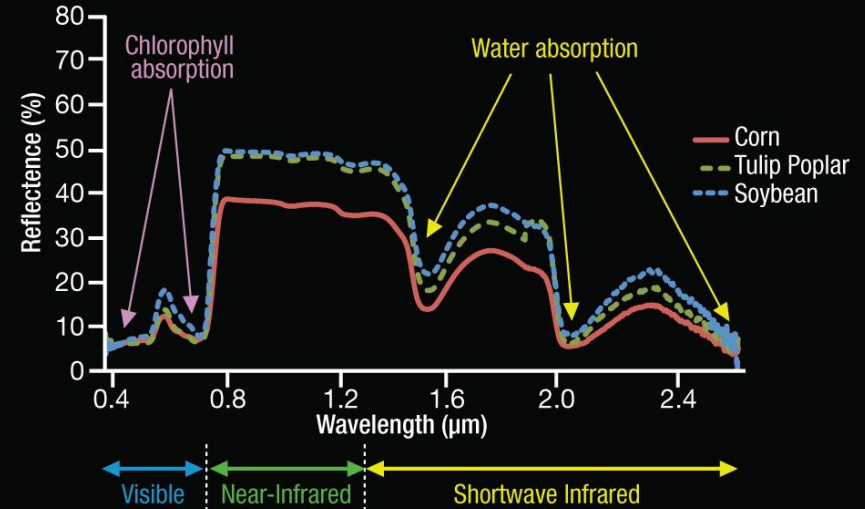
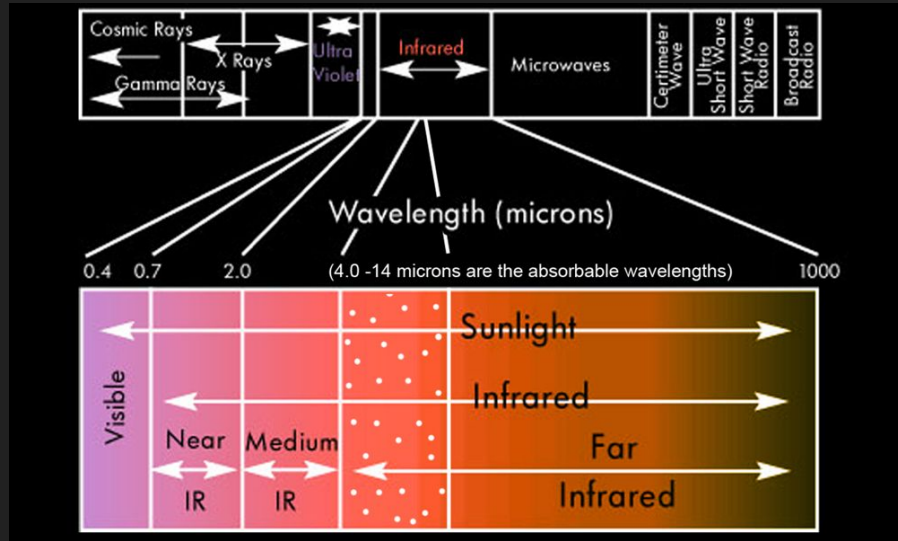


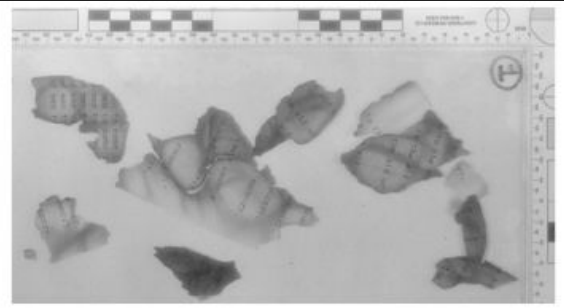
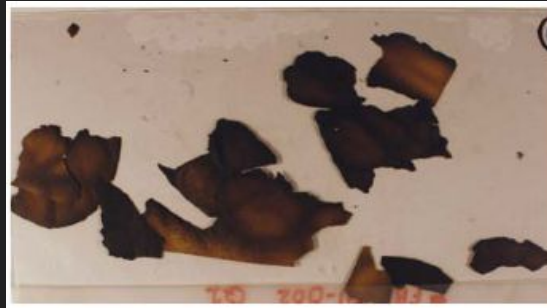
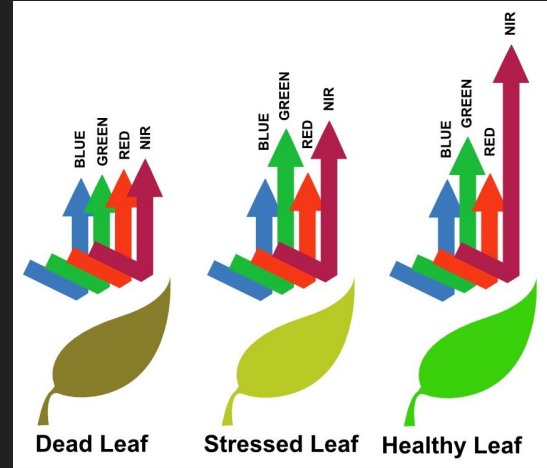
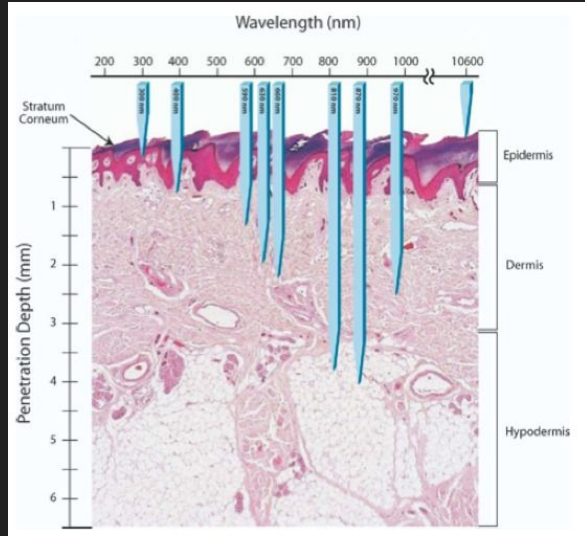
Infrared Camera and Image Processing

Keegan Kroll

Near-Infrared Photography



Near-Infrared Photography Applications



Near-Infrared and Aerochrome

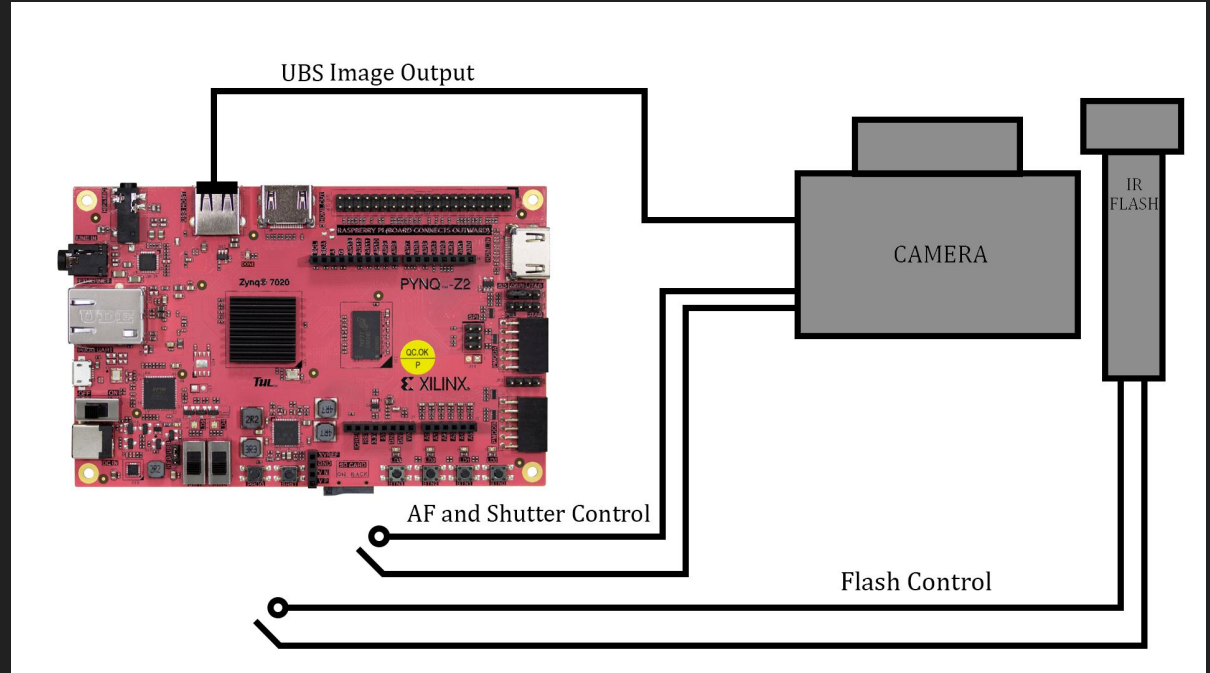


Project Design

Code Start:

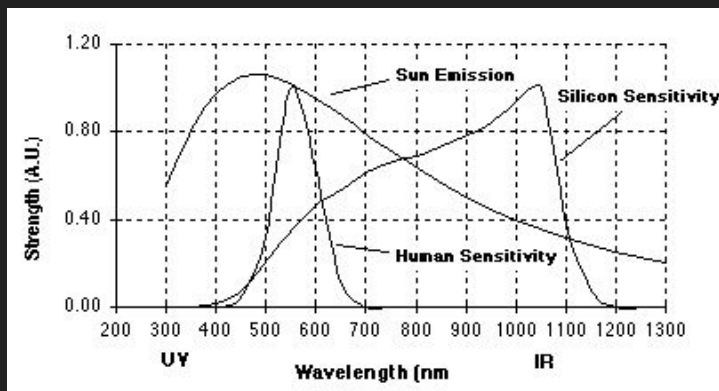
1. Camera
 - a. Auto Focus
 - b. Flash
 - c. Shutter
2. Board
 - a. Load Photo
 - b. Process Multiple Ways
 - c. Export and Display

Code End

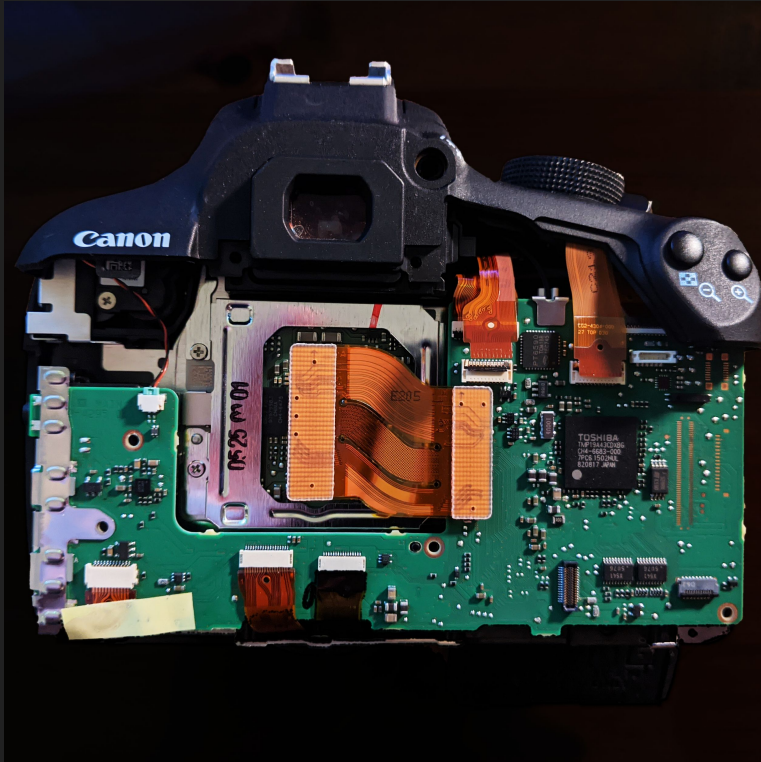


Project Specifications

Camera Type	Modified Canon EOS 1200D
Camera Output	RAW (.CR2), 5,184 × 3,456 Resolution
IR Filter Wavelength	760 nm to 1,200 nm
PYNQ Output	.PNG, 5,184 × 3,456
Flash Wavelength	850 nm



Camera Full-Spectrum Conversion





Auto White Balance

Most important processing step for IR photos

```
## AUTO WB ##
def auto_WB(RGB):
    wbalanced = np.zeros_like(RGB)
    colors = ("r", "g", "b")

    for i in range(3):
        hist, bins = np.histogram(RGB[..., i].ravel(), 256, (0, 256))

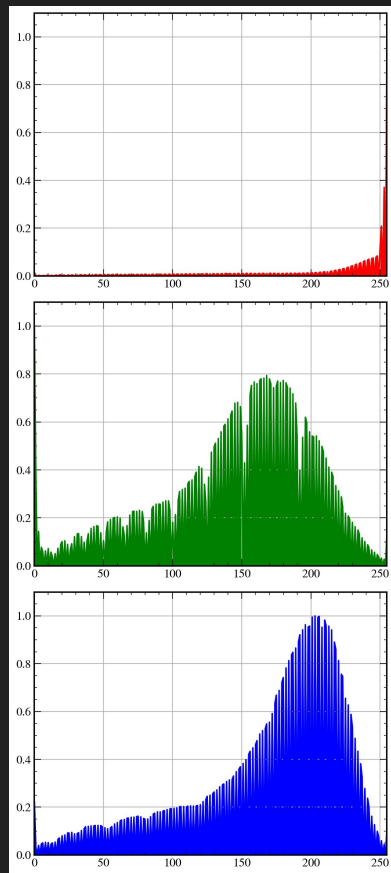
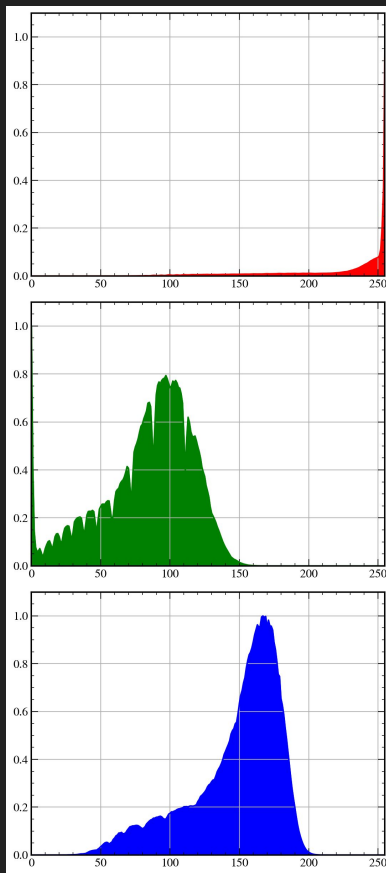
        bmin = np.min(np.where(hist>(hist.sum()*0.0005)))
        bmax = np.max(np.where(hist>(hist.sum()*0.0005)))

        wbalanced[..., i] = np.clip(RGB[..., i], bmin, bmax)
        wbalanced[..., i] = (wbalanced[..., i]-bmin) / (bmax - bmin) * 255

        hist, bins = np.histogram(wbalanced[..., i].ravel(), 256, (0, 256))

    return wbalanced
```

Discard 0.05% of histogram on each side and
“stretch” remaining colors



Auto White Balance

Original



White Balanced



Further Processing

Red-Blue Channel Swap



High-Contrast Black and White



Further Processing

Pseudo-Thermal



Night Vision (Inverted)



Schedule and Future

	December 3	4	5	6	7	8	9	10
RAW File Support								
Camera/Flash Control								
Take More Photos								
Aerochrome Processing								
Temperature Calibration								

Core Coding: Essentially Done

Aerochrome Processing: To-Do

Camera/Flash Control: In Progress

Temperature Calibration: Time Permitting

Further Application Photos: To-Do

Questions?

