embedded VISION SUMMIT 2018

Designing Vision Front Ends for Embedded Systems

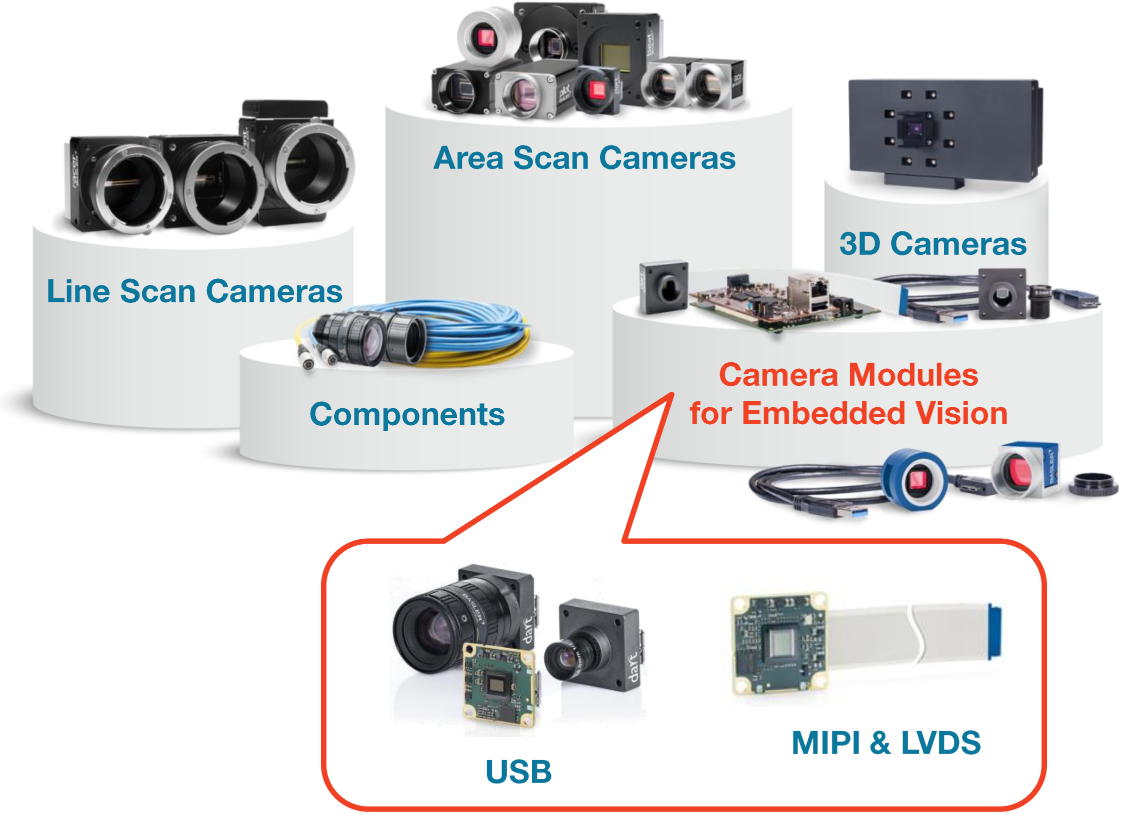
Dr. Fritz Dierks, Director Product Marketing and Development

BASLER

22.05.2018

Who is BASLER ?

- **Market leader for industrial cameras**
  - 150M € revenue in 2017
  - 450k Units camera shipped
- **Headquarter in Germany**
  - Asia focus (46%)
  - EMEA (28%)
  - Americas (26%)
- **Partnership with QUALCOMM**
  - Made ISP available under Linux



Agenda

- **Choosing the Right Sensor**
- **The Image Signal Processor**
- **Designing the System**

→ Denotes rule of thumb

Choosing the Right Sensor

- Sensitivity / Image Quality
- Resolution / Pixel Size
- Mono / Color
- Rolling / Global Shutter

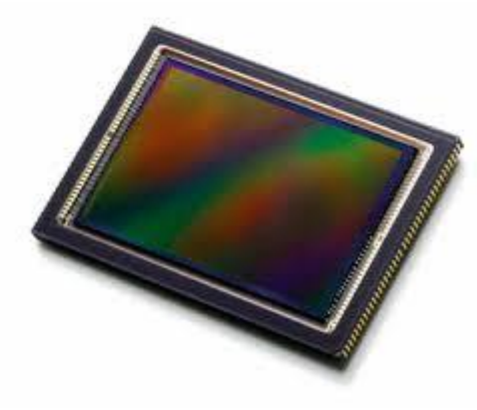


Image Quality and Sensitivity

A perfect image is degraded by

- **Temporal & Spatial Noise** which is measured by **Signal to Noise Ratio** (SNR)
- **Artifacts** (stripes, defect pixels, …); various measures exist; use visual inspection

➔ The less noise and artifacts the better the **Image Quality**

The temporal noise strongly depends on the **Amount of Light** available.
➔ The more light, the less noise, the better the image quality.

For a given amount of light the camera delivering a better image quality is more **sensitive.**

➔ Comparing for sensitivity makes sense only under comparable lighting conditions

Signal to Noise Ratio

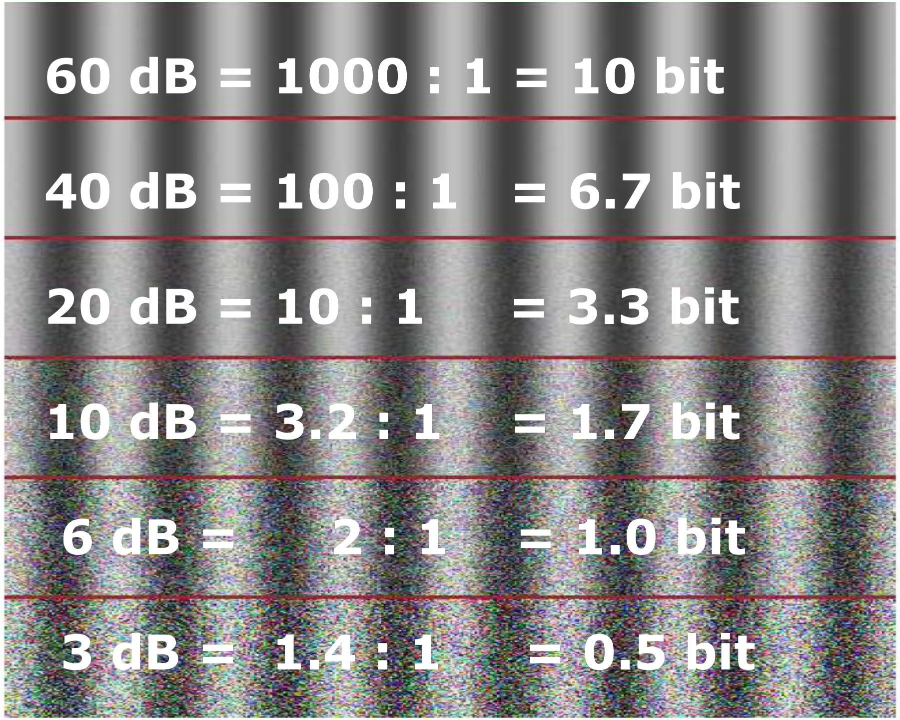
- Visual noise = temporal + spatial

➔ Only temporal noise is relevant for sensitivity

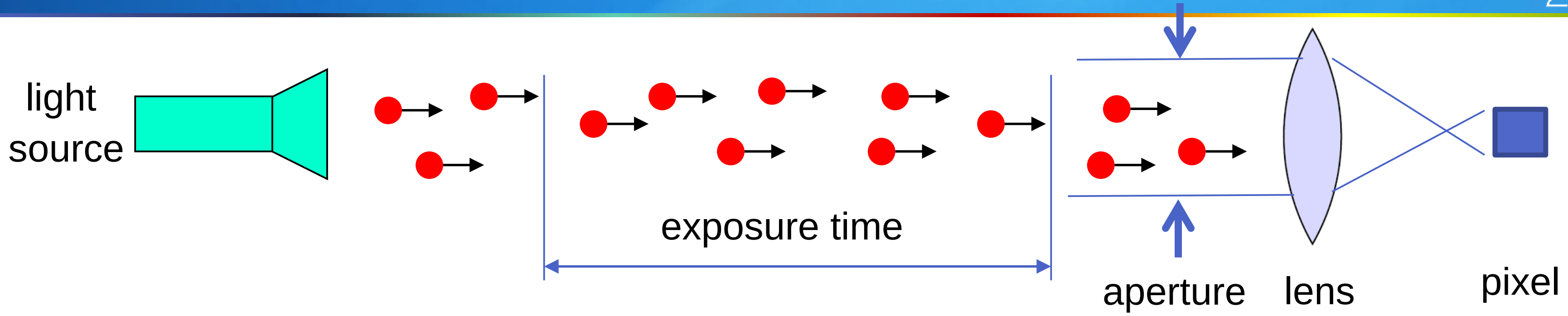
- SNR = (mean signal – dark) / temp noise rms
- Neither depends on Gain nor Offset
- Do **not** judge sensitivity based on brightness

➔ Use tool displaying temporal noise

- 40 dB = “excellent”,
- 20 dB = “acceptable”
- 0 dB = “threshold”



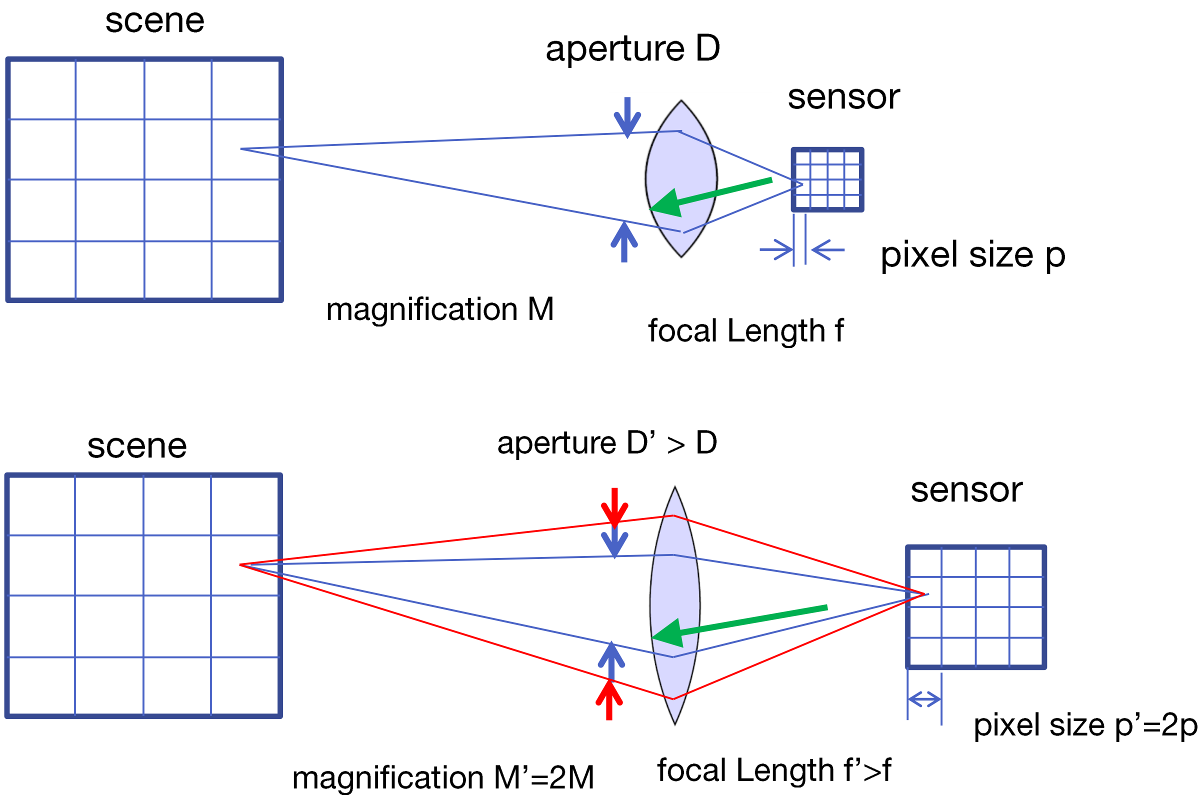
Amount of Light



- Amount of light = number of photons hitting a pixel during exposure time
- SNR light = sqrt(#photons) due to Poisson statistics

→ Brighter illumination → more light (beware: illumination is more expensive)
→ Longer exposure time → more light (beware: motion artifacts occur)
→ Larger aperture opening → more light (beware: depth of field degrades)
→ Larger pixel → larger aperture → more light (beware: larger sensor & lens are expensive)
→ Lower resolution → more light (don't use more pixels than you actually need)

Resolution and Pixel Size



double pixel size

→ larger image

→ lower refractive power

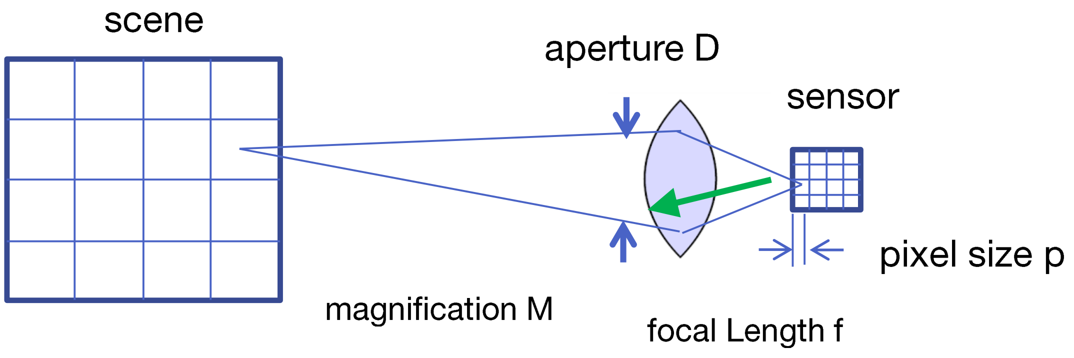
  → larger focal length

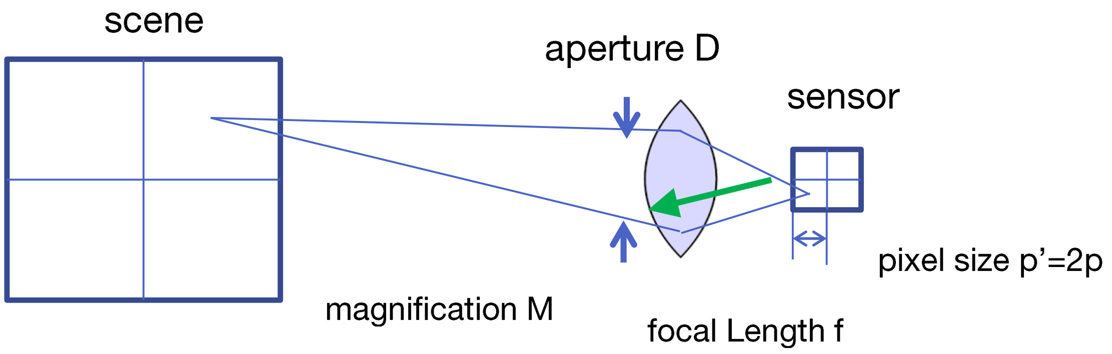
  → larger lens

→ larger max aperture size

  → more light collected from scene

Resolution and Pixel Size





Larger Pixel

→ 2x2 binning

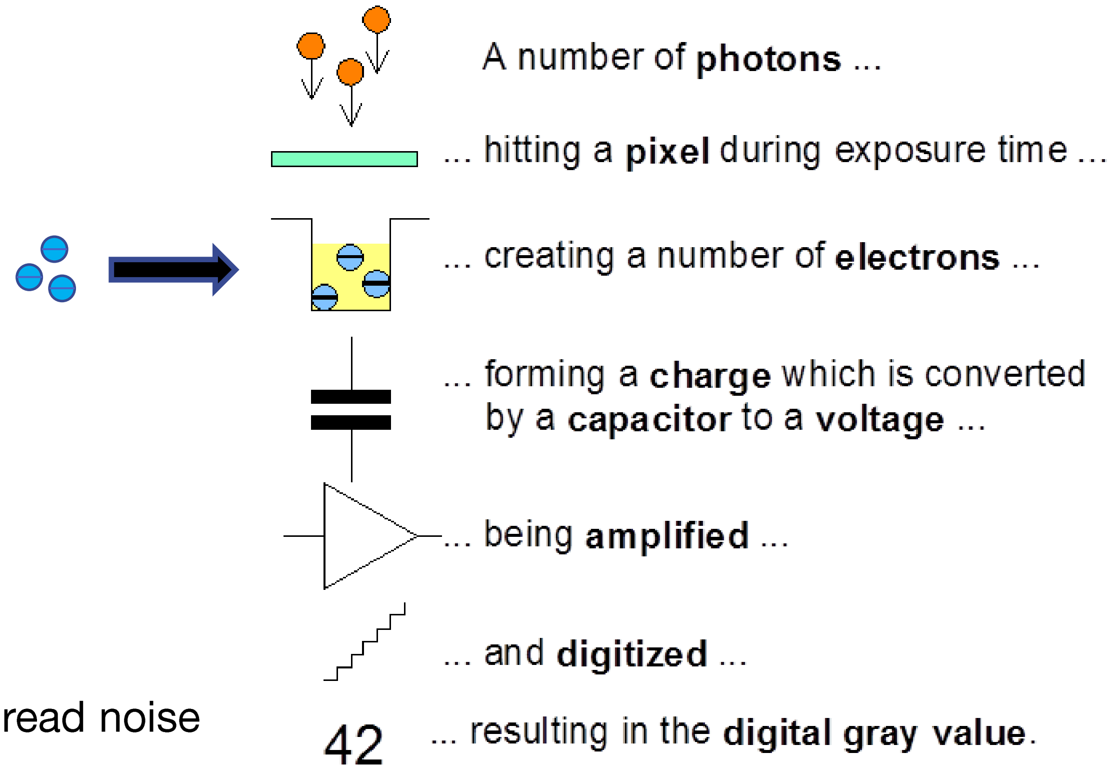
→ fewer #pixels

→ more light per pixel

→ less resolution

Model of a Single Pixel

The noise per readout can be modelled by a constant number of electrons

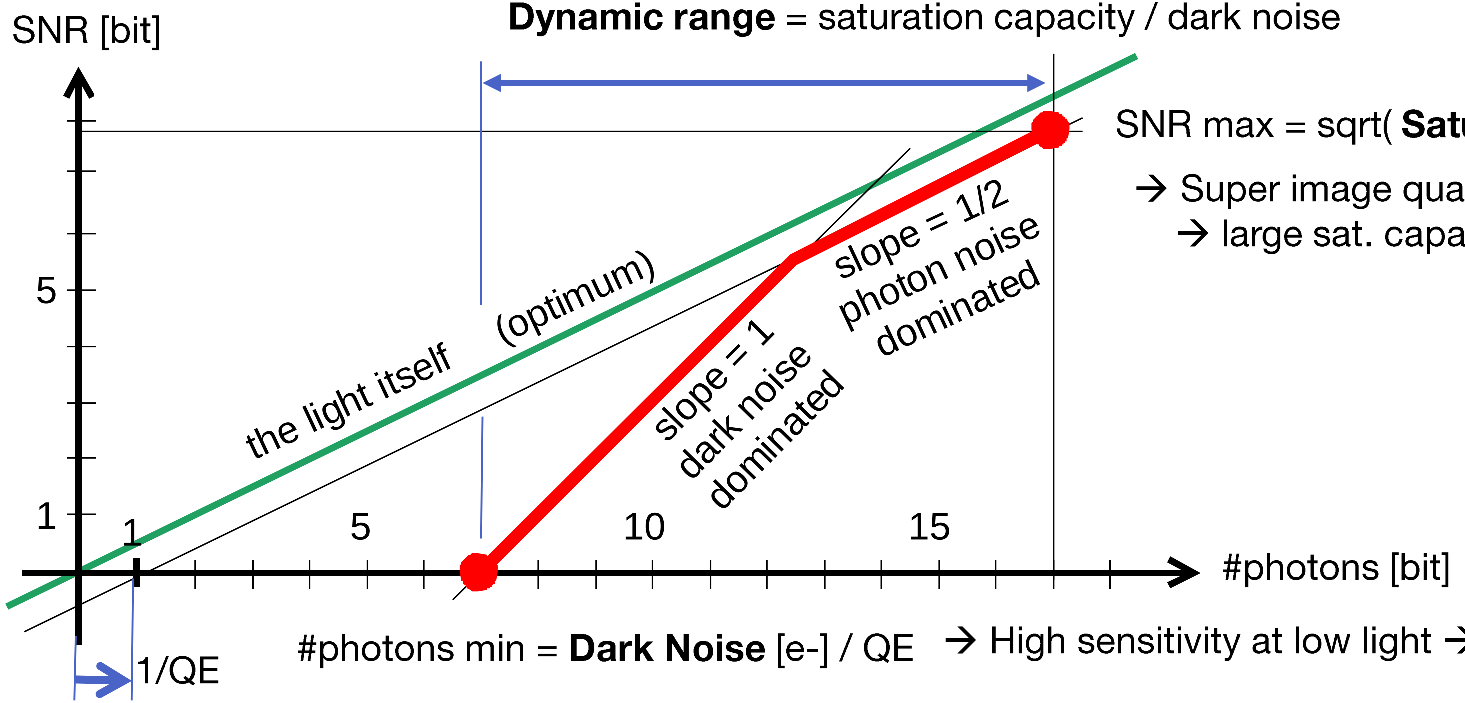


Key Performance Indicators

- Quantum Efficiency (QE)
- Read Noise [e-]
- Saturation Capacity [ke-]
- Dynamic Range = sat cap / read noise

Key Performance Indicators for Sensors

Dynamic range = saturation capacity / dark noise



SNR max = sqrt(**Saturation Capacity** [e-])

→ Super image quality at bright light
→ large sat. capacity → large pixels

#photons min = **Dark Noise** [e-] / QE → High sensitivity at low light → low dark noise

Quantum Efficiency (QE) = #photo electrons / #photons → no big difference for visible light but most relevant for NIR light

EMVA 1288 Standard

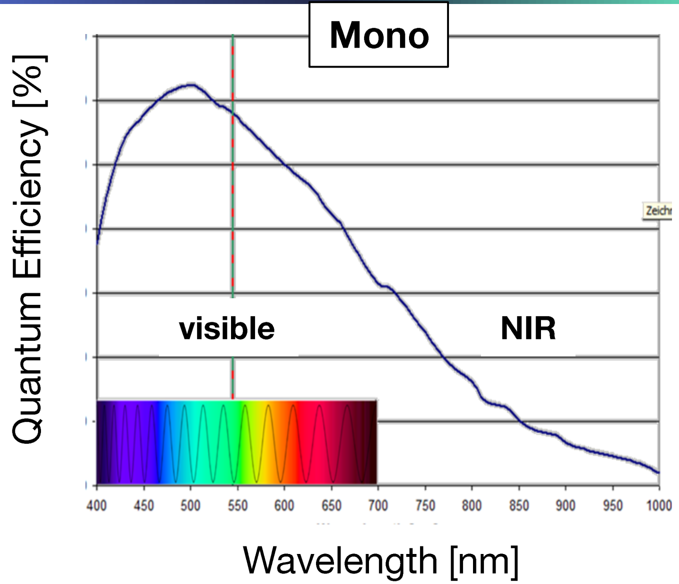
- EMVA 1288 standard defines how to measure key sensor/camera performance indicators

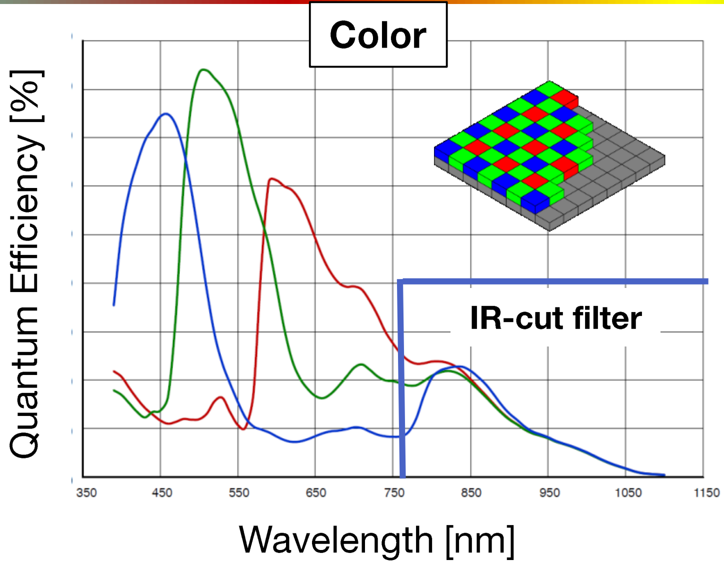
1288
EMVA Standard Compliant

- Comparing non-standard data from different vendors typically does not work
- Many machine vision camera vendors publish standard data for their cameras

Sensor	IMX 226		IMX 178
Resolution	12 MPx	>	6 MPx
Optical format	1 / 1.7”	≈	1 / 1.8”
Pixel Size	1.85 µ	<	2.4 µ
Dark Noise	3 e-	≈	3 e-
Saturation Capacity	11 ke-	<	14 ke-
SNR max	100 : 1 40 dB	<	125 : 1 42 dB
Dynamic Range	3500 : 1 71 dB 11.8 bit	<	4500 : 1 73 dB 12.1 bit
Quantum Efficiency @545 nm	80%	≈	81%

Mono / Color



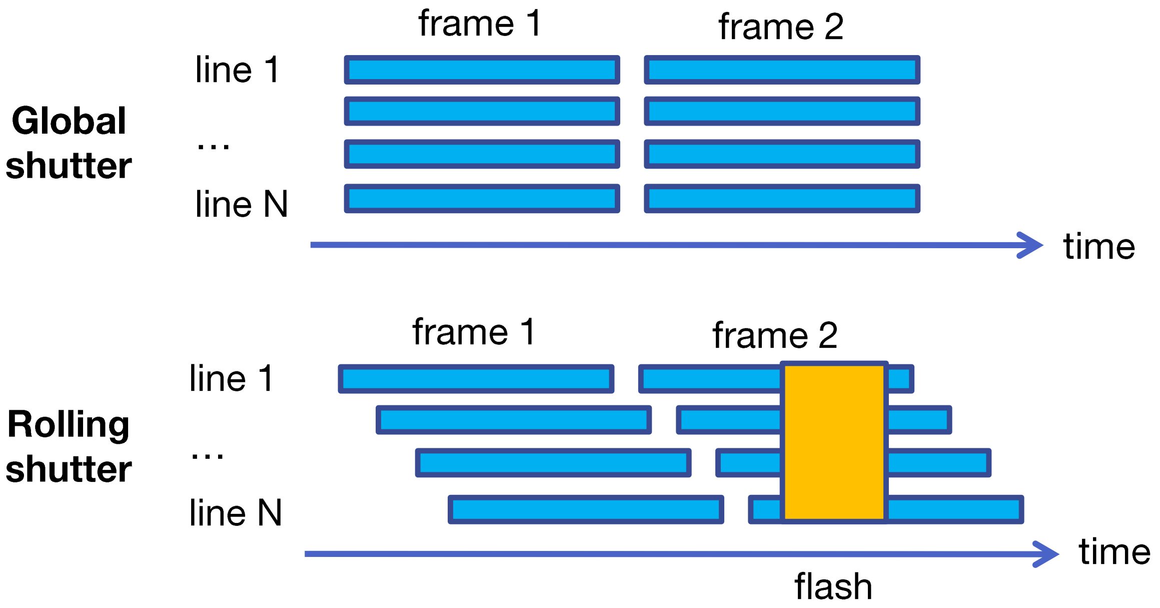


- Most color sensors use Bayer pattern filter (red, 2 x green, blue)
  → 2 out of 3 color components per pixel are thrown away → mono is more sensitive
  → true spatial resolution is replaced by guesswork → use color only when necessary
- Color filters don't work in near infrared (NIR) → add external IR-cut filter
- For NIR illumination (LED flash), use special sensors with high NIR sensitivity

BASLER

Rolling / Global Shutter

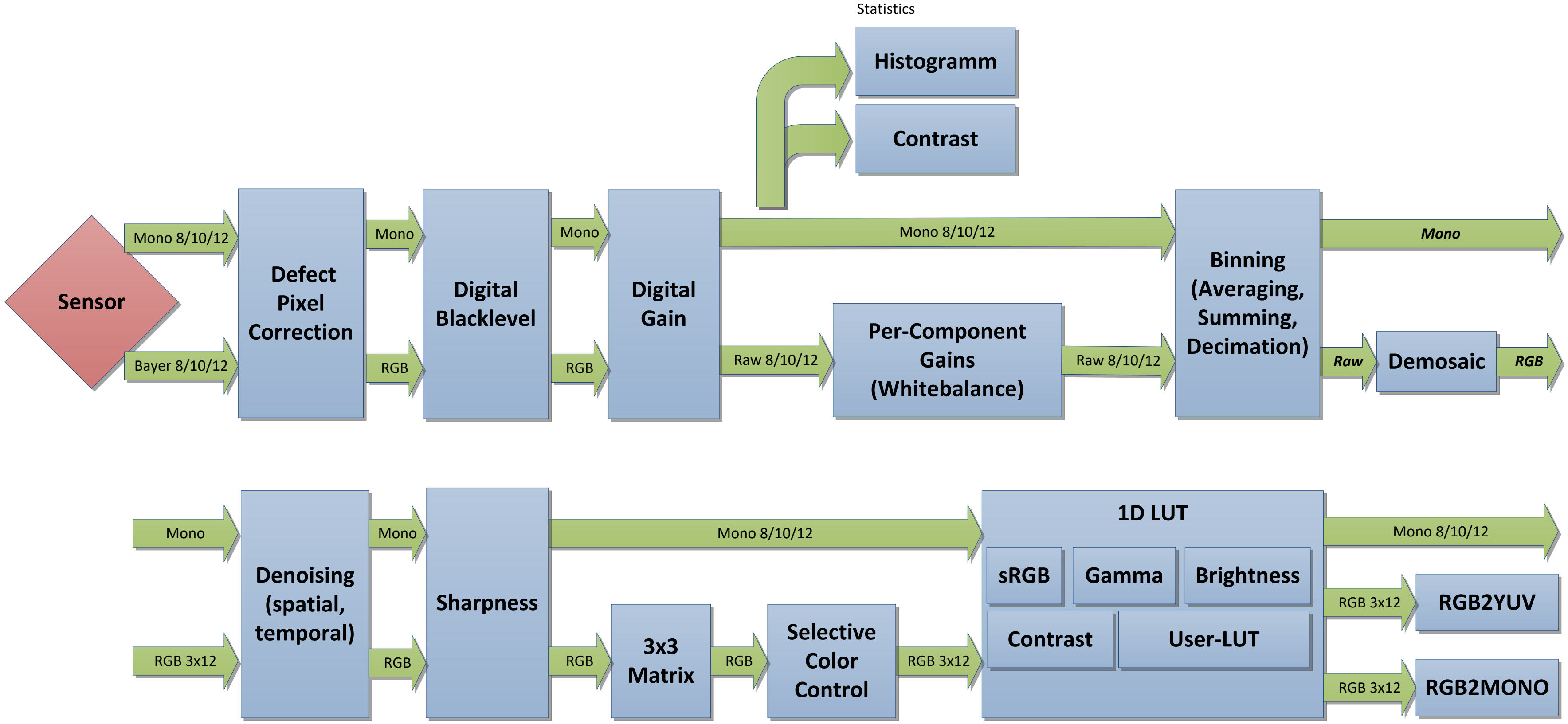
- Rolling shutter sensors need fewer transistors per pixel
  → smaller/cheaper
- Global shutter sensors "freeze" fast moving objects
  → in many cases rolling shutter sensors are fast enough
- You can freeze the image by using a flash pulse while all lines are exposing



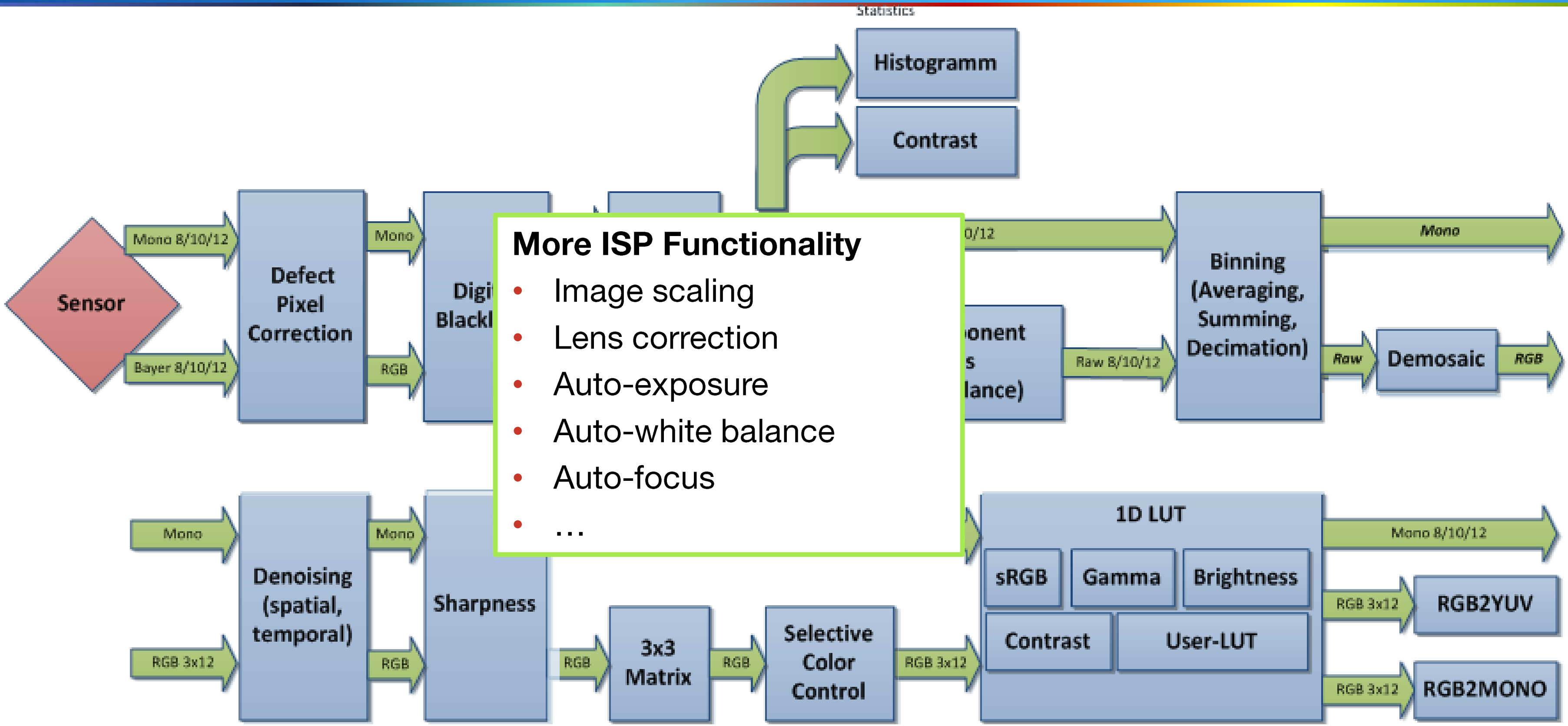
The Image Signal Processor (ISP)

- ISP Stages
- Viewing vs Machine Vision
- ISP Supplier

ISP Overview (simplified)



ISP Stages



More ISP Functionality

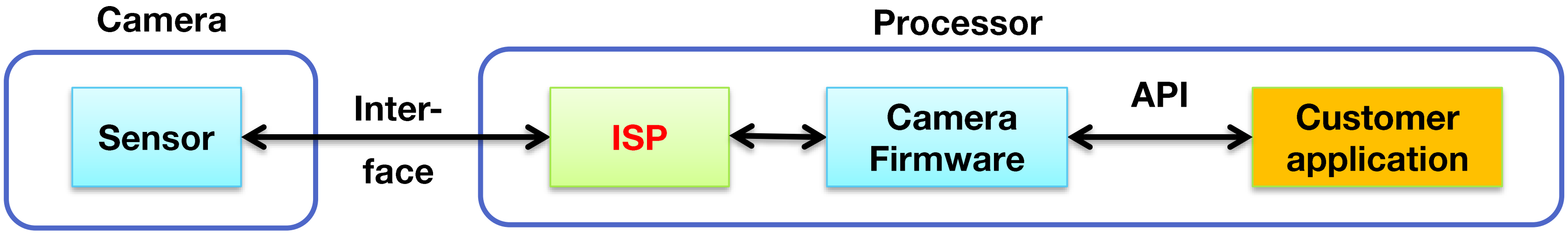
- Image scaling
- Lens correction
- Auto-exposure
- Auto-white balance
- Auto-focus
- …

Tune ISP to Application

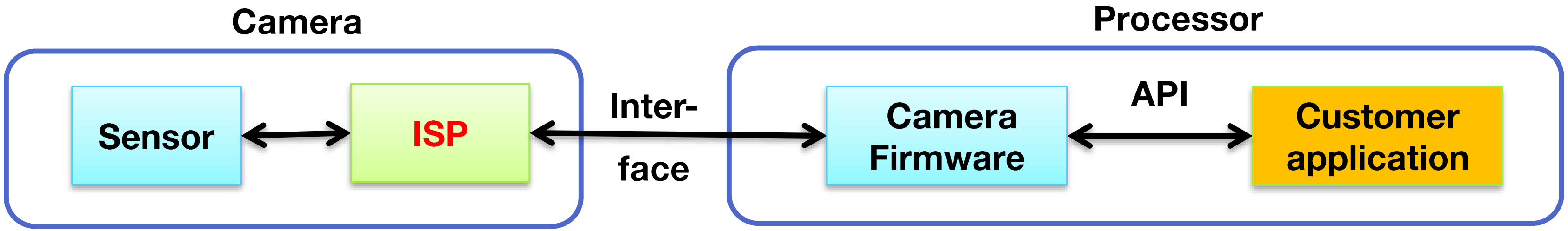
- **Viewing** : Image to be looked at by human being
    - Image must "look good"
    - Take into account human eye/brain
    - What looks good depends on what people are used to
- **Machine Vision** : Image to be processed by SW algorithm
    - Image must be "true"
    - Make visible what the vision algorithm is seeking
    - Some viewing improvements disturb machine vision (sharpness, de-noising)

→ Make sure you (or your supplier) can tune the ISP

ISP Supplier



- ISP can be part of the processor → low system cost, high performance
- ISP can be part of the camera
  - Sensor with integrated ISP → limited selection, limited performance
  - Dedicated ISP chip → higher system cost, processor independence



Designing the System

- Lens Type
- Modular Design
- SDK

Lens Type

Integrated Lens

- Small form factor
- Limited selection of optical parameters
- Autofocus available
- No customizing except for really large volume
- Longevity problems

Separate Lens

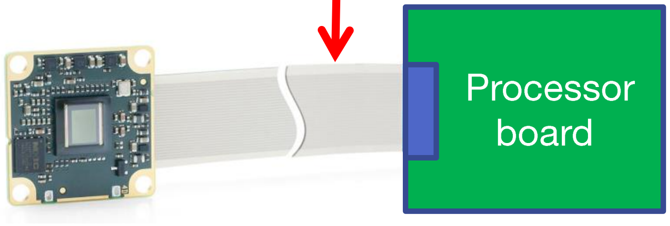
- Optimal fitting lens to application (focal length, …)
- Long-term availability (sensor & lens separately)
- Customization for medium volume

BASLER

Modular Design

Sensor Multitude

ON Semiconductor® · SONY · OmniVision



One generic Interface

- Video + configuration
- GPIOs (e.g. trigger & flash)
- Single voltage (5V)
- Easy to integrate

- For **small & medium volume** use camera module
  - Unified hardware and software interface
  - Pick suitable module from portfolio
  - Re-use knowhow and HW/SW designs
- For **large volume** use tailor-made design
  - Start with module and port to tailor-made with raising volume

Which SDK to Use?

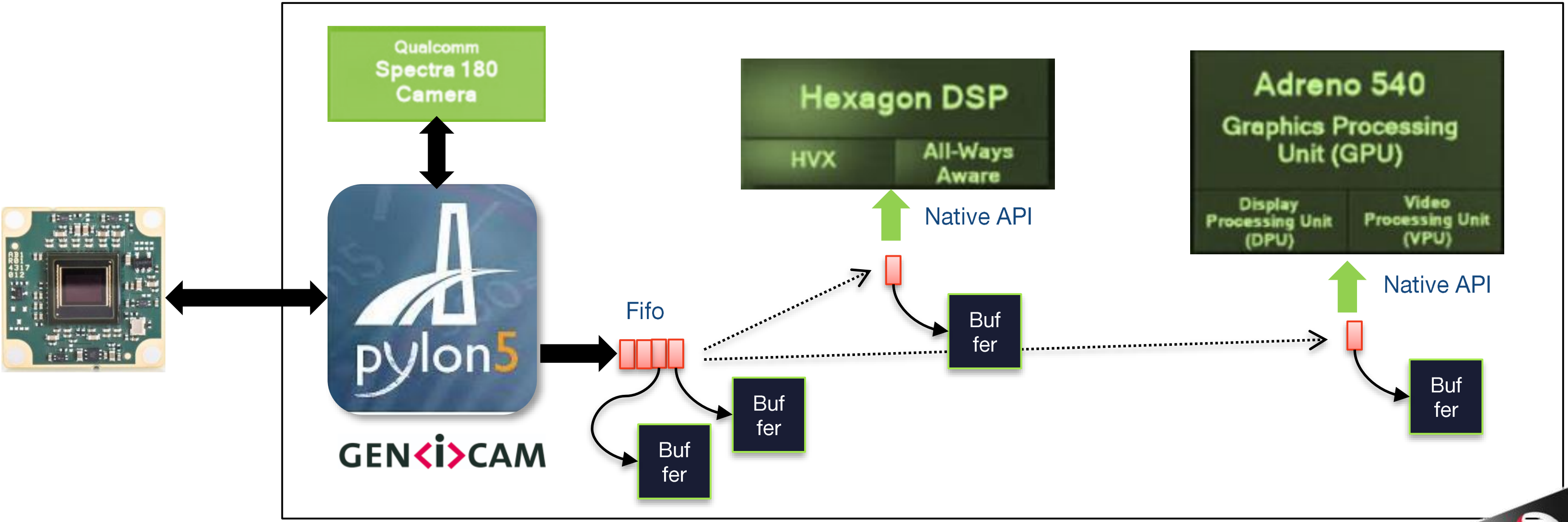
Android

- **Camera2 API** - sophisticated interface giving full control

Linux

- **Video4Linux** – low level interface
- **Gstreamer** – widely used for streaming applications
- **GenICam** – standard for industrial cameras
  - Device discovery, configuration, video & event delivery, chunk data
  - Plug & Play by defining >300 camera features
  - GenICam has >180 member companies
  - Used for all machine vision camera interfaces

SDK – GenICam based Example



- Client code receives buffer handles through Pylon's fifo
- Buffer handles can be used with native HvX and GPU code
- Pylon gives configuration access to camera and ISP

Conclusion

- Picking the right sensor for your task is key
- Make sure you can tune the ISP to your application
- Start with a modular design switching to tailor-made after volume ramp up
- Get yourself a competent camera partner ☺

Resources

- Basler web site: www.baslerweb.com
- EMVA 1288 standard: http://www.emva.org/standards-technology/emva-1288/
- GenICam standard: www.genicam.org