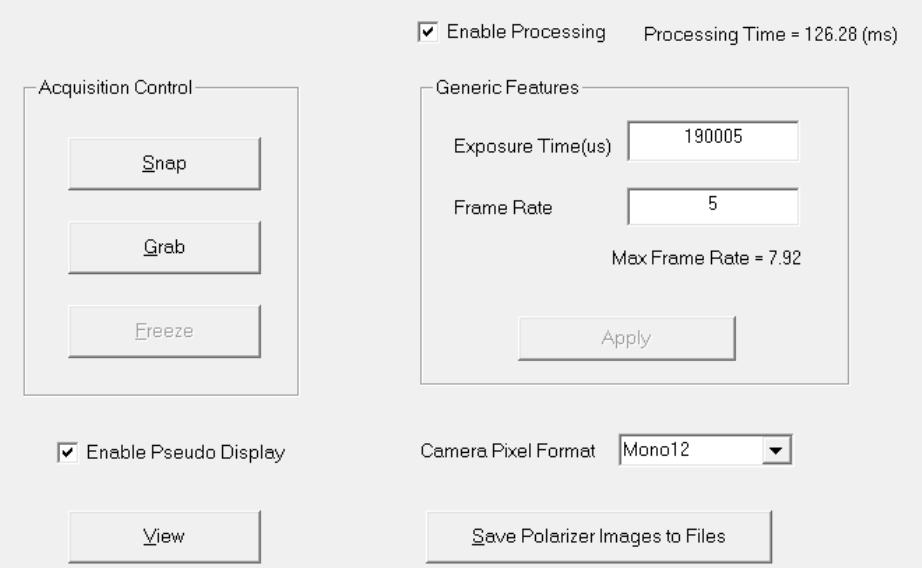
Frame Rate 16

Max Frame Rate = 25.78

Processed Polarization Images

For display purposes, the demo applies a scale and offset to the processed images. The range of pixel values (minimum to maximum) determines the scaling factor to apply such that all the 8-bits (0-255) of the display are used.

The bottom-left area of the window groups demo parameters, such as acquisition control, camera pixel format and displays acquisition related statistics including the current exposure time, processing time, frame rate and the maximum frame rate. In addition, a pseudo-color display can also be enabled.



☑ Enable Processing

Processing Time = 126.28 (ms)

Exposure Time(us): 190005

Frame Rate: 5

Max Frame Rate = 7.92

☑ Enable Pseudo Display

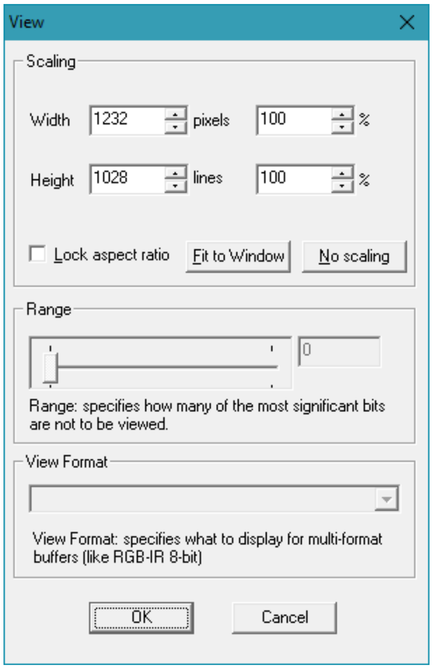
Camera Pixel Format: Mono12

View Settings

To open the View dialog, click the View button in the demo application window.

The View dialog groups various parameters related to the image display in the demo window. For example, to fit the acquisition image to the display area, click the Fit to Window button.

If the acquisition pixel depth is greater than 8-bits, the Range slider can be used to select the bit range to display.



Width: 1232 pixels, 100 %

Height: 1028 lines, 100 %

□ Lock aspect ratio

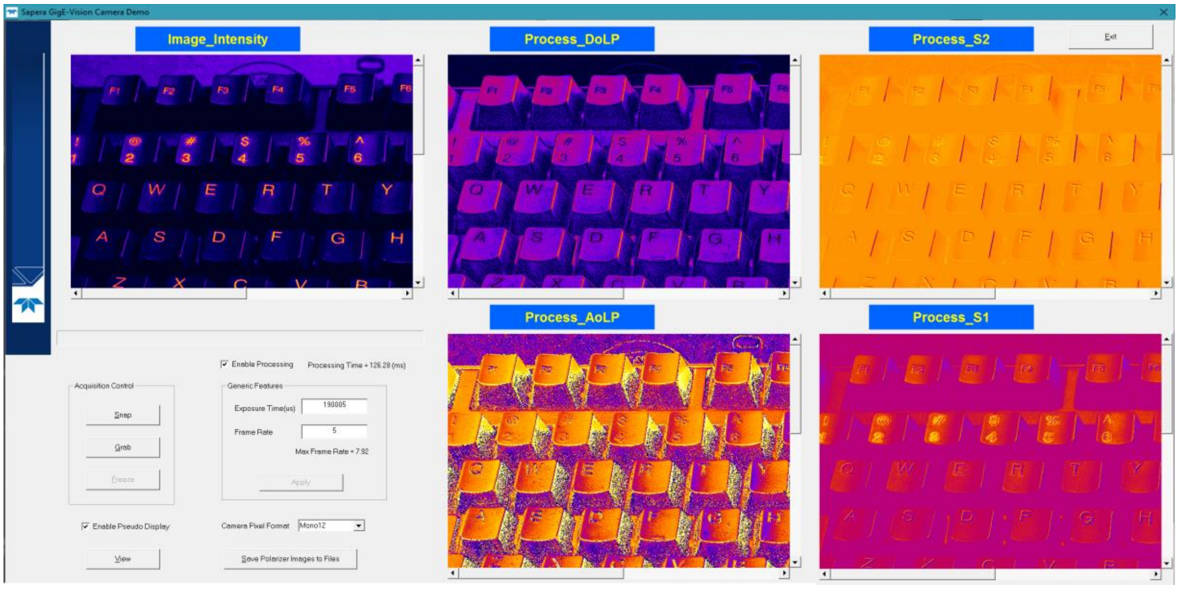
Range: 0

Range: specifies how many of the most significant bits are not to be viewed.

View Format: specifies what to display for multi-format buffers (like RGB-IR 8-bit)

Pseudo Color Display

When enabled, images can be displayed using a pseudo-color mapping. The demo uses a text file containing a lookup table (LUT) of RGB values corresponding to 8-bit grayscale values.



☑ Enable Processing Processing Time = 126.28 (ms)

Exposure Time(us) 190005

Frame Rate 5

Max Frame Rate = 7.92

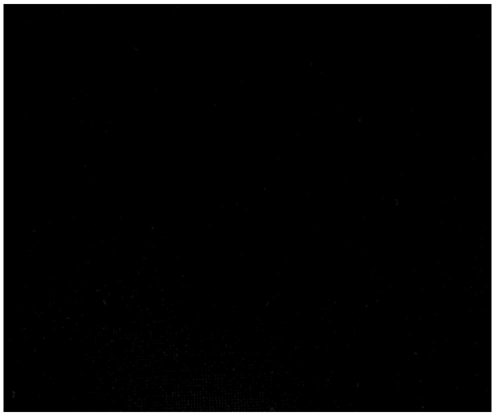
☑ Enable Pseudo Display

Camera Pixel Format Mono12

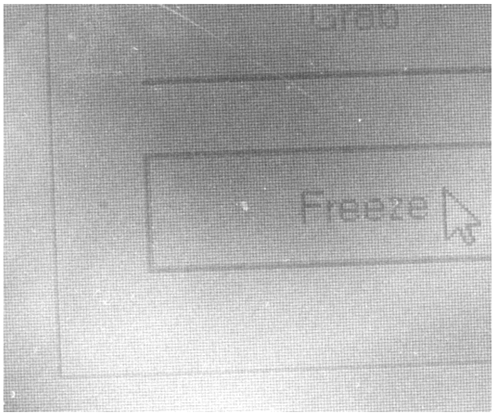
Raw Polarized Images

Raw, unprocessed polarized images can be extracted from the sensor for each of the **polarizer filter** angles (90º, 45º, 135º, 0º) and analyzed.

For example, the raw image of a computer LCD monitor, at the polarizer filter angle 45º, displays a seemingly black image; since this particular LCD is designed with a polarizer filter at this angle, it is effectively blocked. However, by adjusting the contrast and brightness settings (for display purposes only), image details can be revealed.



Raw Polarized Image at 45º



Contrast/Brightness Enhanced Image

Demo Processing of Polarized Image Data

Different algorithms can be applied to the polarized images for image processing. The Sapera LT Polarized Sensor Demo implements the following processing algorithms, where I is the pixel intensity and the numeral represents the angle:

- **Stokes S1**: `S1 = 0 - I90`
- **Stokes S2**: `S2 = I45 - I135`
- **Angle of Polarization (AoP)**: $AoP = \frac{1}{2}arctan\left(\frac{S_2}{S_1}\right)$
- **Degree of Linear Polarization (DoLP)**: $DoLP = \frac{\sqrt{{S_1}^2+{S_2}^2}}{S_0}$

Stokes Parameters

Stokes parameters are linear combinations of power measured in orthogonal polarizations.

Fully unpolarized light at perpendicular angles of 90º cancels itself since both parts are of equal intensity (Intensity 0º - Intensity 90º = 0). However, any polarized light present results in a non-zero value, since a difference in intensity will exist between the amount of polarized light at the two angles. The S1 and S2 stokes formulas represent these differential signals and enhance contrast.

Degree of Linear Polarization (DoLP)

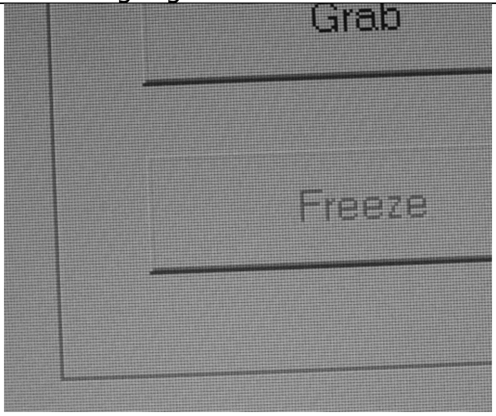
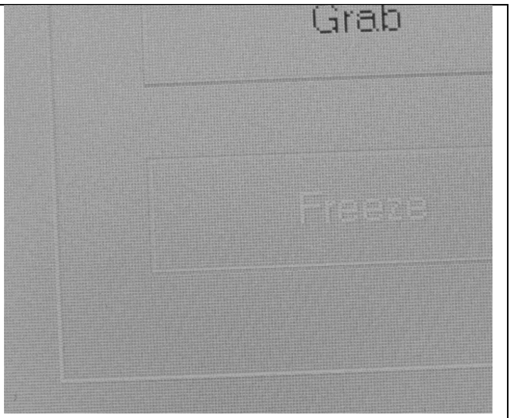
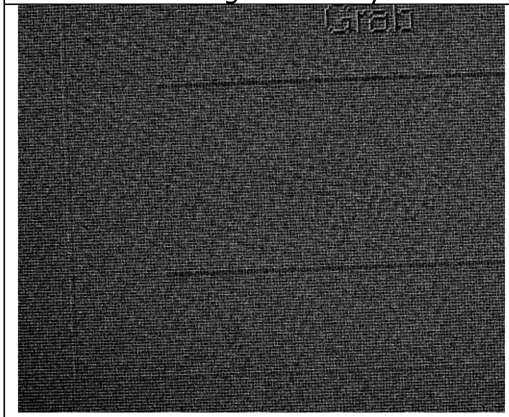
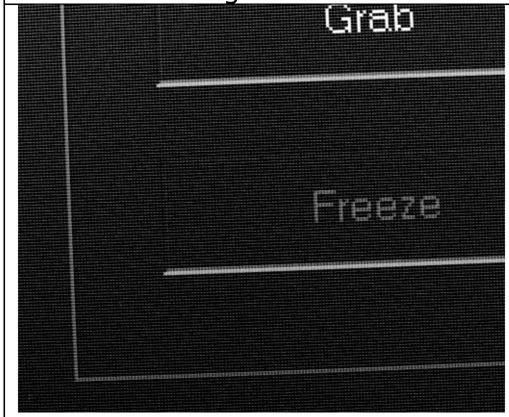
The degree of linear polarization is defined as the ratio of the intensity of the polarized part to the total intensity. That is, it is a measure of the what percentage of the total average lightwave signal is polarized, and can be related directly to mateial properties.

Angle of Polarization (AoP)

Angle of Polarization is the angle of the major axis of the polarization ellipse. This can be related directly to material properties.

Sample Images

The following images of a computer LCD monitor show the effects of the various processing algorithms.

 Raw Image Intensity Summation	 Degree of Linear Polarization
 Angle of Polarization	 Stokes S1
 Stokes S2	

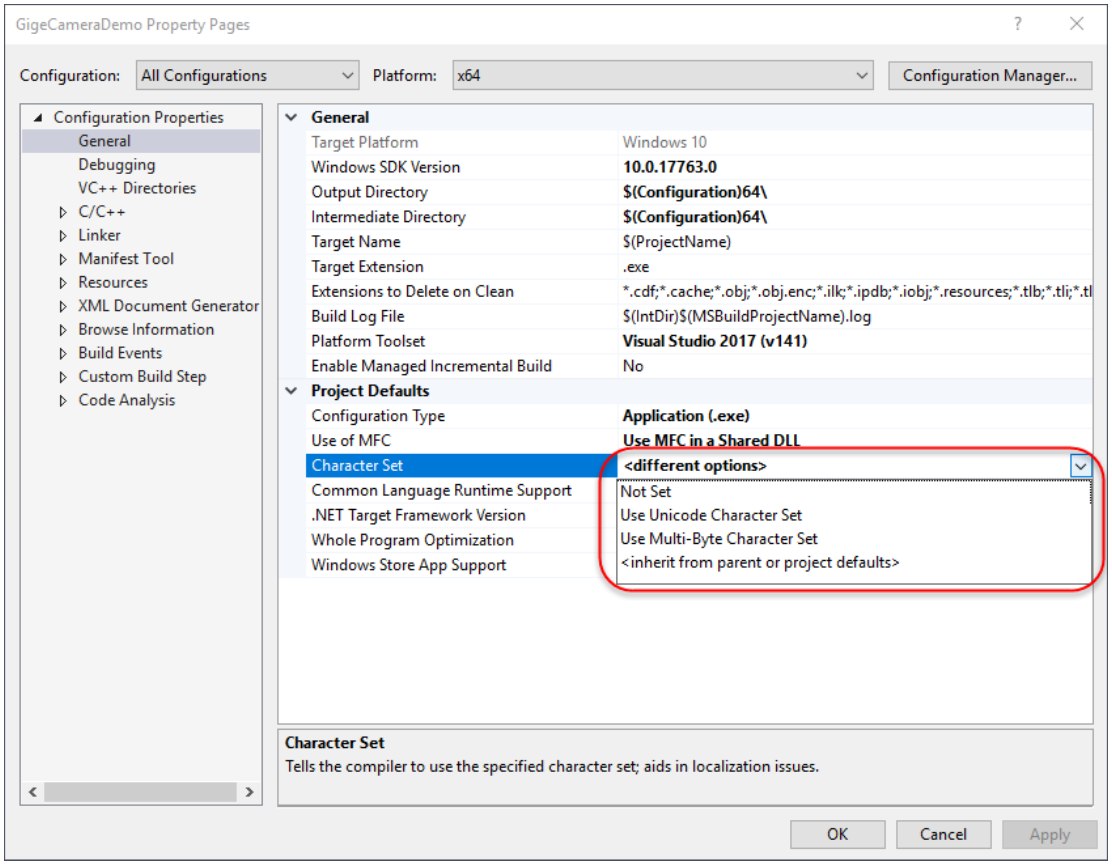
Recompiling the Demo Source Code

The demo program uses the Sapera LT GUI classes to create the graphical user interface.

The Sapera LT installation provides Sapera GUI classes Visual Studio projects and source code, but does not provide Sapera GUI class DLLs. Before recompiling the demo project you must compile the Sapera GUI class project to generate the Sapera GUI DLL.

Note: When recompiling the Sapera GUI class project, the project setting "Character Set" must be set same as that being used in the GigeCameraDemo project.

Build types include debug and release versions using ANSI (Multi-Byte) or Unicode character sets; when building ensure that both the GigECameraDemo and SapClassGui projects have the same Character Set property setting.

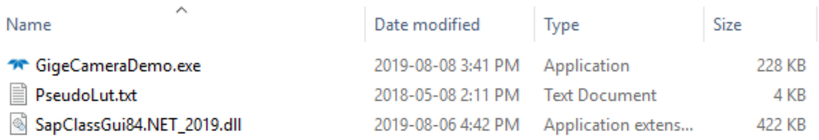


Configuration: All Configurations   Platform: x64

General	
Target Platform	Windows 10
Windows SDK Version	**10.0.17763.0**
Output Directory	**$(Configuration)64**
Intermediate Directory	**$(Configuration)64**
Target Name	$(ProjectName)
Target Extension	.exe
Extensions to Delete on Clean	*.cdf;*.cache;*.obj;*.obj.enc;*.ilk;*.ipdb;*.iobj;*.resources;*.tlb;*.tli;*.tl
Build Log File	$(IntDir)$(MSBuildProjectName).log
Platform Toolset	**Visual Studio 2017 (v141)**
Enable Managed Incremental Build	No
Project Defaults	
Configuration Type	**Application (.exe)**
Use of MFC	**Use MFC in a Shared DLL**
Character Set	**<different options>**
Common Language Runtime Support	
.NET Target Framework Version	
Whole Program Optimization	
Windows Store App Support	

Character Set
Tells the compiler to use the specified character set; aids in localization issues.

When compiling, it is recommended to run Microsoft Visual Studio with administration rights; this copies the new dll (for example, *SapClassGui84.NET_2019.dll*) to the Windows/System32 folder, otherwise ensure that the new .dll is available in the executable file directory:



Name	Date modified	Type	Size
GigeCameraDemo.exe	2019-08-08 3:41 PM	Application	228 KB
PseudoLut.txt	2018-05-08 2:11 PM	Text Document	4 KB
SapClassGui84.NET_2019.dll	2019-08-06 4:42 PM	Application extens...	422 KB