Propagating Random Telegraph Noise in Las Cumbres Observatory's CMOS data

Daniel Harbeck[a], Curtis McCully[a], Matthew Daily[a], and Prerana Kottapali[a, b]

[a]Las Cumbres Observatory, 6740 Cortona Drive, Suite 102, Goleta, CA, USA, 93117

[b]Department of Physics, Durham University, South Road Durham, DH1 3LE, UK

ABSTRACT

At Las Cumbres Observatory we have introduced CMOS-based imagers (QHY600 with Sony IMX455 sensors) as the main cameras in our global network of ten 35-cm telescopes. These telescopes are the foundation of our education program (Global Sky Partners) and are also used for professional astronomy (e.g., TESS planet transit follow-up). The deployment of CMOS detectors in the small telescope network also serves as a pathfinder for a possible future introduction of large-format CMOS cameras to the observatory's 1-meter telescope fleet. In this presentation, we report some of the first lessons learned from using CMOS cameras in a professional astronomy production environment.

In particular, we focus on characterizing and treating random telegraph signal (RTS), an additional noise component in CMOS cameras. While RTS can be neglected in very high Signal to Noise (S/N) situations, such as in planet transits, it might bias photometry in the low S/N case. Traditional CCD data reduction has no established data processing paradigms for this additional noise component. At Las Cumbres Observatory we have developed procedures to first model RTS of a detector, and then seed and propagate a simplified per-pixel noise model in our data processing pipeline BANZAI. We are exploring more advanced mitigation studies, such as using modeled multi-modal RTS distributions as a prior for maximum likelihood fitting when stacking images. Such a method is capable of estimating a pixel's true readout value with smaller samples than simple averaging.

Keywords: CMOS, Random Telegraph Noise, Data Pipeline, Las Cumbres Observatory, Sony IMX455

1. INTRODUCTION

Las Cumbres Observatory operates a global network of 25 robotic 0.4m, 1m, and 2m telescopes to enable time-domain science and education programs.[1] A cornerstone of the outreach program (Global Sky Partners*) is the 0.4-meter network that was originally built upon ten 40cm Meade telescopes, equipped with optical imagers (SBIG ST6303†).

*`https://lco.global/education/partners/`

†https://lco.global/observatory/instruments/sbig-stl-6303/

In 2022 we started a project to refurbish the 0.4m telescope fleet. The telescope tubes were replaced with new PlaneWave DeltaRho 360 telescopes, and the imagers were replaced with QHY600 cameras, which use a Sony Exmor IMX455 CMOS detector. As CMOS sensors are a fairly new addition to ground-based optical astronomy, we adopted our procedures to accommodate CMOS data in the observatory's workflow, from the commissioning of a camera to the daily automatic data processing.

CMOS sensors benefit from commercial development and production in mass applications such as photography and surveillance cameras, whilest reducing the cost per detector compared to custom astronomy-oriented developments. Meeting key requirements on noise, pixel scale, imaging area, and software support in Linux, the QHY600 camera was selected over a conventional CCD system for its very competitive price point, as well as its availability and track record in the amateur astronomy world. The upgrade project was described in a contribution to the SPIE 2024 Ground-based and Airborne Instrumentation for Astronomy X conference.[2]

All data originating from Las Cumbres Observatory's telescopes, including the refurbished 0.4m fleet, are automatically transferred to an archive[‡] hosted in the Amazon AWS cloud, and processed by a data pipeline (BANZAI[3]). The imaging pipeline of BANZAI is a standard CCD data processing system, with overscan, bias, and dark frame subtraction, flat field correction, first-order source extraction, and world coordinate system generation based on catalog cross-matching. CMOS sensors were not considered in the initial design of the BANZAI pipeline, but we will see later in this paper that only minimal changes were needed to accommodate the new CMOS sensors.

We will briefly review new issues with CMOS detectors compared to CCDs, explore ideas for mitigating the effects of Random Telegraph Noise (RTN), and discuss the potential benefits of advanced RTN modeling for future data processing in astronomy. Finally, we will describe how we integrated simplified, but application-appropriate CMOS-specific processing steps into the established BANZAI data pipeline at Las Cumbres.

2. WHAT'S NEW WITH CMOS SENSORS?

CMOS sensors differ from CCDs as each pixel is equipped with an individual amplifier: read-noise and conversion gain are properties of individual pixels. For LCO's ground-based imaging applications, the per-pixel gain can easily be treated by established flat fielding procedures and is not of further concern in this paper. However, applications that need detailed noise and gain modelling may require detailed per pixel gain measurements as well.

Some CMOS pixels can exhibit additional non-Gaussian noise behavior, known as Random Telegraph Noise (RTN), which is caused by charge traps in the MOSFET amplifiers. Per pixel noise and RTN are well described in the literature, both from an electrical engineering point of view,[4] as well as for practical applications and workarounds to RTN, e.g., in life sciences.[5] Recently, Alarcon et al.[6] published an in-depth study of the IMX455 sensors in the QHY600 camera for use in astronomy, and for details about the camera performance, including RTN, we refer to that paper.

[‡] `archive.lco.global`

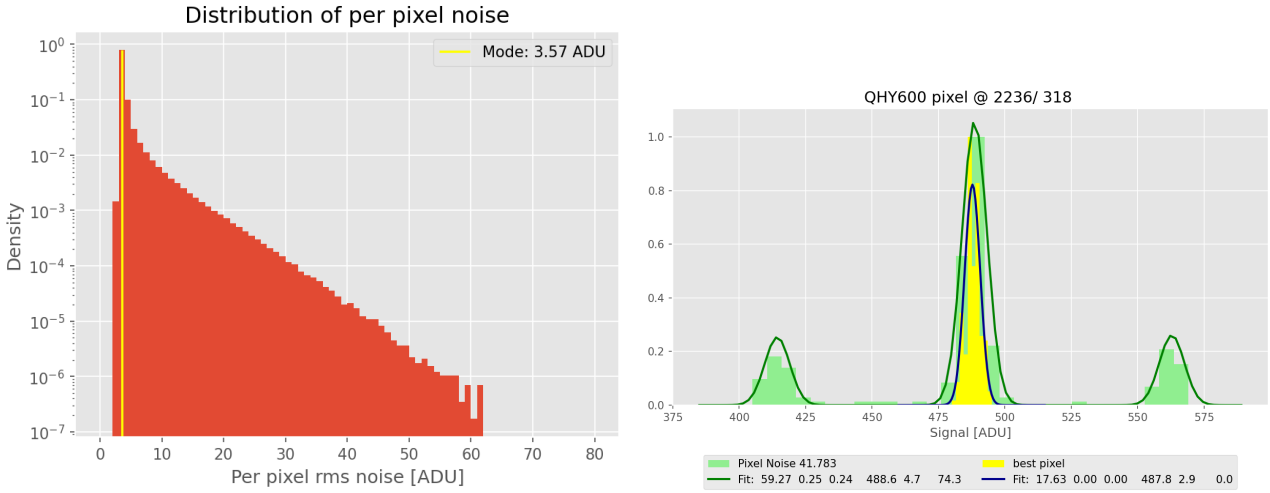

Figure 1: Left: Histogram of the per pixel standard deviation of the signal in a series of 200 bias exposures. Right: Pixel distribution of the lowest noise pixel (yellow) and a high noise pixel (green). A three-component Gaussian model is fitted to the distributions; the legend shows the parameters of the fit according to Eqn. 1. The noisy (measured rms of 41.78 ADU) pixel's best fits reveal a peak separation of ±74 ADU (about 58e$^-$) from the central peak, with relative peak heights of 0.25 to the central peak, respectively.

The RTN is a most intriguing behavior: The low current involved in CMOS detector's MOSFET circuits makes the amplification sensitive to charge traps, possibly resulting in two or more distinct output currents for a given MOSFET gate voltage. Combined with correlated double-sampling (CDS), the statistical interplay between trap capture and release time constants and CDS clocking pattern can result in a multimodal response of a pixel at a given input level.[4] This phenomenon is known as random telegraph noise (RTN), also sometimes described as salt & pepper noise for its appearance.

2.1 Random Telegraph Noise in the QHY600 Camera

Readnoise in a CCD is commonly measured from a single image as the standard deviation from many pixels, with some correction for fixed pattern noise. For the analysis of CMOS detectors in this paper, we will measure the noise from a series of images (typically 200 bias frames) for an individual pixel. Fig.1, left, shows a histogram of the per-pixel noise of a subsection of QHY600 camera's detector. The mode of this distribution is close to the readnoise specified for a camera.

The noise histogram exhibits a tail of noisy pixels that extends to an rms noise of $\geq$ 60 ADU, which corresponds to about 42 e$^-$ at the readout mode's conversion gain of 0.78e$^-$/ADU. Fig.1, right, shows individual readout value distributions of the lowest noise pixel (yellow) and an exemplary noisy pixel (green). The low-noise pixel shows a single narrow distribution as expected for the low readnoise of 3.1 e$^-$ of the camera; a CCD pixel would result in a similar distribution. In contrast, the noisy pixel exhibits a trimodal distribution — this behavior is expected for RTN for a single trap in the amplifier circuitry.[4] Note how the output amplifier is not significantly noisier by itself — the three individual distributions of the three peaks are still narrow —, but the trimodal distribution due to the RTN causes this wider distribution.

The noise characterization of a CMOS pixel by rms noise alone is incomplete. The fraction of pixels with an elevated noise above 10 ADU $\approx$ 37%, which is a non-negligible fraction in the QHY600 cameras and could cause concern for precision astronomy applications.

2.2 When Random Telegraph Noise is of concern

Fig. 1 reveals that the per pixel rms noise of the QHY600 camera evaluated here ranges from about 3 ADU to 60 ADU. This corresponds to a readnoise range of about 2.3e$^-$ to 40e$^-$ with a mode of the readnoise distribution of $\approx$ 3.6ADU= 2.8e$^-$. For a symmetric trimodal distribution, the derived rms noise corresponds to about $\sqrt{2}\times$ rms peak-to-peak separation. A single measurement at a pixel with a measured rms noise of 20e$^-$ could be randomly wrong by $\pm$28e$^-$; let's assume 30e$^-$ in the following:

High S/N use cases, such as a time series of a planet transit, will only see a minor impact from RTN compared to photon shot noise or other systematics, such as atmospheric scintillation. The impact will be relevant for low S/N situations where only very few observations can be taken at an epoch and a precise flux measurement is required. Such might be the case, e.g., when CMOS detectors are used in a UV small satellite to distinguish supernova models of the early explosion phases.[7] Most pixels of a CMOS detector are well-behaved, but a single measurement of a target positioned onto a high RTN pixel might be compromised.

As an example (ignoring read noise and systematics), a typical signal to noise (shot noise) in a photometric supernova small-satellite follow-up may be of the order S/N=20, corresponding to a measured object flux of 400e$^-$. A measurement that is offset by 30e$^-$ due to RTN-affected pixel in the target's aperture would introduce an *additional systematic* 7.5% error for a single measurement that would have been at 5% measurement when based on photon statistics alone.

Further contributing factors to systematic photometry errors caused by RTN would be in the measurement of a background signal in a sky aperture and the propagation of RTN through calibration products such as bias and dark calibration frames.

3. CHARACTERISATION OF INDIVIDUAL PIXELS

We will explore if the RTN behavior can be modeled and then used to improve subsequent data processing. In particular, as it is the easiest case, we will focus on the process of stacking pixels in a bias calibration product:

We make two key assumptions:

1. The RTN behavior is an immutable property of a pixel under constant operating conditions, and

2. The mean level of a bias image is also the expectation value for each individual pixel, especially in the case of asymmetric RTN distributions.

The QHY600 cameras in operation at Las Cumbres Observatory have not violated these assumptions thus far, however, in substantially different operating environments such as those present in space-based missions, external factors such as radiation exposure might affect and degrade the RTN behavior of a CMOS sensor over time.

In Fig. 2 we show the per-pixel readout value histograms for select representative noisy pixels, following the same schema as in Fig. 1. We will refer to those examples by the Roman numbers (I-VI) as shown in the histograms.

Distributions I, II, and III show the expected trimodal distribution, but also more complex distributions are present as in examples IV-VI. In general, one can identify distributions that are (i) trimodal but asymmetric, (ii) multimodal with more than 3 peaks at different amplitudes, or (iii) very complex.

We model the readout distribution assuming one or three peaks, where each peak is described as a Gaussian distribution. For low-noise pixels, a single-Gaussian model will be the better fit. The simplified tri-modal model can easily be expanded for multiple RTN peaks by allowing for additional (symmetric) peaks in the future.

Our model forces all peaks of a pixel to have the same intrinsic width, representing the amplifier's read noise. The amplitudes of the two side peaks to the left and right are free parameters as this is an expected behavior for RTN, but the left/right separation from the central peak is forced to be symmetric according to the originating CDS mechanism. The probability distribution we use to model a pixel's readout, ignoring some scaling factors, is then written as:

$$\begin{aligned} P_{\text{single}}(x, x_0, \sigma) &= e^{-\frac{(x-x_0)^2}{2\sigma^2}} \\ P(x, A, A_\text{l}, A_\text{r}, x_0, \sigma, \delta) &= A \cdot \Big(P_{\text{single}}(x, x_0, \sigma) \\ &\quad + A_\text{l} \cdot P_{\text{single}}(x, x_0 + \delta, \sigma) \\ &\quad + A_\text{r} \cdot P_{\text{single}}(x, x_0 - \delta, \sigma) \Big) \end{aligned} \tag{1}$$

with:

- A Generic scaling factor.
- A_l, A_r Relative amplitude of the left and right side peaks relative to the center peak.
- σ The classical read noise, i.e., the width of the individual distribution.
- δ The separation from the center of the individual side lobe distributions.
- x_0 The center/expectation value of the distribution.

We use a simplified approach to model a pixel's readout distribution by binning the readout data into a histogram and then fit a function as in Eqn.1 using the python `scipy.optimize.curvefit`[8] function:

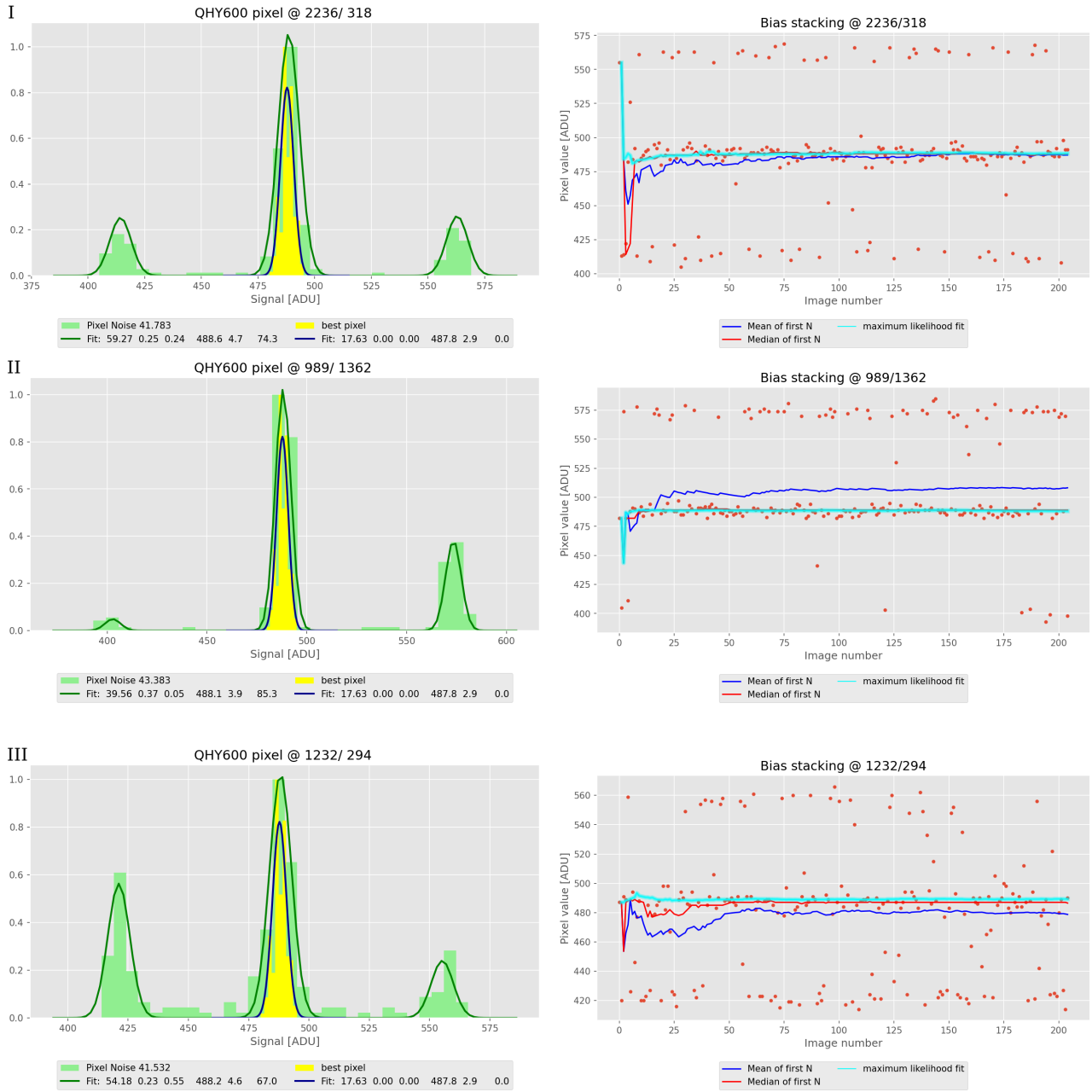

Figure 2: RTN readout distributions for exemplified pixels (left), and the same pixel's data presented as time series (right). The convergence rates of each method is represented in each time series plot.

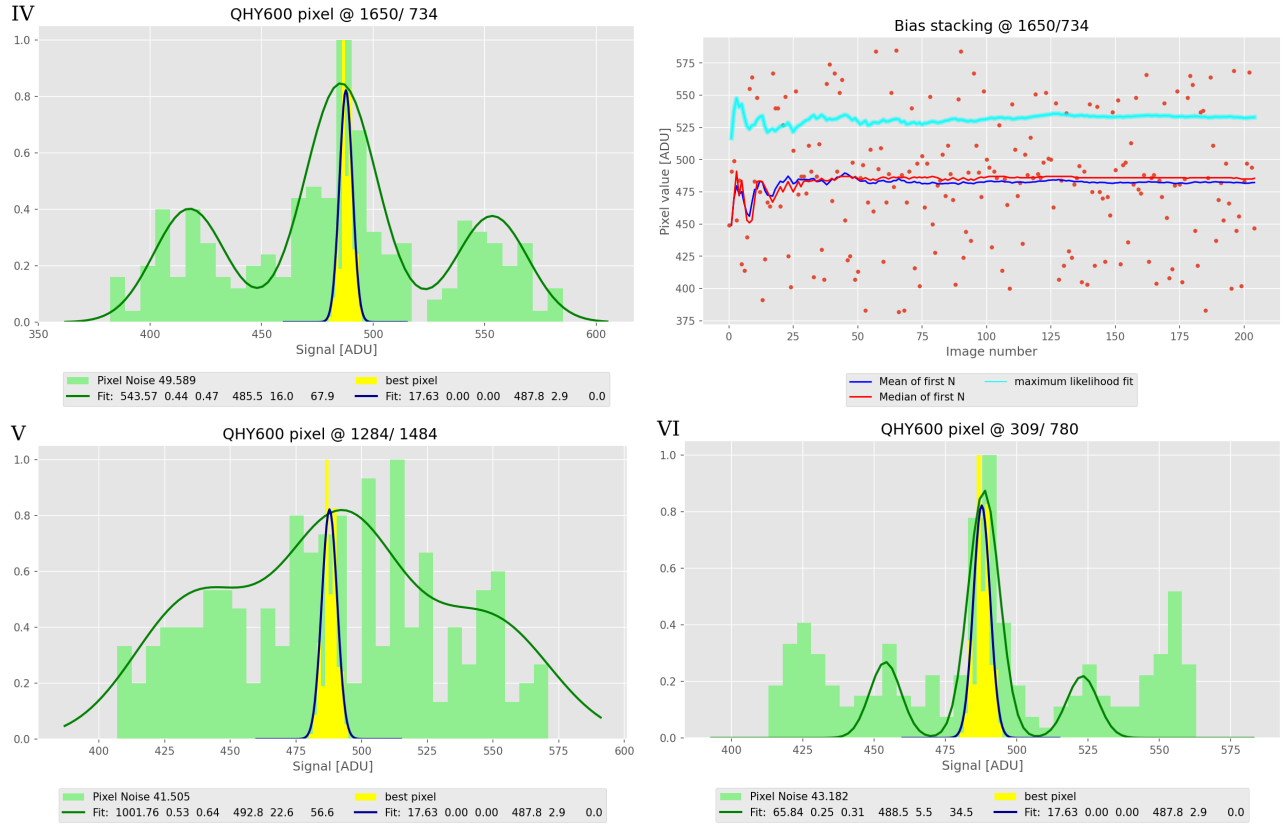

Figure 2: (continued) For the noisier pixel (example IV), the maximum likelihood method of estimating the pixel value does not converge to the expectation value, but performs very well in cases with well-defined multi-modal distributions. Examples V and VI show RTN readout distributions that are more complex than simple trimodal distributions.

```
popt_3, pcov_3 = curve_fit(np_triple_gaussian_function, xdata=
                                    binscenters, ydata=histogram, )
```

The fit function `np_triple_gaussian_function` is hardcoded as a three-gaussian distribution as in Eqn1 using python's `numpy`.[9] The data input is a histogram built from the individual pixel measurements, `bincenters` and `histogram`. In practice, we add constraints and starting points for the `curve_fit` function.

The fitting process can be further refined in the future: The model would ideally be fitted to the unbinned data using a maximum likelihood approach (e.g., `scipy.optimize` package), where the result from the prior binned data fit is used as a starting point. An alternative approach we have explored is to use Gaussian mixture model fitting (GMM), an algorithm frequently used in machine learning libraries. However, modeling with several independent Gaussian functions as implemented in GMM algorithms would ignore important constraints set by the detector's physics, in particular, the symmetry and equal width of RTN peaks. Tradeoffs of different fitting mechanisms remain to be explored. E.g., the number of peaks in the distribution can be easily expanded by successively fitting 1, 3, 5, or more Gaussians to the RTN distribution.

In the pixel distributions we overplot the best-fit trimodal model. For the "well-behaved" noisy pixels, Examples I-III, this fitting approach works very well and robustly. For more complex pixels, examples IV-VI, the model is too simple and inappropriate. A model distribution using 5 or 7 Gaussians would be more appropiate.

3.1 Forward Modelling

In this section, we will explore if the prior knowledge about the RTN distribution of a given pixel can be utilized in data processing, and we do so at the simplest example of stacking bias images. In Fig. 2, we plot the time series of the individual pixel readout values to the right of the example distributions. The histograms at the left, the model, and the time series, are based on 200 subsequent bias readouts.

How many bias images are needed for a stack in the presence of RTN to converge to the expectation value, and can we improve the converging behavior by incorporating the prior knowledge of a pixel's RTN distribution? The converging behavior is investigated by stacking only the first 1...N readouts of that sequence with the following algorithms:

1. the average,
2. the median, and
3. a maximum likelihood estimator (MLE) with the RTN model as the prior.

The results of the pixel stack are plotted as the blue, red, and cyan lines in the time series on the right in Fig 2, respectively.

The MLE is formulated as: Given the distribution of real measurements, and the prior knowledge of their distribution function Eqn. 1, parameterized by A_l, A_r, σ, δ, what is the most likely x_0 parameter of the distribution. In software this was implemented via the python `scipy.optimize.minimize` function:

```
def summed_ln_likelihood(x, A, Aleft, Aright, x0, sigma, delta):
    sum = np.sum(np.log(np_triple_gaussian_function(x, A, Aleft,
                                        Aright, x0, sigma, delta)))
    return sum

def findx0(xvec, A, Aleft, Aright, sigma, delta):
    def lnprob(paramters, xvec):
        x0 = paramters[0]
        return -1 * summed_ln_likelihood(xvec, A, Aleft, Aright, x0,
                                         sigma, delta)

    guess = [np.mean(xvec), ]
    res = scipy.optimize.minimize(lnprob, guess, xvec, )
    return res
```

The `findx0` function will fill in the derived scaling parameters from the result of the earlier calculated `popt_3` array. In a real-time processing scenario, the input for a pixel's distribution parameters would be read from a precomputed table, a super-calibration product comparable to a bad pixel mask.

For symmetric distributions (example I), all three stacking methods converge to the same value at the end of the 200 readouts. The asymmetric distributions (examples II and III) exemplify how a simple average stack will result in a wrong estimation from the pixel's expectation value and would lead to a systematic over/under subtraction of, e.g., dark current. In all cases, simple average stacking of a small number of images ($\leq$25) is deviant from the full sample's expectation value and can require as many as 50 to 100 readouts to converge within a few ADUs.

Median stacking approaches the expectation value with significantly fewer sample points than simple average stacking, usually within ≤ 25 readouts and most often faster, and is less prone to systematic errors in cases of an asymmetric RTN distribution. Median stacking should hence be preferred in a data processing system when combining small numbers of bias and dark frames from CMOS cameras.

In examples I through III, the maximum likelihood fit can converge significantly faster than median stacking, within less than 10 readouts, while average and median stacking can still produce larger deviations from the expectation value for a small subset of data. Upon visual inspection of such convergence plots of several hundred pixels, we confirm the trend of faster convergence for RTN pixels that model well with a trimodal distribution. However, depending on the actual realization of the initial random readout distribution in a series, the advantage might not be that clear in all cases.

Example IV illustrates how an insufficient model will cause the maximum likelihood fitting process to fail and result in a wrong result. A MLE can only succeed if the underlying model is appropriate.

MLE-based stacking is very computing intensive, and we estimate a fully modeled stack for each pixel of a QHY600 9kx6k array would take several hours on a single CPU system with standard Python code. When a significantly large number of images is available, median stacking would be the preferred method. When only a small number of exposures is available, such as for dark calibration images and science frames, using the maximum likelihood fitting approach has the potential to better mitigate the impact of RTN noise for affected pixels when deemed appropriate for the use case.

We summarize and add the following finer points:

1. We assumed that all pixels should read out the same bias level, and the central peak of an asymmetric RTN distribution should be the correct value. For such pixels, simple average stacking will create a biased result.

2. In the histogram's time series plots of the exemplified pixels, singular readout values fall in between the three peaks. We ascribe such readouts to the RTN state changing during the digitation process.

3. Some pixels do not model well with a tri-modal RTN distribution. A higher-order model might suffice in some cases, whereas some pixels are just too noisy to be described by a parameterized model, and might be best disregarded via a bad pixel map.

4. Advanced modeling of pixel readout values hinges on a robust way to create the pixel's RTN model — given the size of current and future CMOS arrays, manual verification of such models at scale is not feasible and poses a problem. At the same time, for certain applications, knowledge of a pixel's RTN behavior might be important, and more investment into software for more detailed RTN characterization might be warranted.

5. Stacking CMOS images via median combining instead of averaging will be more resilient against biases due to RTN, and for well-behaved RTN pixels, the maximum likelihood method will converge even faster. Non-linear stacking methods complicate error propagation significantly.

4. CMOS NOISE PROPAGATION IN THE BANZAI PIPELINE

We have demonstrated that the full characterization of each CMOS pixel w.r.t. RTN is in principle feasible but is complex and very computationally intensive. At the time of this writing, there is no astronomical software known to us that could utilize a full RTN description. A complete treatment of RTN in the data reduction process remains elusive and unreasonably expensive in practice for now; a more pragmatic solution was called for at Las Cumbres Observatory.

We have therefore established the following workflow: When commissioning a new QHY600/ CMOS camera, a per-pixel CMOS noise map is created as part of the commissioning process using a dedicated tool, `rts-maker`[§].[10] This map is based on the standard deviation as measured from a sequence of the order of 50 or more bias images. By design, the BANZAI pipeline fully propagates read noise and shot noise throughout data processing: when stacking bias and dark frames, but also when subtracting overscan, bias, and dark frames upon flat field division, and finally, adding the shot noise of the target. The resulting noise map is stored in an extra extension in the final FITS image. For CCD detectors, the noise map of an image is initially seeded with a single readnoise value governing all pixels. For CMOS detectors, this map is initialized with the rms noise for each individual pixel.

An example of a raw image, the initial CMOS noise map, and the resulting noise extension is shown in Fig.3.

A cornerstone of traditional CCD processing is the bad pixel map which is usually designed to mark or even interpolate pixels or columns with defects; this concept is built into the BANZAI workflow for indicating questionable pixels. However, we found that most noisy pixels will be masked as bad when using the typical rms-based tools to create such a mask. This led to undesirable behavior when CMOS data was calibrated by BANZAI, e.g., in the treatment of flat field scaling. We set bad pixel map for the CMOS detectors to constant "0", i.e., all pixels are masked as good. **A binary good/bad classification is insufficient for CMOS pixels,**

[§]https://doi.org/10.5281/zenodo.10933980

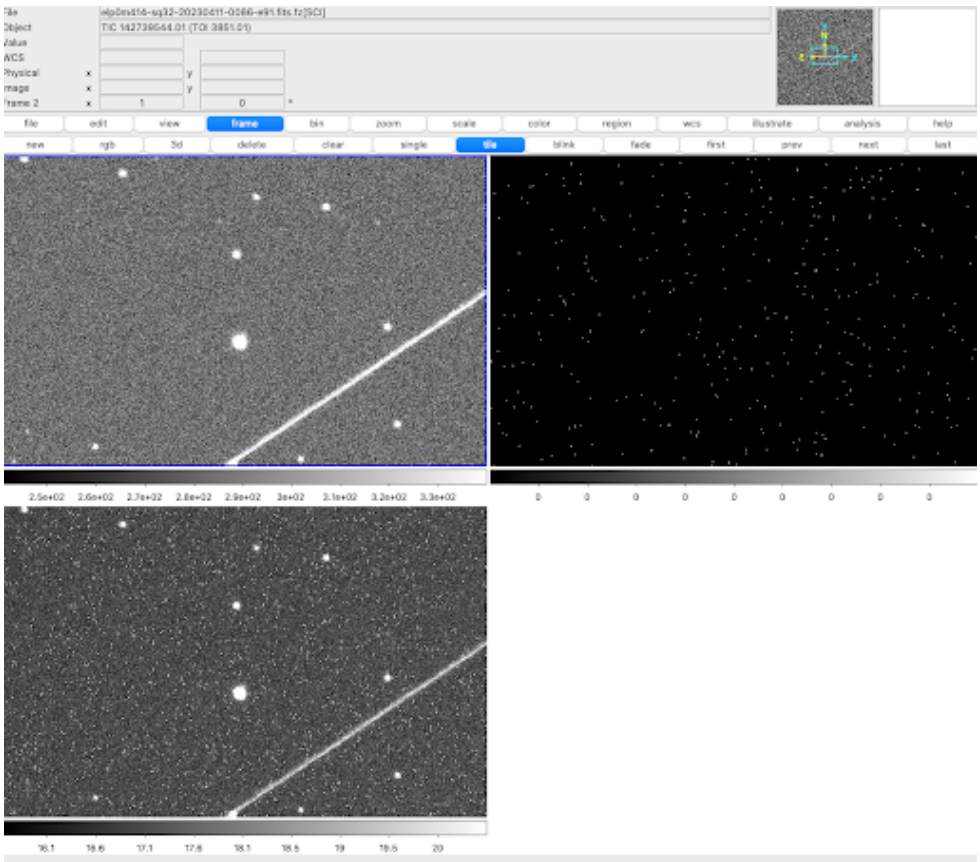

Figure 3: Propagation of the per-pixel CMOS noise in the BANZAI pipeline. Upper left: A BANZAI-processed image taken with a QHY600 camera. Upper right: For the same region of the detector, the CMOS noise mask is shown. Lower left: The propagated noise map (including shot noise) of the same region. The individual noisy pixels in the background are clearly identifiable.

and the information about pixel quality in processed Las Cumbres' CMOS data is encoded in the CMOS noise map.

BANZAI extracts sources from the images using the astropy-affiliated `photutils` package, and the propagated noisemap is used as the variance map for the source detection and photometry code, i.e., the object catalog derived by the BANZAI pipeline already utilizes the per-pixel noise.

Modeling the RTN as a rms noise is overly simplified and underestimates the noise systematics, however, it is a pragmatic approach with minimum change and impact on the existing CCD data processing that adds information about CMOS pixels right now.

Future additional data products associated with CMOS images could include a full per-pixel parameterization of the CMOS noise.

5. CONCLUSION & OUTLOOK

CMOS detectors have become a useful and cost-effective reality in astronomical instrumentation. Existing data pipelines developed for CCD processing can be immediately used for CMOS detectors with optional additional refinements. Per pixel-defined gain and read noise, and the addition of possible RTN, add complexity to the data processing workflow.

Per pixel noise propagation in the data processing system - approximated as rms noise - was identified as a first pragmatic step to acknowledge the different noise behavior of the QHY600's CMOS detectors in the BANZAI pipeline. This additional information can be readily utilized for subsequent data processing by the end user.

The IMX455 detectors in the QHY600 camera have a fairly large fraction of pixels affected by RTN. For most projects executed at Las Cumbres Observatory — both educational and scientific — this has not yet been identified as a limiting factor for high S/N observations. RTN affects pixels on a continuum, ranging from an effect of only a few electrons for most, and up to $\geq 80e^-$ pixels separation for the few worst pixels. However, with CMOS detectors being increasingly considered for wider application in astronomy, it remains to be seen how significant the impact of RTN will be for precision application in a low S/N regime, e.g., in UV small or cube satellite missions.

Not all CMOS sensor designs appear equally affected by RTN. Characterization of the RTN behavior should hence be an integral part of the detector selection, evaluation, and commissioning process for CMOS-based cameras.

When RTN cannot be mitigated through large S/N or stacking of a large number of pixels and is of concern for precision measurements, mitigation strategies can be developed. For the example of stacking bias images, we demonstrated that a full model of the RTN can lead to better stacking result with small samples of readouts. This result should be directly applicable when stacking dark calibration data and low-background science images. The concept of using the RTN noise model of a pixel as a prior could be extended for more advanced observing and data processing workflows: For single-object photometry, the prior knowledge of which pixels have bad RTN behavior could inform the positioning of a target on a detector. However, with

the potentially large number of RTN-affected pixels, the avoidance strategy quickly becomes less viable when comparison stars are to be observed as well.

Alarcon et al.[6] propose a spatial median filtering method to flag or interpolate RTN-affected pixels in single images. With the CMOS noise map, filtering can be limited to the affected pixels, reducing the time for processing. If the RTN distribution was known in advance, it could then be used as a prior when interpolating the noisy pixel, and using the information from that pixel's readout instead of discarding it.

In a similar manner, point spread function-based photometry could benefit from incorporating the likelihood distributions of a pixel's readout into the modeling. An RTN-affected pixel would be rejected in a PSF fit, but with the expected readout value known from the PSF model, one could decide in which of the three (or more) RTN states the pixel has been. The knowledge about a pixel's non-gaussian readout response can be utilized for a full error model, where the single read noise is replaced with a per-pixel error distribution.

In differential imaging applications, knowing in advance which pixels are blinking between RTN states could help to reduce false detections.

RTN in CMOS detectors — as seen in the QHY600 cameras — is a nuisance for the use cases at Las Cumbres Observatory. Still, the negative impact of RTN is by far outweighed by the benefits of cost and simplicity of those cameras. For precision low-signal applications, RTN can potentially be limiting and needs to be considered when designing an instrument with CMOS sensors.

Data Availability

Bias images used in this work are acquired as part of the observatory's calibration plan. Calibration data do not have a proprietary period at Las Cumbres Observatory and can be downloaded from the data archive web page `archive.lco.global`.

Acknowledgments

This paper is based on observations made with the Las Cumbres Observatory's education network telescopes that were upgraded through generous support from the Gordon and Betty Moore Foundation.

REFERENCES

[1] Brown, T. M., Baliber, N., Bianco, F. B., Bowman, M., Burleson, B., Conway, P., Crellin, M., Depagne, E., Vera, J. D., Dilday, B., Dragomir, D., Dubberley, M., Eastman, J. D., Elphick, M., Falarski, M., Foale, S., Ford, M., Fulton, B. J., Garza, J., Gomez, E. L., Graham, M., Greene, R., Haldeman, B., Hawkins, E., Haworth, B., Haynes, R., Hidas, M., Hjelstrom, A. E., Howell, D. A., Hygelund, J., Lister, T. A., Lobdill, R., Martinez, J., Mullins, D. S., Norbury, M., Parrent, J., Paulson, R., Petry, D. L., Pickles, A., Posner, V., Rosing, W. E., Ross, R., Sand, D. J., Saunders, E. S., Shobbrook, J., Shporer, A., Street, R. A., Thomas, D., Tsapras, Y., Tufts, J. R., Valenti, S., Horst, K. V., Walker, Z., White, G., and Willis, M., "Las Cumbres Observatory Global Telescope Network," *Publications of the Astronomical Society of the Pacific* **125**, 1031 (Sept. 2013). Publisher: IOP Publishing.

[2] Harbeck, D.-R., Taylor, B., Kirby, A., Bowman, M., Foale, S., McCully, C., Daily, M., DeVera, J., Douglass, D., Willis, M., Volgenau, N., Conway, P., Haworth, B., Estrada, J., Seale, S., Hopkinson, A., Rios, F., Kotapali, P., and Rosing, W., "An upgraded 0.4-meter telescope fleet for Las Cumbres Observatory's Educational and Science Programs," **13096**, 131 (2024).

[3] McCully, C., Volgenau, N. H., Harbeck, D.-R., Lister, T. A., Saunders, E. S., Turner, M. L., Siiverd, R. J., and Bowman, M., "Real-time processing of the imaging data from the network of Las Cumbres Observatory Telescopes using BANZAI," in [*Software and Cyberinfrastructure for Astronomy V*], Guzman, J. C. and Ibsen, J., eds., *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **10707**, 107070K (July 2018). _eprint: 1811.04163.

[4] Wang, X., Rao, P. R., Mierop, A., and Theuwissen, A. J., "Random Telegraph Signal in CMOS Image Sensor Pixels," in [*2006 International Electron Devices Meeting*], 1–4 (Dec. 2006). ISSN: 2156-017X.

[5] Ishida, H., Kagawa, K., Komuro, T., Zhang, B., Seo, M.-W., Takasawa, T., Yasutomi, K., and Kawahito, S., "Multi-Aperture-Based Probabilistic Noise Reduction of Random Telegraph Signal Noise and Photon Shot Noise in Semi-Photon-Counting Complementary-Metal-Oxide-Semiconductor Image Sensor," *Sensors (Basel, Switzerland)* **18**, 977 (Mar. 2018).

[6] Alarcon, M. R., Licandro, J., Serra-Ricart, M., Joven, E., Gaitan, V., and de Sousa, R., "Scientific CMOS Sensors in Astronomy: IMX455 and IMX411," *Publications of the Astronomical Society of the Pacific* **135**, 055001 (May 2023). ADS Bibcode: 2023PASP..135e5001A.

[7] Hoadley, K., McCully, C., Kyne, G., Cruz Aguirre, F., Andrews, M., Basset, C., Bostroem, K. A., Brown, P. J., Davis, G., Hamden, E. T., Harbeck, D., Hennessy, J., Hoenk, M., Hosseinzadeh, G., Howell, D. A., Jewell, A., Jha, S., Li, J., Milne, P., Moustakas, L., Nikzad, S., Pellegrino, C., Polin, A., Sand, D. J., Shen, K. J., and Storrie-Lombardi, L., "The Ultraviolet Type Ia Supernova CubeSat (UVIa): Science Motivation & Mission Concept," (Feb. 2025). ADS Bibcode: 2025arXiv250211957H.

[8] Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, , Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., and van Mulbregt, P., "SciPy 1.0: fundamental algorithms for scientific computing in Python," *Nature Methods* **17**, 261–272 (Mar. 2020). Publisher: Nature Publishing Group.

[9] Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E., "Array programming with NumPy," *Nature* **585**, 357–362 (Sept. 2020). Publisher: Nature Publishing Group.

[10] PK0207 and Daily, M., "LCOGT/cmos-noise-map: Production Snapshot," (Apr. 2024).