Exposure Time Calculator
Theory, Functionality & User Manual

Version 1.00

Ideal Sub-Exposure Time for Astrophotography

> **Origins:** This software implements the methodology from the original **Excel spreadsheet** created by **Benoit Saintot**. The Python application provides a cross-platform graphical interface while preserving the same theoretical foundations and calculation methods.

Contents

1 Introduction

Deep-sky astrophotography relies on stacking many sub-exposures to improve signal-to-noise ratio. A fundamental question every imager faces is: *How long should each sub-exposure be?* Too short, and read noise degrades the image. Too long, and you gain little while risking saturation, trailing, or wasted time on lost frames. The Exposure Time Calculator addresses this by combining sky background, sensor characteristics, and two complementary mathematical approaches to recommend optimal sub-exposure times per filter.

The application uses two complementary approaches:

- **Swamp Factor** — target a sky-to-read-noise ratio (typically 3–10)
- **Optimal Time** — accept a small amount of extra noise (%) and compute the recommended exposure per filter

2 Objectives for the Astrophotographer

2.1 What the imager is trying to achieve

The astrophotographer's goal is to maximise the **signal-to-noise ratio (SNR)** in the final stacked image while making efficient use of time and equipment. Concretely:

- **Reveal faint details** — galaxies, nebulae, and dust lanes are buried in noise. Higher SNR makes faint structures visible and reduces grain.

- **Minimise noise** — Each pixel contains useful signal plus several noise sources. The aim is to minimise the relative contribution of noise so the signal stands out.

- **Optimise integration time** — For a given total imaging time, choosing the right sub-exposure length can improve the final SNR or allow shorter total sessions.

2.2 Why sub-exposure length matters

Read noise is *fixed* per exposure: it is added once per frame regardless of exposure time. Sky background and dark current, however, accumulate with time. Thus:

- **Short exposures** — More frames for the same total time, but each frame is dominated by read noise. Stacking helps, but you start from noisier subs.

- **Long exposures** — Fewer frames, but each frame has more sky and dark signal relative to read noise. Beyond a certain point, extra length brings little gain (diminishing returns).

- **Sweet spot** — There is an exposure length that balances read noise against sky/dark in an optimal way. The calculator finds this for your conditions and gear.

2.3 Practical constraints and trade-offs

> **Important**
>
> **Guiding & tracking** — Longer subs require better tracking. If your mount drifts or guiding fails, a long sub is wasted. Shorter subs can be more robust in difficult conditions.
> **Saturation** — Bright stars or cores can saturate pixels. Very long exposures increase this risk; shorter subs may be necessary in high-contrast regions.
> **Light pollution & filters** — Sky background (e^-/px/s) varies with filter: L and RGB are higher; narrowband (Ha, Oiii, Sii) is lower. Narrowband often needs longer subs to reach the optimal zone.
> **Efficiency** — Aim for the shortest exposure that still keeps you in the optimal noise regime. This maximises the number of usable subs and can reduce the impact of clouds or equipment issues.

2.4 Summary of objectives

The Exposure Time Calculator helps you:

1. **Determine target sky background median (ADU)** — So you can measure a raw frame and adjust exposure until you hit the right level.

2. **Obtain recommended exposure times** — Per filter (L, RGB, NB), based on your accepted additional noise tolerance.

3. **Compare strategies** — Evaluate, for example, $120\,\text{s}$ vs $180\,\text{s}$ subs and see the impact on noise.

4. **Use your camera effectively** — The built-in database fills sensor parameters automatically for 40+ cameras.

3 Theory

3.1 Noise Sources

In each pixel, the measured value (in electrons before digitisation) is the sum of:

$$\text{Total} = \underbrace{\text{Useful signal}}_{\text{target}} + \underbrace{\text{Sky background}}_{\text{unwanted}} + \underbrace{\text{Read Noise}}_{\text{RN}} + \underbrace{\text{Dark Current}}_{\text{DC}} + \underbrace{\text{Photon noise}}_{\text{Poisson}} \quad (1)$$

Useful signal — Photons from the target (galaxy, nebula, star). This is what we want to maximise relative to noise.

Sky background — Light pollution, airglow, moonlight, scattered light. Unwanted but unavoidable; it accumulates linearly with exposure time (e^-/pixel/s). Varies strongly with filter: L and RGB are highest; narrowband filters (Ha, Oiii, Sii) dramatically reduce sky glow.

Read noise (RN) — Electronic noise introduced by the sensor and amplifier when the pixel is read out. Expressed in electrons rms. It is *constant per exposure*: you pay it once per frame, regardless of exposure length. Depends on gain: higher gain usually lowers RN but reduces full-well capacity.

Dark current (DC) — Thermally generated electrons in the sensor. Increases with sensor temperature. Expressed in e^-/s/pixel. Cooled cameras reduce DC significantly. Accumulates linearly with time.

Photon noise — Fundamental quantum noise: the arrival of photons is Poissonian, so the uncertainty is $\sqrt{N}$ for N photons. Inherent to the signal and sky; cannot be eliminated.

3.2 Total Noise

> **Theory**
>
> The total noise (standard deviation) in electrons per pixel is:
>
> $$\sigma = \sqrt{\mathrm{RN}^2 + \mathrm{DC} \cdot t + \mathrm{Sky} \cdot t} \tag{2}$$
>
> where t is the exposure time in seconds, RN is read noise (e$^-$ rms), DC is dark current (e$^-$/s/pixel), and Sky is sky background (e$^-$/pixel/s).

Interpretation: The variance σ^2 is the sum of independent noise variances. Read noise contributes a fixed RN^2 per frame. Dark and sky add in proportion to time (they accumulate). For short t, RN dominates. For long t, Sky·t (and DC·t if significant) dominate. The transition occurs when $\mathrm{Sky} \cdot t \approx \mathrm{RN}^2$ — this is the essence of the Swamp Factor.

3.3 Signal-to-Noise Ratio (SNR) When Stacking

When stacking N identical sub-exposures of exposure time t:

$$\mathrm{SNR}_{\mathrm{stack}} = \frac{\mathrm{Signal} \times \sqrt{N}}{\sigma} \tag{3}$$

The signal increases linearly with total integration time ($N \cdot t$); the noise in the stack is $\sigma/\sqrt{N}$ because independent noise adds in quadrature. Thus SNR grows as $\sqrt{N}$, or equivalently as $\sqrt{\text{total time}}$ — doubling integration time improves SNR by $\sqrt{2} \approx 1.41$.

Practical consequence: For a fixed total time, using more (shorter) subs or fewer (longer) subs changes σ per sub. If subs are too short, σ is large (read-noise dominated) and you pay a penalty. If subs are long enough that sky dominates, further lengthening brings little gain. The calculator finds the exposure that optimises this balance.

3.4 Swamp Factor (First Approach)

> **Theory**
>
> The **Swamp Factor** (SF) is the ratio of sky background signal to read noise variance:
>
> $$\mathrm{SF} = \frac{\mathrm{Sky} \times t}{\mathrm{RN}^2} \tag{4}$$
>
> It answers: "How many times does the sky signal exceed the read noise variance?" When SF is large, sky "swamps" read noise; when SF is small, read noise dominates.
>
> - **SF < 3:** Read noise dominates. Each sub is noisy; you are wasting integration time. Increase exposure time.
> - **3 ≤ SF ≤ 10:** Optimal zone. Sky and read noise are balanced; you get good efficiency.
> - **SF > 10:** Sky dominates. Little gain from longer exposures; you can reduce if guiding or saturation are concerns.

3.4.1 Target Median (ADU)

The calculator converts SF into a target median ADU so you can measure a raw frame. To achieve a given SF, the target sky background median in ADU is:

$$\text{Median}_{\text{target}} = \text{INT}\left(\frac{\text{SF} \times \text{RN}^2}{\text{Gain}_{e/\text{ADU}}} + \text{Offset}\right) \times \frac{2^{16}}{2^{\text{Bits}}} \tag{5}$$

On a raw frame, select a sky region without stars, measure the median (e.g. Process Statistics in PixInsight), and compare to this target. Adjust exposure time so the median falls in the optimal band (SF×3 to SF×10).

3.5 Additional Noise (Second Approach)

This approach answers: "If I use sub-exposures of length T instead of one infinitely long exposure, how much extra noise do I accept?" In practice, we never take infinite exposures, but the theoretical limit gives a reference. Accepting a small extra noise (e.g. 5%) lets you use shorter subs — useful when guiding, clouds, or saturation limit exposure length.

Compared to an infinitely long single exposure, a finite sub-exposure adds extra noise. The **additional noise** (as a fraction) is:

$$\text{Add}_{\text{noise}} = \sqrt{\frac{(T \cdot \text{Sky}/\text{RN}^2) + 1}{(T \cdot \text{Sky}/\text{RN}^2)}} - 1 \tag{6}$$

where T is the exposure time in seconds.

> **Theory**
>
> The **C factor** relates accepted additional noise (%) to exposure time:
>
> $$C = \frac{1}{\left(1 + \frac{\%\text{noise}}{100}\right)^2 - 1} \tag{7}$$
>
> Typical values: 1% ⇒ C≈50   2% ⇒ C≈25   5% ⇒ C≈10   10% ⇒ C≈4

The **optimal exposure time** for a given C is:

$$T_{\text{optimal}} = \text{ceil}\left(C \times \frac{\text{RN}^2}{\text{Sky}}\right) \tag{8}$$

Higher sky background shortens T_{optimal} (L/RGB in light pollution); lower sky lengthens it (narrowband, dark sites). The C factor encodes your tolerance: 5% extra noise ⇒ C≈10; 1% ⇒ C≈50 (longer subs required).

3.6 Combining Both Approaches

> **Important**
>
> 1. Compute T_{optimal} from the second approach (accepted noise %)
> 2. Check that SF is between 3 and 10 at that exposure
> 3. If SF < 3 ⇒ increase exposure; if SF > 10 ⇒ can reduce

4 Application Functionality

4.1 Parameters Tab

Sky background levels (e^-/pixel/s):
Luminance (L), RGB, Narrowband 12 nm, 7 nm. Narrowband 3 nm is computed automatically as NB12/4. Obtain values from SharpCap: https://tools.sharpcap.co.uk/. Use max Qe in the linear range, not the absolute peak.

Sensor parameters:

- **Read Noise (e^- rms)** — depends on gain
- **Gain (e^-/ADU)** — electrons per ADU, not manufacturer gain index
- **Dark Current (e^-/s/px)** — depends on cooling temperature
- **Bits** — ADC depth (12, 14, 16)
- **Offset (ADU)**

Camera database: Select brand (ZWO, QHY, Moravian, ATIK, Player One, ToupTek), model, gain preset, and cooling temperature. Click **Apply** to auto-fill sensor parameters. 40+ camera models supported.

4.2 Swamp Factor Tab (1st Approach)

- Set desired Swamp Factor (3–10)
- Read target medians: SF×3, SF×N, SF×10
- On a raw frame, measure median in a sky region; compare to targets and adjust exposure

4.3 Optimal Time Tab (2nd Approach)

- Set accepted additional noise (%) — C factor is computed
- Read recommended sub-exposure time per filter (L, RGB, NB 12/7/3 nm)
- For any exposure time you enter, the app shows the corresponding additional noise (%) — compare to your threshold

4.4 Comparison Tab

Compare two exposure strategies (e.g. 120 s vs 180 s) in Luminance and RGB. Displays additional noise for each strategy and the Delta SNR between them.

4.5 Charts Tab

- **L / RGB** — Additional noise (%) vs exposure time for L and RGB
- **Narrowband** — Same for NB 12 nm, 7 nm, 3 nm

The accepted threshold line is shown. Exposure times below the line are within your tolerance.

4.6 Tables Tab

Tables of additional noise (%) for fixed exposure times (4–2000 s for L/RGB, 100–2000 s for NB). Green values are below your threshold.

4.7 Export

Export current parameters and results to JSON: parameters (sky, sensor), approach 1 (SF, medians), approach 2 (C factor, optimal times per filter).

5 Installation & Running

Windows: Double-click `run.bat`. If Python is missing, the script installs it via winget. If winget is missing, you are guided to install it.

Linux/macOS: Run `./run.sh`. Python and tkinter are installed via the system package manager if missing.

The application installs `matplotlib` automatically on first run.

6 Credits

Original Excel spreadsheet & theory: © Benoit SAINTOT
Python GUI application: NGC4565
Version: 1.00
License: Use and share with attribution.