TIPS AND TRICKS FOR GUI DESIGN

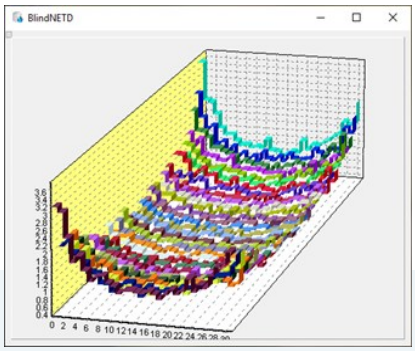
HOW TO OBTAIN THE BEST VISUAL PERFORMANCE FOR YOUR IR IMAGE

1. False Color Scaling Limits

The first parameter to evaluate is the noise performance of the IR sensor. Due to vignetting and the $\cos^4$-law, the signal level decreases toward the outer pixels compared to the central region. Since the electrical noise of each pixel remains essentially constant, pixels farther from the center exhibit a reduced signal-to-noise ratio **(SNR)**.

This reduction in SNR results in a higher Noise Equivalent Temperature Difference **(NETD)** for the corner pixels of the Focal Plane Array **(FPA)**. In the case of wide-angle optics, corner pixels may receive little or no signal, effectively rendering them blind. Such pixels can exhibit noise levels that significantly exceed the actual scene content.

Figure on the right shows an example NETD map of an HTPA32x32d sensor with wide Field of View **(FOV)** optics, illustrating this effect.



The noise level of the corner pixels is approximately four times higher than that of the central region of the FPA. If the minimum and maximum temperature values of each frame are directly mapped to black and white (for grayscale display), the image will exhibit pronounced flicker noise. This effect is caused by the extreme corner values continuously shifting the scaling range from frame to frame, leading to significant offset variations in the grayscale levels of the central pixels, even when the scene itself remains temporally stable.

1.1 Solutions for Noise Depression

- One approach to improve stability is to calculate the average of, for example, the three lowest and three highest pixel values in a frame and use these averages as the grayscale limits. While this method results in a small number of pixels falling outside the scaling range, it significantly increases the temporal stability of the boundaries.
- Alternatively, the minimum and maximum frame values can be smoothed over time using a moving average. This also reduces boundary noise.
- A combination of both methods can be applied for further improvement. These options are available through the smoothing function in the **Heimann Sensor GUI v2**.
- Additionally, introducing a small hysteresis in the scaling algorithm helps to avoid frequent dynamic changes in frame content. If scale.max and scale.min represent the current scaling limits and frame.max and frame.min represent the frame values to be displayed, the implementation can be expressed as follows:

The parameter **AUTOSCALEHYST** must be configured according to the specific system requirements. In most cases, the **Heimann GUI v2** applies a default value of 0.8.

The method used to define the boundaries of the false-color scale has a significant impact on image quality. In high-contrast scenes, it is often not optimal to set the grayscale limits strictly to the minimum and maximum pixel values within a frame. For example, if the image contains both humans and very hot objects, but the application is focused on human detection, the upper boundary should be set to a value slightly above typical human body temperature.

```
bool ReScale=false;
if((scale.max>(frame.max+AUTOSCALEHYST))||(scale.max<(frame.max-AUTOSCALEHYST)))
    ReScale=true;
if((scale.min>(frame.min+AUTOSCALEHYST))||(scale.min<(frame.min-AUTOSCALEHYST)))
    ReScale=true;
```

1.2 Solutions for min/max Determination

These depend strongly on your application. In most scenarios, a simple detection as described previous (combined with the "smoothing") is the best approach. Please check the manual scaling methods in the **GUI v2** to get an overview on possible solutions. For example, for the above described method a dynamic minimum scale, but a fixed maximum at 37 °C will help tremendously to shift the dynamic range of the image towards the humans.

HEIMANN Sensor GmbH
Maria-Reiche-Str. 1
D-01109 Dresden Germany

Contact / Customer Support
Phone 49 (0) 6123 60 50 30
sales@heimannsensor.com

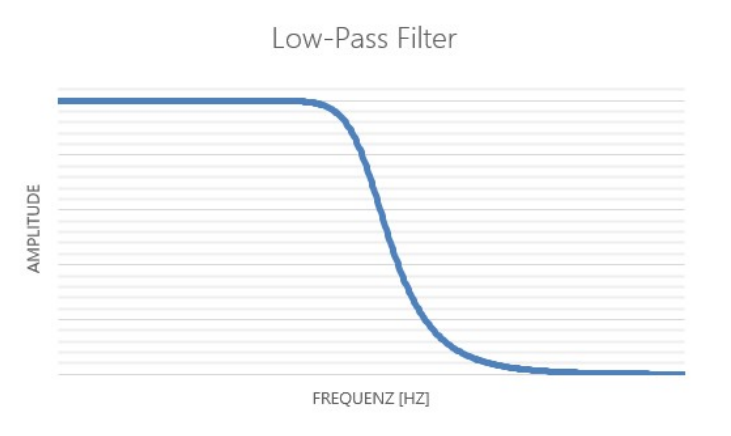
Internet
www.heimannsensor.com
mail: info@heimannsensor.com

TIPS AND TRICKS FOR GUI DESIGN

2 Graphic Improvements

2.1 Noise Depression

The noise in the image can be decreased by the following methods:



Low Pass Filters

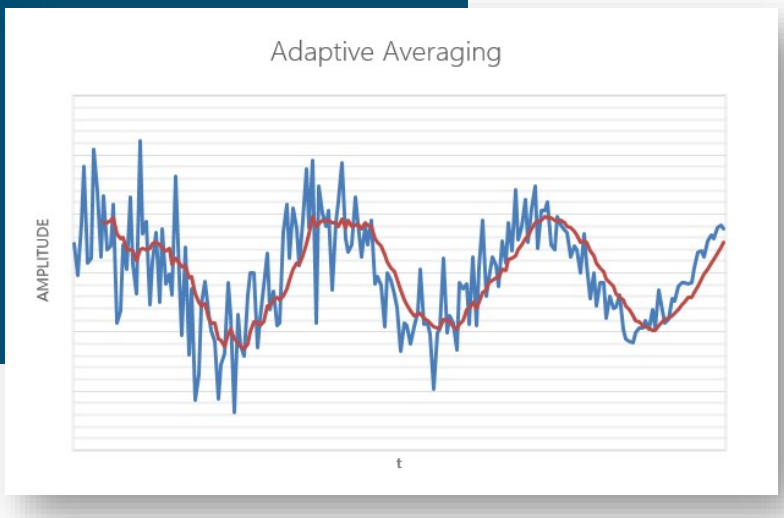
In the GUI there are two low pass filters implemented. One **IIR** (Infinite Impulse Response) and a **FIR** (Finite Impulse Response) low pass filters. These can cut of high frequent noise. Both are designed to a cut-off frequency of 3Hz, when input is driven with 10Hz. They utilize a second order Bessel window. Code for these can be found in the subfolder **"CodeExamples"** of the **GUIv2** folder.

Pro/Con: Very good high frequency noise depression without the huge influence on the bandwidth (approx. -20dB at 4Hz, but around -80dB for 5Hz). Both filters create a significant group delay, approx. 130ms for the IIR and 1.5s (!) for the FIR. Needs huge amounts of memory (TAP_NUM+1) *pixelcount of the device.

Adaptive Averaging

This is basically also a moving averaging of each pixel over time, but with the precondition, that the actual difference of the reading to the pre-calculated average must be smaller than a given threshold to apply the new average. If the difference is larger, than the actual reading is displayed. Otherwise, the moving average is displayed. Check the **"AdapAv"** option in the filter settings of the **Heimann GUI v2**. The threshold can be set in this case with the slider, but for a single system, it can be set i.e. to five times the NETD of the system. The average is calculated in the GUI for four frames, but of course different values can be used.

Pro/Con: Good noise depression, but can generate "ghosts" in the image. Dynamic influence on the bandwidth of each pixel, which cannot be predicted.



HEIMANN Sensor GmbH
Maria-Reiche-Str. 1
D-01109 Dresden Germany

Contact / Customer Support
Phone 49 (0) 6123 60 50 30
sales@heimannsensor.com

Internet
www.heimannsensor.com
mail: info@heimannsensor.com

TIPS AND TRICKS FOR GUI DESIGN

Median Filter

Can be used to get rid of "salt and pepper noise", while it does not matter, if this is static (offset differences of the pixel) or dynamic noise. This is a simple sorting of the eight adjacent neighbors of each pixel and taking the median of these readings for the central pixel reading. If there are large contrasts in the respective pixels, the contrast may be reduced by taking simply the median. Therefore the **Heimann GUI v2** utilizes a threshold, which determines below, which contrast ratios of the respective pixels the median should be applied. A careful design of this threshold will show good results.

Pro/Con: No influence on the bandwidth, it needs small processing power and almost no memory. May decrease contrast (blurry image), if applied not carefully and without threshold.

Default vs Median

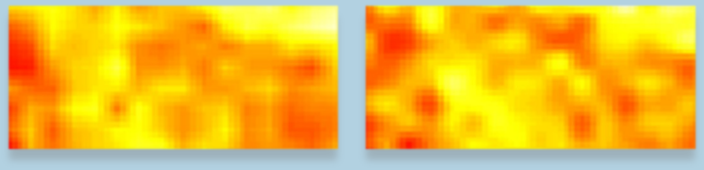
Moving average of the readings of each pixels over time

Pro/Con: Decreases the noise, but has a tremendous effect on the bandwidth slow. Recommended for static or slow changing scenarios.

2.2 Interpolation

By using interpolation methods the resolution of the image can be enhanced. The easiest way to do this is the bilinear algorithm. Since this relatively trivial please refer to Wikipedia for this. Another way is the bicubic interpolation, which shows less artifacts but needs much more processing time. A bicubic interpolation algorithm can be found as plain c-code in the **"CodeExamples"** folder of the **GUI v2** subfolder.

Bilinear algorithm **(left)** compared to the bicubic **(right)**:



HEIMANN Sensor GmbH
Maria-Reiche-Str. 1
D-01109 Dresden Germany

Contact / Customer Support
Phone 49 (0) 6123 60 50 30
sales@heimannsensor.com

Internet
www.heimannsensor.com
mail: info@heimannsensor.com