Adaptive Sensitivity

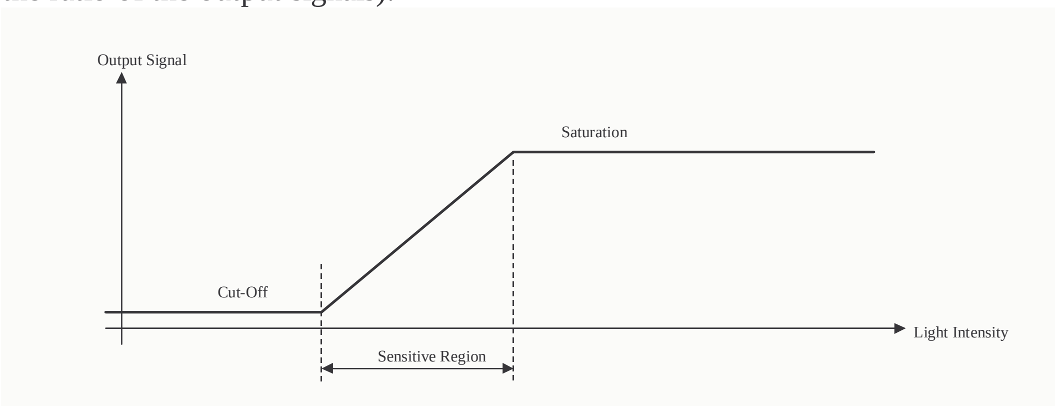
Ran Ginosar

Introduction

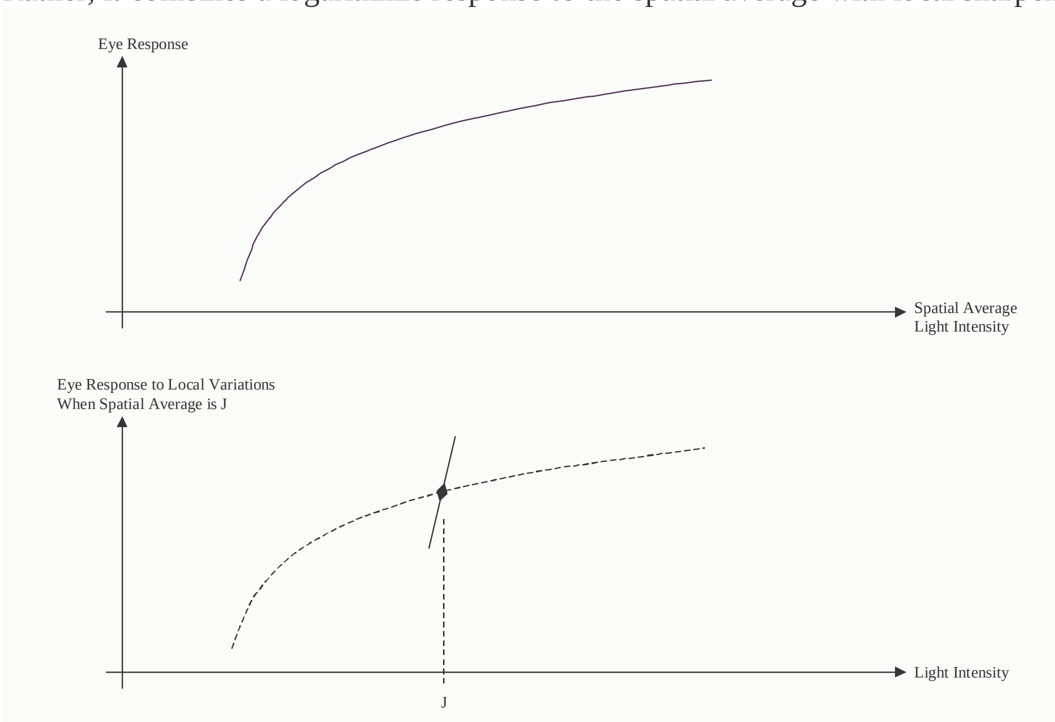
Solid state (and other) sensors have a limited *dynamic range* D_{in}:

$$D_{in} \equiv \frac{\text{saturation threshold}}{\text{cut-off threshold}}$$

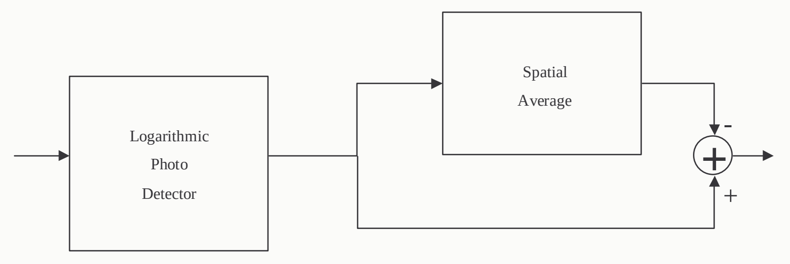
where both thresholds are measured in terms of the light intensity (D_{out} is specified as the ratio of the output signals).



Typically, $100:1 < D_{in} < 5{,}000:1$ (40 - 74 dB, or 6 - 12 bits). In contrast, the human eye has a much wider dynamic range, in excess of $10^8:1$, but the response is non linear. Rather, it combines a logarithmic response to the spatial average with local sharpening:

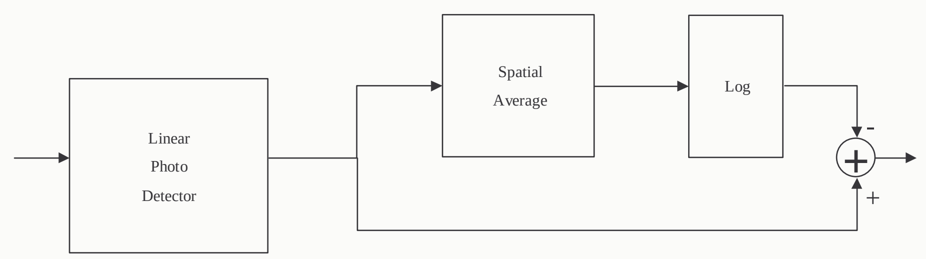


The Silicon Retina (C. Mead, Analog VLSI and Neural Networks, Ch. 15) approximate the human eye model:



Deficiencies of the Silicon Retina model:

- The logarithm is applied before spatial averaging, resulting in loss of details. In contrast, an approximate eye model is:



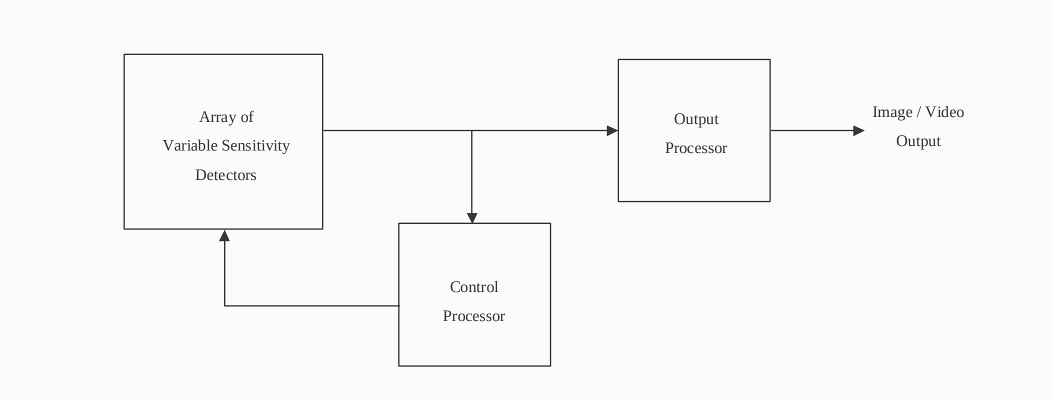
- A complicated circuit at the focal plane, resulting in lower fill factor and resolution.

The Adaptive Sensitivity Sensor

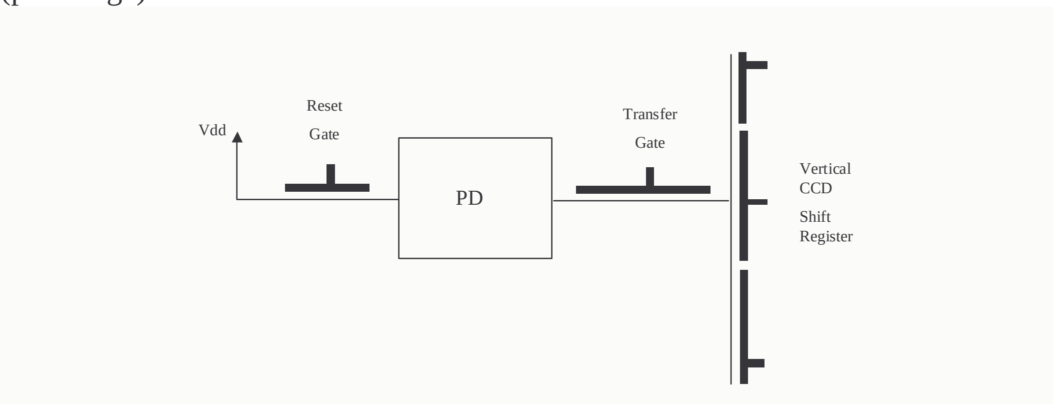
Motivation: Minimal circuitry at the focal plane, leading to:

- Sensitivity control outside the sensor array (either on or off the same chip)
- Exploit the fact that VLSI circuits are substantially faster than biological “circuits”. The Silicon Retina is far too fast. The bandwidth gap can be used for communications.

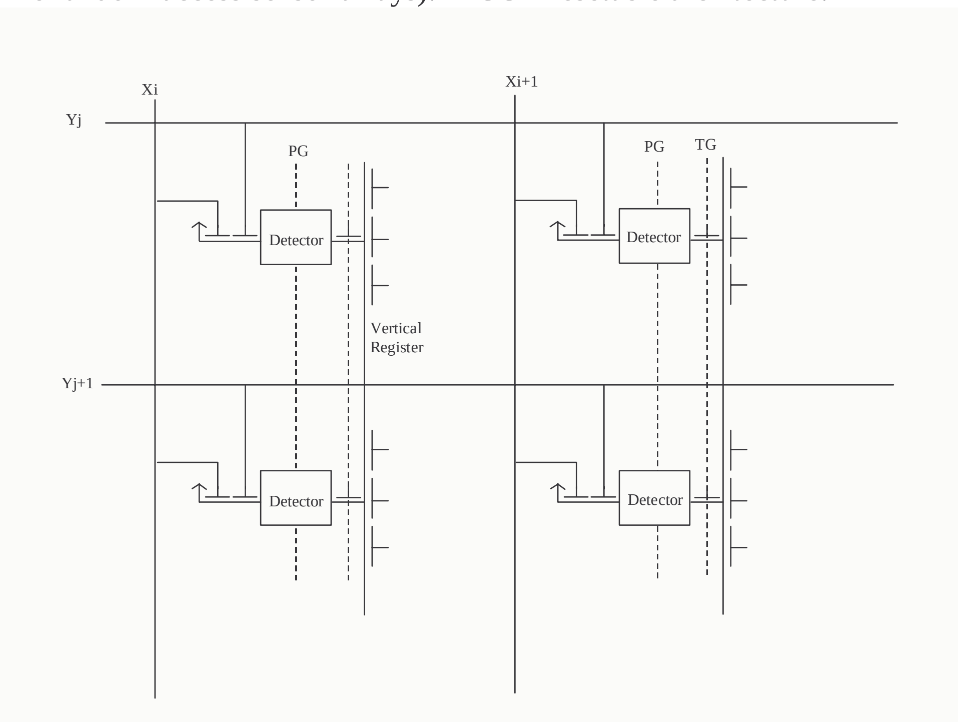
Architecture:



The simplest means of per-pixel sensitivity control is varying the exposure time.
The simplest method for varying exposure time is by resetting (pre-charging) the detectors. A resetable CCD pixel has a reset (precharge) gate(s) and a reset (precharge) drain:



Reset (precharge) control is achieved through a X-Y access array (like memory, and like random-access sensor arrays). A CCD resetable architecture:

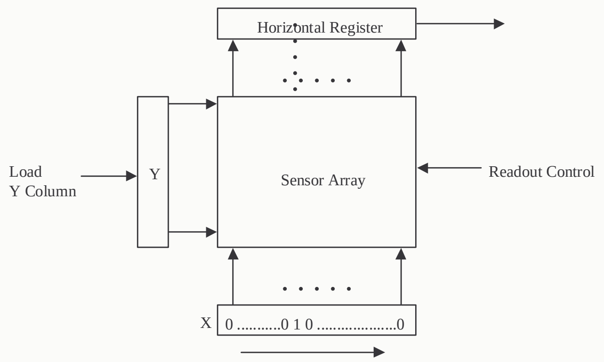


Operation of the A/S Sensor

Modes of operation:

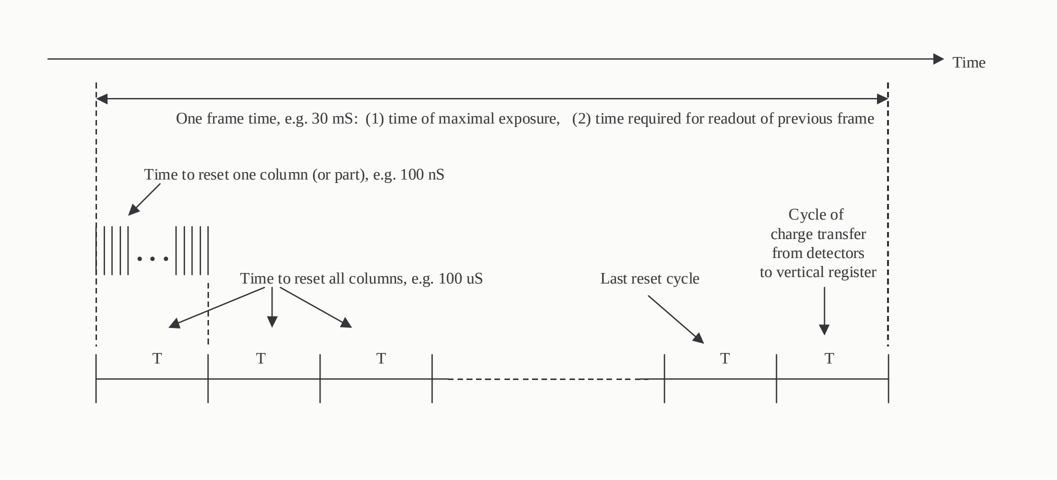
1. One row, one column ON $\rightarrow$ one pixel is reset (precharged)
2. Several rows, one column ON $\rightarrow$ several pixels along a single column are reset simultaneously
3. Several rows, several columns ON $\rightarrow$ a complete block can be reset at once.

Architecture allowing modes 1, 2:

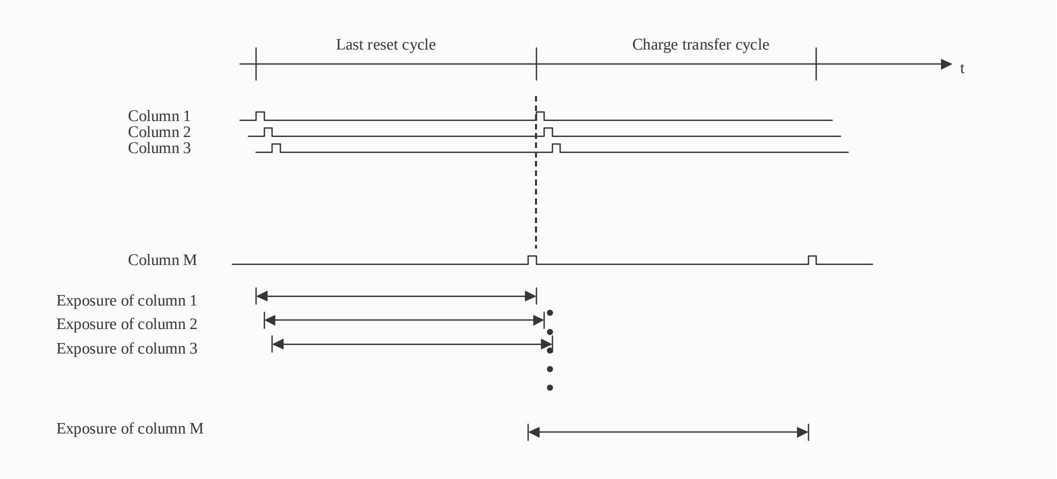


A single ‘1’ bit rotates at the X register, enabling a single X_i column each time. For each column, a set of Y values are loaded into the Y register. Each ‘1’ bit in the Y register resets the corresponding pixel on column X_i. To allow mode 3 operation, multiple bits are set ON in register X.

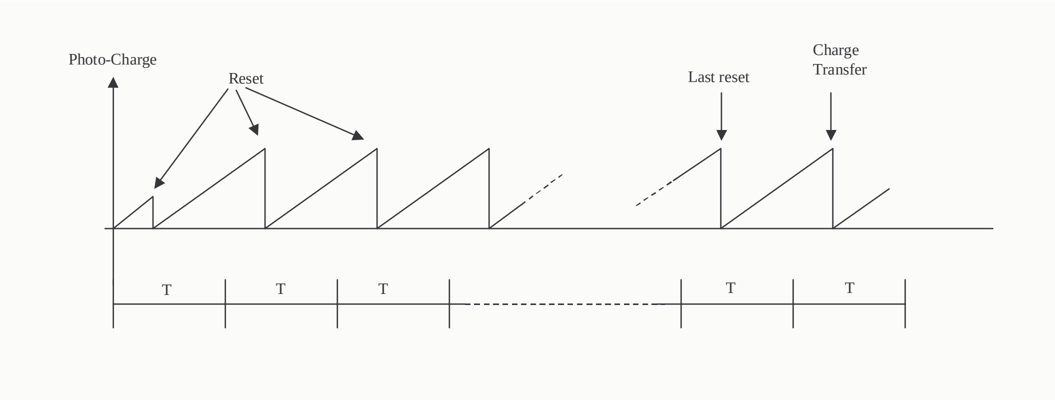
Consider mode 2 schedule:



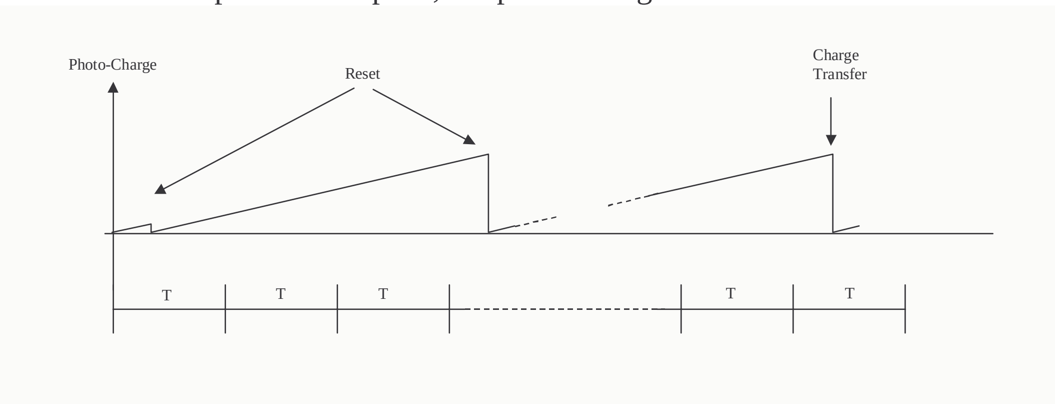
For each individual pixel, the exposure time is kT ($k \in N$). In the example above, $1 \le k < 300$. When the exposure time of a certain pixel is T, that pixel is reset during the last reset cycle, and is read out during the next cycle (the last cycle of the frame). When the exposure time is maximal (299T in the example), the pixel is reset once only during the first cycle. When implemented in CCD, charge transfer to the vertical register is skewed from one column to the next, to assure same exposure times for all columns. Assume all pixels are minimally exposed for T, namely all pixels are reset during the last reset cycle, the process can be visualized as follows:



Note that we would select the shortest exposure time for a pixel if it is exposed for a very strong light. That pixel is reset not only during the last reset cycle, but also during all other preceding cycles. The photo-charge accumulated on that pixel looks typically as follows:



For a medium exposure time pixel, the photo-charge accumulation is:



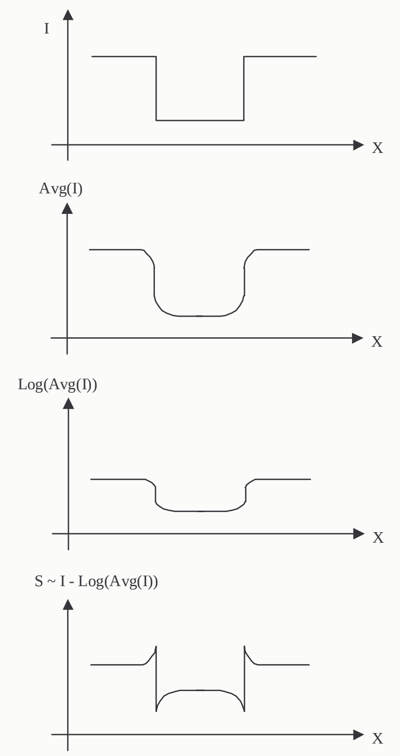
As can be seen from these figures, the Adaptive Sensitivity architecture achieves an inherent *anti-blooming* capability. Blooming is the spill-off of extraneous photo-charges from a saturated detector into neighboring pixels, resulting in bright fuzzy areas surrounding the image of a bright object. Conventional methods of anti-blooming employ process changes for the creation of special charge barriers and charge drains between pixels, and they make the process more expensive. Such means are usually unavailable in common digital CMOS processes, and thus blooming is often a serious limitation of integrated camera chips. As we see here, the Adaptive Sensitivity architecture avoids the problem.

Exposure Control

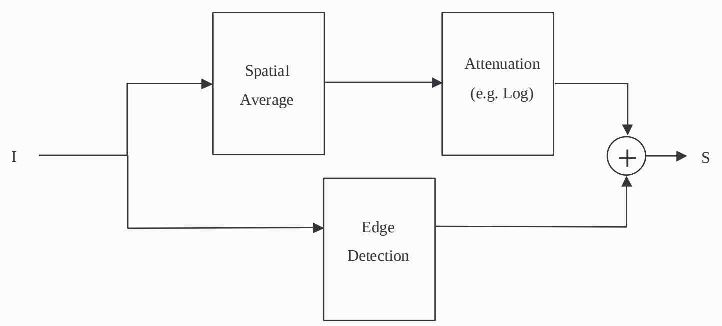
How is the proper exposure time determined? There is no control circuit on the pixel, and there is no mid-frame read-out in this scheme. Thus, exposure time is determined based on the reading of each pixel during the previous frame. We assume that the image is sufficiently stationary, which is a valid assumption in all video applications. Following any discontinuity (abrupt change in the image), conversion must be swift. In conventional video, an AGC process requires approximately 1 second (30 frames) to readjust exposure following a change in lighting, whereas the conversion of Adaptive Sensitivity is typically achieved in 2-3 frames (0.1 second).

Control Algorithm

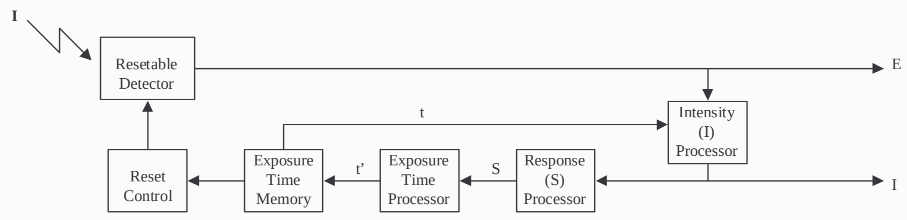
Say the incoming wide dynamic range light intensity is I_{ij} at pixel i,j, and the desirable reduced dynamic range output signal is S_{ij}. The latter imitates the eye response, and can be approximated by combining the attenuation of the spatial average and with edge enhancement:



A simple architectural model for this processing is:



This leads to the following control model. The *memory* holds exposure time codes for all pixels. The *reset control* converts those codes into reset signals (the Y values to be loaded into the Y register):



E_{ij} is the sensor's linear response at pixel i,j,

$$E_{ij} = c \cdot I_{ij} \cdot t$$

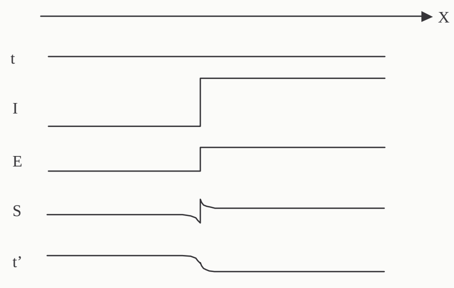
where c is a constant, I the incident light intensity, and t the exposure time. The goal of this system is to converge towards $E_{ij}=S_{ij}$ for every i,j. I is estimated by the *Intensity Processor*,

$$I = \frac{E}{c \cdot t}$$

According to the model above, S=NP(I) is some neighborhood function of I, and is computed by the *Response Processor*. Since we wish E→S, the *Exposure Time Processor* computes *t'*, the exposure time for the next frame, by substituting S for E:

$$t' = \frac{S}{c \cdot I} = \frac{NP(I)}{c \cdot \frac{E}{c \cdot t}} = \frac{NP(E,t)}{E/t}$$

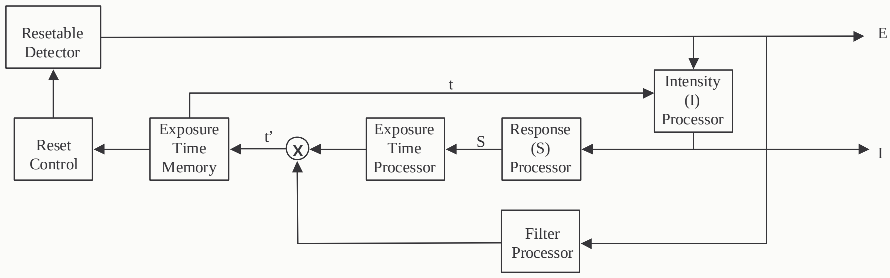
For example, suppose we start with a uniform exposure time and acquire a spatial step function:



In this case, t' is the complement of the spatial average of E, and is achieved in a single iteration.

Advantages of this scheme include:

- The system output E, which approximates S at steady state, is the direct (unprocessed) output of the sensor. All computation is limited to the feedback loop.
- Inherent anti-blooming
- Possibility of combining with other algorithms: t' can be multiplied by correction factors, which can affect filtering etc. for tasks such as sharpening, noise removal, color enhancement, and anti-aliasing. A possible modified architecture:

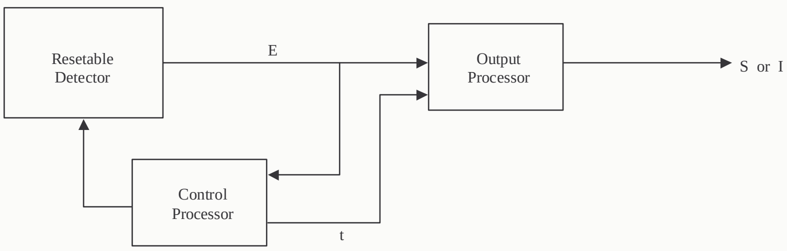


The drawbacks are:

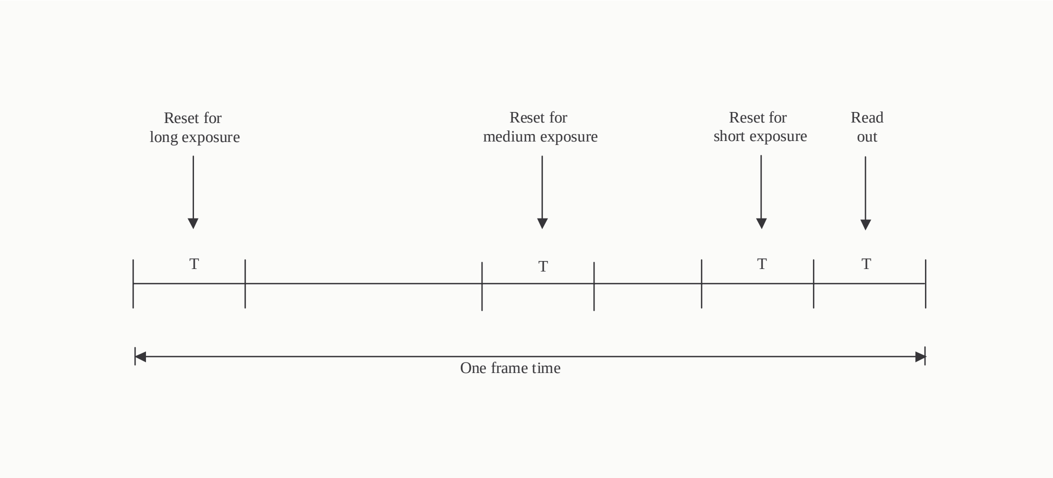
- A high resolution of *t* is required.
- The high frequency switching of the X and Y lines may induce noise.

A Simpler Adaptive Sensitivity Architecture

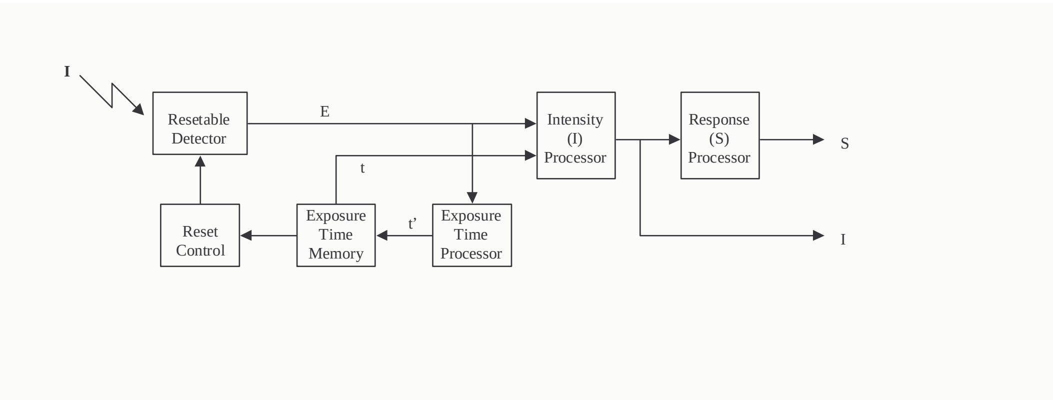
By adding a processor at the output, we can substantially reduce the required *t* resolution:



For instance, three different exposure times may suffice - a maximal time (full frame exposure), a minimal time, and some intermediate time, e.g. at mutual ratio 1:8:64,



The modified exposure time *t'* can be computed by simply comparing the output of each pixel to several threshold levels. The intent is that each pixel generates a response as close to the middle of its sensitivity function as possible, where it is most linear and most reliable. The control processor increases the exposure time (if possible) for pixels that are too low, decreases the exposure time (if possible) if the pixel value is too high, and retains the old value otherwise. The detailed architecture is:

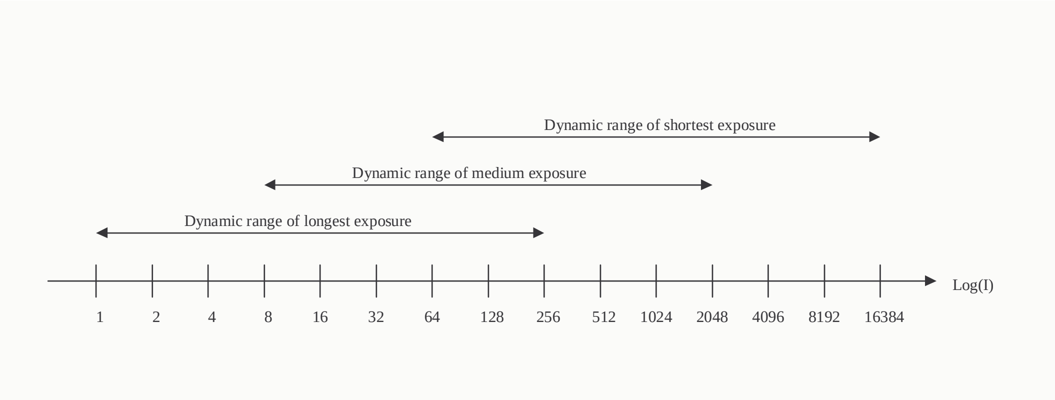


Advantages of the simpler architecture include:

- Lower noise due to reset signals
- Lower power consumption

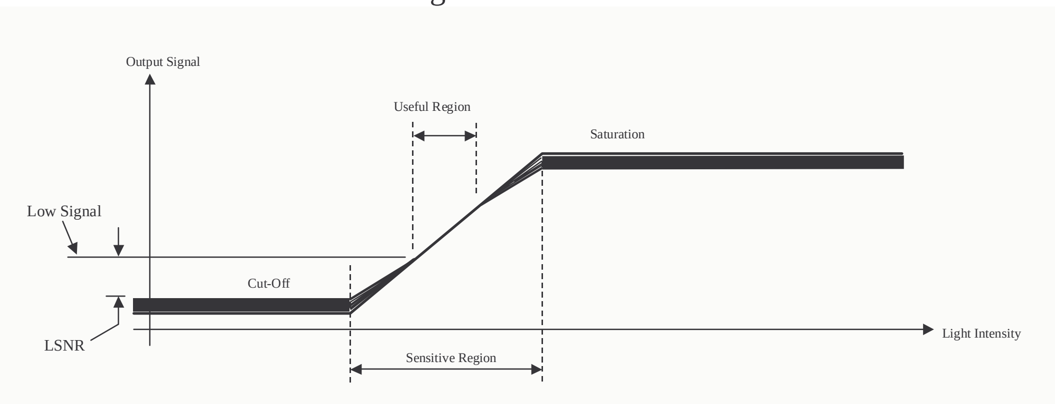
Estimating the dynamic range

Assume that the dynamic range of each pixel is D_{pix} , e.g. 1:256 (this is a typical number, and requires 8 bit digitizing). Further assume that the three different exposure times are related as 1:8:64. Then any pixel yielding an output I_{mid} with the medium exposure time is expected to yield $I_{short} = I_{mid} / 8$ at the shorter exposure, and $I_{long} = I_{mid} \times 8$ at the longer exposure. Thus, the medium range increases the dynamic range of the short exposure by a factor 8, and the long range adds another factor, resulting in ×64 total increase, from 1:256 to about 1:16,000 (from 48 dB to 84 dB, or adding 6 bits to the range). Graphically, on a log scale, the response of each exposure range and the total dynamic range in the example appear as follows:



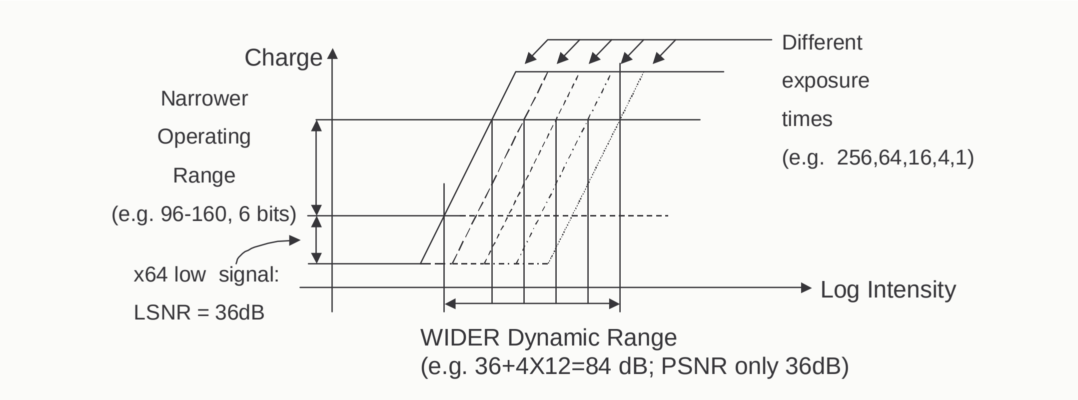
Increased Noise Immunity

Adaptive Sensitivity can be utilized not only for increasing the dynamic range, but also for noise reduction. Certain sensor types, especially those implemented in digital CMOS technology, achieve an inferior electro-optical performance as compared to CCD. As a result, their effective dynamic range is reduced. A typical response of an inferior detector is shown in the figure:



Under such constraints, it is desirable to assure that most pixels operate in the useful linear clean region, near the center of their sensitive region.

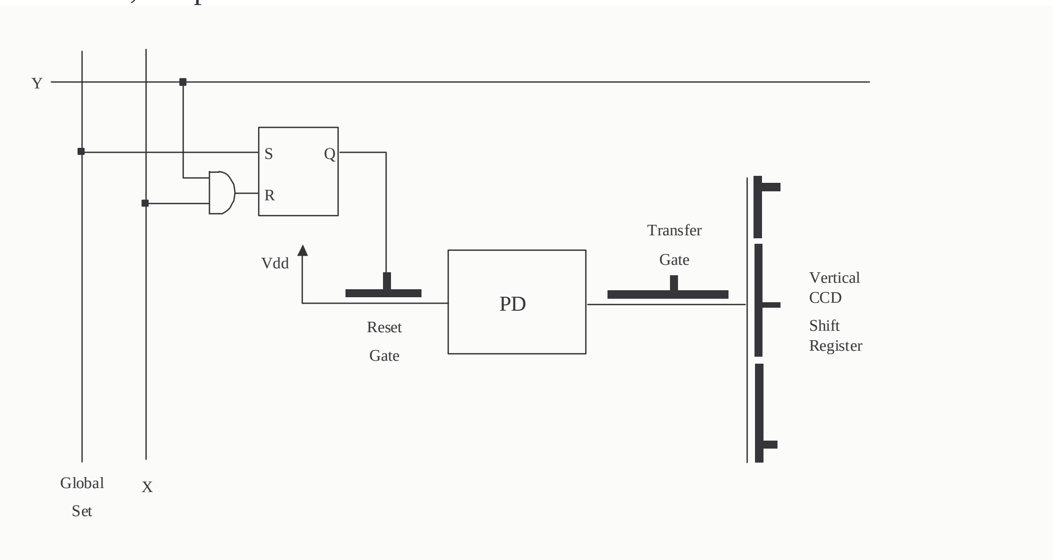
Dynamic range is often cited in terms of PSNR (Peak Signal to Noise Ratio). However, this model assumes that every signal that is above the noise floor is fully usable. As can be seen above, this is not the case in some difficult situations. To describe this fact, we define LSNR (Low Signal to Noise Ratio) as the ratio of the lowest usable signal to the highest level of noise. In well-behaved situations this ratio is 1:1 (0dB), but in noisy environments LSNR can be substabtially higher. In the example below LSNR=36dB. In such situations, Adaptive Sensitivity can be used to obtain a workable dynamic range, although most pixels must be operating near the middle of their sensitive range. In the example below, five different exposure times are employed, achieving a dynamic range of 84dB. For certain application relying on low-cost, noisy CMOS sensors integrated with digital logic, Adaptive Sensitivity or a similar method is an absolute must.



A Static Sensor Architecture

The sensor described thus far is reset by means of repetitive signals. The simpler architecture designed for a mere three reset cycles can span a dynamic range as wide as 84 dB (in the example above) or wider. But, at such wide ranges, the brighter illumination will saturate the pixels, and since there are long intermissions between successive reset signals, the saturated charges will surly bloom to neighboring pixels.

To counter this danger, a static architecture is described, which attaches a single bit memory (a flip-flop) to each pixel. The flip-flop is globally set upon the start of the frame, and its output continuously resets (discharges) the pixel. When the exposure of a pixel is to start, the flip-flop is toggled, and charge accumulation commences. For a CCD sensor, the pixel architecture is as follows:



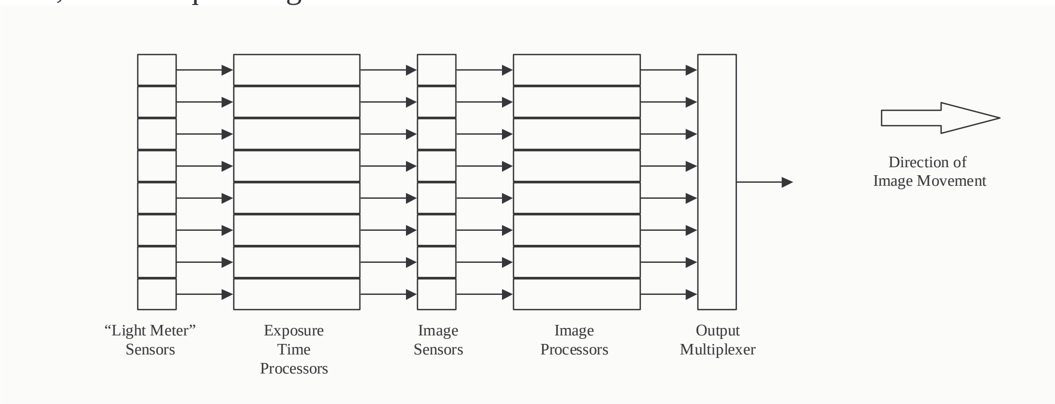
Discussion

The main challenge is the need to include a non-trivial circuit in every pixel, leading to possible degradation of fill factor and/or resolution. However, the progress in reducing feature size and in introducing more conducting layers promises to allow reasonable implementations of the technology.

Linear Adaptive Sensitivity

Linear scanning imagers can easily accommodate lots of circuitry per pixel, along the dimension perpendicular to the sensor (as long as the circuit pitch is the same as the sensors'). On the other hand, in many linear sensor applications the image is acquired only once, so a converging feedback algorithm cannot be employed.

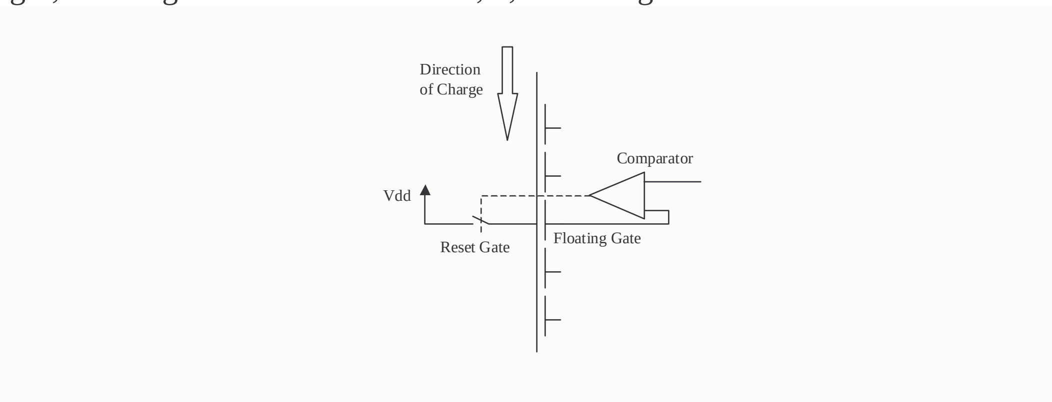
A second linear sensor, or sensors, can substitute the function of the feedback algorithm. The image is acquired once by the first sensor, analyzed and processed, and as a result the second sensor is adjusted. By that time the image arrives at the second sensor, and is acquired again:



TDI Adaptive Sensitivity Sensor

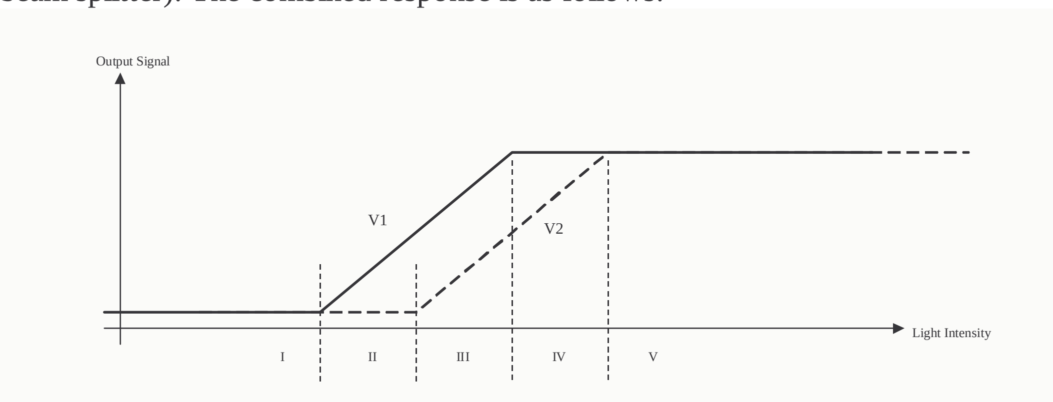
TDI (Time Delay and Integrate) are 1½-dimension sensors, designed for high speed, low light applications where the image moves too fast under the sensor. A parallel stack of interconnected linear sensors sample the same image multiple times. In CCD, the partly accumulated charge travel across the stack at the same rate as the image, so that the charge packet accumulates over many stages. Typical TDI sensors have 100-10,000 pixels along 16-128 parallel stages. For N stages, the sensitivity is increased by N, but noise increases by $\sqrt{N}$ so signal-to-noise improvement is also limited by $\sqrt{N}$ rather than N, and this in turn limits the dynamic range.

An Adaptive Sensitivity TDI CCD sensor has been designed. The accumulated charge is sensed non-destructively 'on the fly', allowing conditional discharge. The experimental sensor has 16 stages, and charge can be removed after 12 and/or 15 stages, resulting in accumulation over 1, 4, or 16 stages:



Adaptive Sensitivity with Non-Resetting Sensors

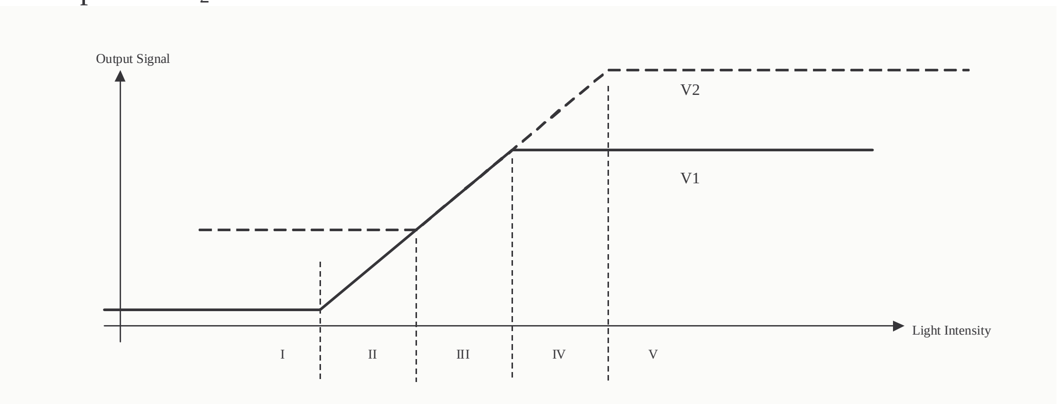
Dynamic range can be increased with Adaptive Sensitivity even when off-the-shelf sensors are employed. The sensor can be exposed twice for two different exposure times (successively in time, or by exposing two sensors simultaneously, e.g. through a beam splitter). The combined response is as follows:



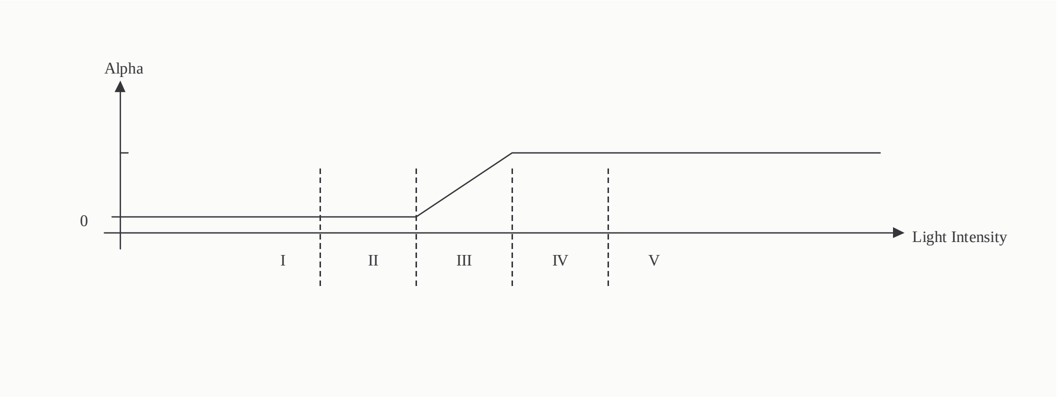
Five regions are defined:

A. Both sensors cut-off
B. Sensor V_1 responds, V_2 cut off
C. Both sensors respond linearly
D. Sensor V_1 saturated

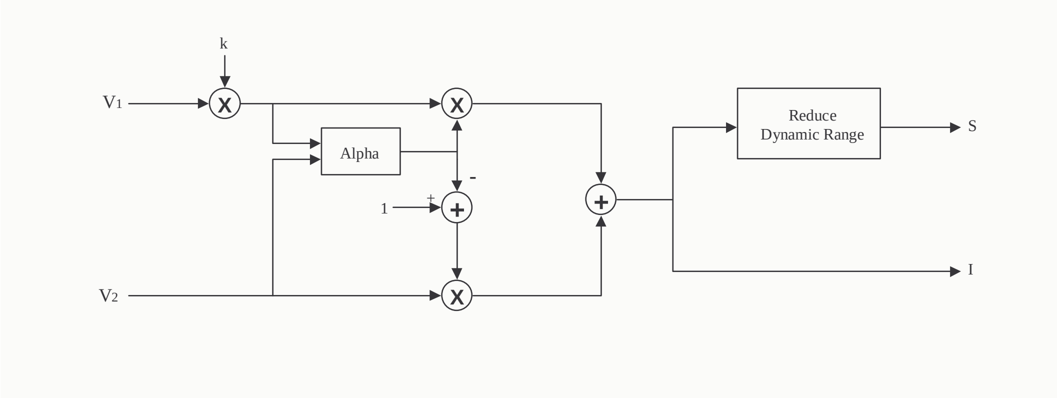
E. Both sensors saturated.
If sensor V_1 is exposed k times faster than sensor V_2, then, in region III, $V_2 = V_1 \times k$. We effectively unite the baseline of both sensors by applying the same scaling factor to all outputs of V_2 :



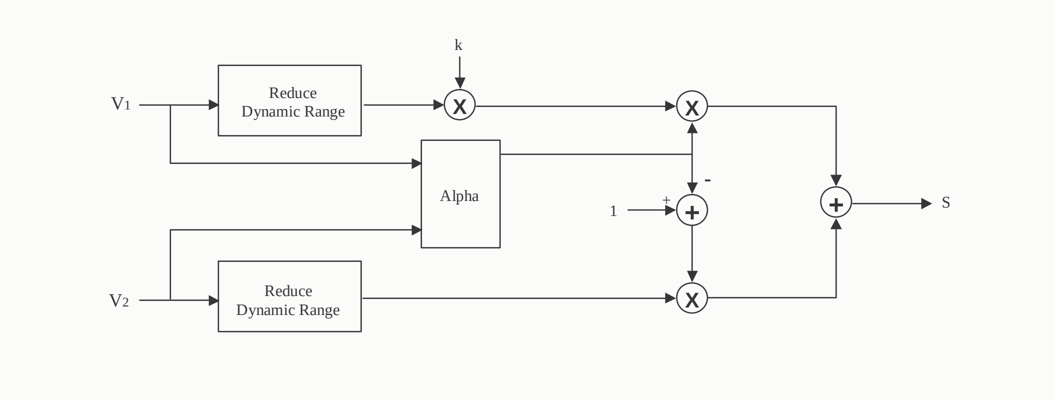
We create the combined signal by the linear combination $V_o = \alpha k V_1 + (1-\alpha)V_2$, where α is:



Note that α is also a function of the input (V_1 and V_2) and k. While the dynamic range of each individual exposure is limited to regions II+III (sensor V_1) or II+IV (sensor V_2), the dynamic range of the scaled and combined output V_o is expanded to regions II+III+IV. V_o can subsequently be reduced in the same manner that I is reduced to S above. A schematic block diagram for this process is:

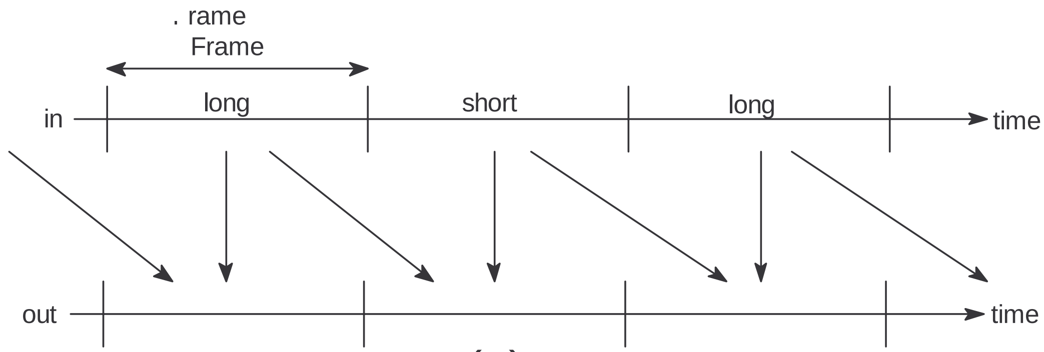


The order of operation can be reversed: First, the dynamic range of each of the inputs, V_1 and V_2 , is reduced, and then the results are linearly combined:



The drawback of the latter architecture is that *I* cannot be computed, but the advantage is that no high-precision computation is required (*I* may require 12-16 bit arithmetic, whereas *S* can be computed with merely 8-10 bits).

To apply this kind of Adaptive Sensitivity to a continuous video signal based on a single sensor, long and short exposure times are employed alternatively as follows:



(a)

The principal problem is that the maximal dynamics supported is twice as slow as the frame rate. This issue can be addressed by operating the sensor twice as fast, by processing interlaced video fields rather than frames, and by employing two sensors simultaneously. Additional issues are involved in the case of color processing.