Analysis of human visual experience data

Authors

Johannes Zauner^[1, 2] [0000-0003-2171-4566]

Aaron Nicholls [3] [0009-0001-6683-6826]

Lisa A. Ostrin [4] [0000-0001-8675-1478]

Manuel Spitschan[1,2,5] [0000-0002-8572-9268]

^ corresponding author

Affiliations

1 Technical University of Munich, TUM School of Medicine and Health, Department Health and Sport Sciences, Chronobiology & Health, Munich, Germany

2 Translational Sensory & Circadian Neuroscience, Max Planck Institute for Biological Cybernetics, Tübingen, Germany

3 Reality Labs Research, USA

4 University of Houston College of Optometry, Houston, Texas, USA

5 TUM Institute for Advanced Study (TUM-IAS), Technical University of Munich, Garching, Germany

Abstract

Exposure to the optical environment — often referred to as *visual experience* — profoundly influences human physiology and behavior across multiple time scales. In controlled laboratory settings, stimuli can be held constant or manipulated parametrically. However, such exposures rarely replicate real-world conditions, which are inherently complex and dynamic, generating high-dimensional datasets that demand rigorous and flexible analysis strategies. This tutorial presents an analysis pipeline for visual experience datasets, with a focus on reproducible workflows for human chronobiology and myopia research. Light exposure and its retinal encoding affect human physiology and behavior across multiple time scales. Here we provide step-by-step instructions for importing, visualizing, and processing viewing distance and light exposure data. This includes time-series analyses for working distance, biologically relevant light metrics, and spectral characteristics. The tasks are standardized through the open-source R package LightLogR. By leveraging a modular approach, the tutorial supports researchers in building flexible and robust pipelines that accommodate diverse experimental paradigms and measurement systems.

Keywords: wearable, light logging, viewing-distance, near-work, visual experience, metrics, circadian, myopia, risk factors, time series, spectral analysis, spatial analysis, open-source, reproducibility

Introduction

Exposure to the optical environment — often referred to as *visual experience* — profoundly influences human physiology and behavior across multiple time scales. Two notable examples, from distinct research domains, can be understood through a common retinally-referenced framework.

The first example relates to the non-visual effects of light on human circadian and neuroendocrine physiology. The light–dark cycle entrains the circadian clock, and light exposure at night suppresses melatonin production (Brown et al. 2022; Blume, Garbazza, and Spitschan 2019). The second example concerns the influence of visual experience on ocular development, particularly myopia. Time spent outdoors — which features distinct optical environments — has been consistently associated with protective effects on ocular growth and health outcomes (Dahlmann-Noor et al. 2025).

In controlled laboratory settings, light exposure can be held constant or manipulated parametrically. In contrast, real-world conditions are inherently complex and dynamic, and cannot be captured by single spot measurements. As people move in and between spaces (indoors and outdoors) and move their body, head, and eyes, exposure to the optical environment varies significantly (Webler et al. 2019) and is modulated by behavior (Biller, Balakrishnan, and Spitschan 2024). Wearable devices for measuring light exposure have thus emerged as vital tools to capture the ecological visual experience. These tools generate high-dimensional datasets that demand rigorous and flexible analysis strategies.

Starting in the 1980s (Okudaira, Kripke, and Webster 1983), technology to measure optical exposure has matured, with miniaturized illuminance sensors now (in 2025) very common in consumer wearables (van Duijnhoven et al. 2025). In research, several devices are available that differ in functionality, ranging from small pins measuring ambient illuminance (Mohamed et al. 2021) to head-mounted multi-modal devices capturing nearly all relevant aspects of visual experience (Gibaldi et al. 2024). Increased capabilities in wearables bring complex, dense datasets. These go hand-in-hand with a proliferation of metrics, as highlighted by recent review papers in

both circadian and myopia research (Hönekopp and Weigelt 2023; Hartmeyer and Andersen 2023).

At present, the analysis processes to derive metrics are often implemented on a per-laboratory or even per-researcher basis. This fragmentation is a potential source of errors and inconsistencies between studies, consumes considerable researcher time (Hartmeyer, Webler, and Andersen 2022), and these bespoke processes and formats hinder harmonization or meta-analysis across multiple studies. It is very common that more time is spent preparing data than gaining insights through rigorous statistical analysis. These preparation tasks are best handled, or at least facilitated, by standardized, transparent, community-based analysis pipelines (Zauner, Udovicic, and Spitschan 2024).

In circadian research, the R package LightLogR was developed to address this need (Zauner, Hartmeyer, and Spitschan 2025). LightLogR is an open-source, MIT-licensed, community-driven package specifically designed for data from wearable light loggers and optical radiation dosimeters. It contains functions to calculate over sixty different metrics used in the field (Hartmeyer and Andersen 2023). The package functions come with light-related defaults, but they remain fundamentally agnostic to modality. As a result, parameters like viewing distance and light spectra, both highly relevant to myopia research (Hönekopp and Weigelt 2023), can easily be handled.

In this article, we demonstrate that LightLogR's analysis pipelines and metric functions apply broadly across the field of visual experience research, not just to circadian rhythms and chronobiology. Our approach is modular and extensible, allowing researchers to adapt it to a variety of devices and research questions. Emphasis is placed on clarity, transparency, and reproducibility, aligning with best practices in scientific computing and open science. We use example data from two devices (worn by different individuals and at different times) to showcase the LightLogR workflow with metrics relevant to myopia research, covering working distance, (day)light exposure, and spectral analysis. Readers are encouraged to recreate the analysis using the provided code. All necessary data and code are openly available in the GitHub repository.

Scope

This article focuses on workflows for deriving condensed metrics from time-series data collected with wearable devices in the visual-experience domain. Specifically, we address illuminance, viewing distance, and spectral irradiance. Example datasets from two types of wearable devices are used for illustration.

Many relevant considerations arise when collecting data with wearable devices. This article covers only a subset of these. In particular, it does not address:

- device selection (see, e.g., van Duijnhoven et al. 2025; Zauner et al. 2025a)
- measurement accuracy or device calibration
- auxiliary data such as sleep/wake information (see, e.g., Zauner et al. 2025b; Guidolin et al. 2024)

More information on those aspects can be found in the *Technical guide for wearable optical radiation dosimetry and visual experience assessment* (Zauner et al. 2025c).

To demonstrate the workflows, this article uses expert-informed definitions of metrics and metric parameters (see, e.g., **Table 1**, **Table 2**, and the non-wear detection rules based on activity data described in Supplement 1). These definitions and thresholds should not be interpreted as universal standards, nor are they hard-coded into the software package. For any application, parameter choices must be tailored to the research domain, study context and design, and the specifications of the wearable device.

Further, the article is split up in the main analysis part, where all metrics are calculated, and the Supplement 1, where data is imported, screened, and prepared. Thus, the reader is referred to Supplement 1 for all aspects regarding data formats, preparation steps, and handling of gaps, i.e., missing data.

Lastly, the example data used in the article do not stem from a controlled experimental data collection but consist of pilot data gathered in an ecological setting without a fixed protocol. Given the substantial interindividual differences in visual experience metrics, and because the analyses focus on one participant at a time, the reported results should be interpreted as illustrative rather than representative of typical or population-level values.

Method and Materials

Software

This tutorial was built with Quarto, an open-source scientific and technical publishing system that integrates text, code, and code output into a single document. The source code to reproduce all results is included and accessible via the Quarto document's code tool menu. All analyses were conducted in *R* (version 4.5.0, "How About a Twenty-Six") using *LightLogR* (version 0.10.0 "High noon"). We also used the *tidyverse* suite (version 2.0.0) for data manipulation (which LightLogR follows in its design), and the *gt* package (version 1.1.0) for generating summary tables. A comprehensive overview of the R computing environment is provided in the session info (see Session info section of the online tutorial).

Metric selection and definitions

In March 2025, two workshops with myopia researchers — initiated by the Research Data Alliance (RDA) Working Group on Optical Radiation Exposure and Visual Experience Data — focused on current needs and future opportunities in data analysis, including the development and standardization of metrics. Based on expert input from these workshops, the authors of this tutorial compiled a list of visual experience metrics, shown in **Table 1.** These include many currently used metrics and definitions (Wen et al. 2020, 2019; Bhandari and Ostrin 2020; Williams et al. 2019), as well as new metrics enabled by spectrally-resolved measurements. While they are not derived by a formal consensus process, they are expert-informed and used in current scientific research, and thus will serve as example-definitions for metrics and thresholds throughout this article.

No.	**Name**	**Implementation**[1]
Distance		
1	Total wear time daily	durations()*
2	Duration of Near work, Intermediate Work, Near + Intermediate Work, or per each *Distance range*	filter for distance range + durations()* (for single ranges) or grouping by distance range + durations()* (for all ranges)

	(10cm steps)	
3	Frequency of Continuous near work	extract_clusters()* + summarize_numeric()*
4	Frequency, duration, and distances of *Near Work episodes*	extract_clusters()* + extract_metric()* + summarize_numeric()*
5	Frequency and duration of *Visual breaks*	extract_clusters()* + filter
Light		
6	Light exposure (in lux)	summarize_numeric()*
7	Duration per *Outdoor range*	grouping by Outdoor range + durations()*
8	The number of times light level changes from indoor (<1000 lx) to outdoor (>1000 lx)	extract_states()* + summarize_numeric()*
9	Longest period above 1000 lx	period_above_threshold()*
Spectrum		
10	Ratio of short vs. long wavelength light	spectral_integration()* + summarize_numeric()*
11	Melanopic daylight efficacy ratio (MDER)	spectral_integration()* + summarize_numeric()*
12	Short-wavelength light at certain times of day	spectral_integration()* + filter_Time()* (for defined times) or cut_Datetime()* (for regular time intervals) or add_photoperiod()* (for solar times) + grouping by time state + summarize_numeric()*

[1] Functions from `LightLogR` are presented with an *. Non-asterisk functions refer to tidyverse functions.

Table 1: Overview of metrics. In all cases, the averages for weekday, weekend, and the mean daily value are calculated. Definitions are shown in ***Table 2****.*

Table 2 provides definitions for the terms used in **Table 1.** Note that specific definitions may vary depending on the research question or device capabilities.

Term / metric	Description / pseudo formula
Total wear time	$\sum(t) * dt$, where t: valid observations
Mean daily	$\frac{5 * \text{weekday} + 2 * \text{weekend}}{7}$
Near work	working distance, $[15,60)cm$
Intermediate Work	working distance, $[60,100)cm$
Total work	working distance, $[10,120)cm$
Distance range	working distance, $[15,20)cm$, Extremely near $[20,30)cm$, Very near $[30,40)cm$, Fairly near $[40,50)cm$, Near $[50,60)cm$, Moderately near $[60,70)cm$, Near intermediate $[70,80)cm$, Intermediate $[80,90)cm$, Moderately intermediate $[90,100)cm$, Far intermediate
Continuous near work	$T_{\text{t}} = \sum(t) * dt$, where working distance, $[20,60)cm$, $T_{\text{t}} \geq 30minutes$, $T_{interruptions} \leq 1minute$
Near work episodes	$T_{\text{t}} = \sum(t) * dt$, where working distance, $[20,60)cm$, $T_{\text{interruptions}} \leq 20seconds$
Ratio of daily near work	$\frac{T_{\text{near work}}}{T_{\text{total wear}}}$
Visual break	$T_{\text{t}} = \sum(t) * dt$, where working distance $\geq 100cm$, $T_{\text{t}} \geq 20$ seconds, $T_{t-1\ (\text{working distance}<100\,cm)} \leq 20$ minutes
Outdoor range	illuminance, $[1000,2000)lx$, Outdoor bright $[2000,3000)lx$, Outdoor very bright $[3000,\infty)lx$, Outdoor extremely bright
Light exposure[1]	$Illuminance$
Spectral bands	spectral irradiance, $[400,500]nm$, short wavelength light $[600,700]nm$, long wavelength light
Ratio of short vs. long wavelength light	$\frac{E_{e,\text{short wavelength}}}{E_{e,\text{long wavelength}}}$

[1] This deviates from the common definition of luminous exposure, which is the sum of illuminance measurements scaled to hourly observation intervals

Table 2: Definitions of terms and metrics

It should be noted that although daylight levels can far exceed the thresholds defined in **Table 2** - and may reach or even exceed 10^5 lux - empirical daylight levels measured at eye level are much lower, typically around 10^3 lux, especially when considering aggregated time-series data over minutes to hours.

Devices

Data from two wearable devices are used in this analysis:

- `Clouclip`: A wearable device that measures viewing distance and ambient light simultaneously [Glasson Technology Co., Ltd, Hangzhou, China; Wen et al. (2021); Wen et al. (2020)]. The Clouclip provides a simple data output with only distance (working distance, in centimeters) and illuminance (ambient light, in lux). Data in our example were recorded at 5-second intervals. Approximately one week of data (~120,960 observations) is about 1.6 MB in size.
- `Visual Environment Evaluation Tool` (VEET): A head-mounted multi-modal device that logs multiple data streams [Reality Labs Research, Menlo Park, CA, USA; Sah, Narra, and Ostrin (2025); Sullivan et al. (2024)]. The VEET dataset used here contains simultaneous measurements of distance (via a time-of-flight sensor), ambient light (illuminance), activity (accelerometer & gyroscope), and spectral irradiance (multi-channel light sensor). Data were recorded at 2-second intervals, yielding a very dense dataset (~270 MB per week).

Data processing summary

The Results section uses imported and pre-processed data from the two devices to calculate metrics. Supplement 1 contains the annotated code and description for the steps involved, which are summarized as follows, and as shown in Figure 1. Please refer to the supplement for details.

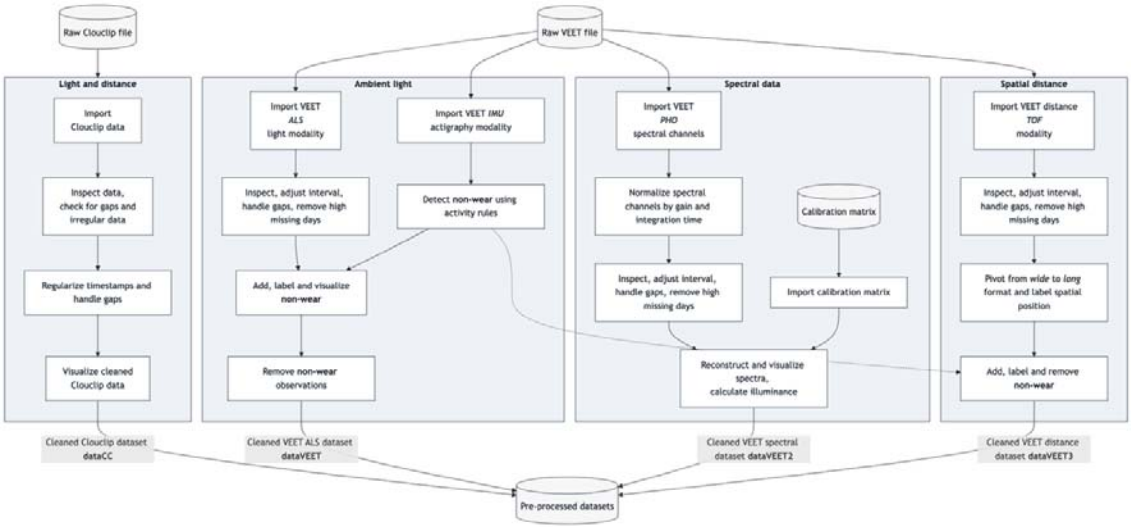


Figure 1: Pre-processing steps in the supplement document

Data import: We imported raw data from the `Clouclip` and `VEET` devices using LightLogR's built-in import functions, which automatically handle device-specific formats and idiosyncrasies.

The `Clouclip` export file (provided as a tab-delimited text file) contains timestamped records of distance (cm) and illuminance (lux). LightLogR's `import$Clouclip` function reads this file, after specifying the device's recording timezone, and converts device-specific sentinel codes into proper missing values. For instance, the Clouclip uses special numeric codes to indicate when it is in "sleep mode" or when a reading is out of the sensor's range, rather than recording a normal value. LightLogR identifies `-1` (for both distance and lux) as indicating the device's sleep mode and `204` (for distance) as indicating the object was beyond the measurable range, replacing these with `NA` and logging their status in separate columns. The import routine also provides an initial summary of the dataset, including start and end times and any irregular sampling intervals or gaps.

For the `VEET` device, data were provided as CSV logs (zipped on Github, due to size). We focused on the ambient light sensor modality first. Using `import$VEET(..., modality = "ALS")`, we extracted the illuminance (`Lux`) data stream and its timestamps. The raw VEET data similarly contains irregular intervals and can contain missing periods (e.g., if the device stopped recording or was reset); the import summary flags these issues.

Besides the `Clouclip` and `VEET`, LightLogR 0.10.0 contains import functions for 18 more wearable devices. The package further supports versions due to evolving data formats, and includes documentation for both code-based and code-less additions of new device import-functions.

Irregular intervals, gaps, non-wear times: Both datasets showed irregular timing and missing data, i.e., gaps. Irregular data means that some observations did not align to the nominal sampling interval (e.g., slight timing drift or pauses in recording). For the `Clouclip` 5-second data, we detected irregular timestamps spanning all but the first and last day of the recording. Handling such irregularities is important because many downstream analyses assume a regular time series. We evaluated strategies to address this, including:

- Removing an initial portion of data if irregularities occur mainly during device start-up.
- Rounding all timestamps to the nearest regular interval (5 s in this case).
- Aggregating to a coarser time interval (with some loss of temporal resolution).

Based on the import summary and visual inspection of the time gaps, we chose to round the observation times to the nearest 5-second mark, as this addressed the minor offsets without significant data loss. After rounding timestamps, we added an explicit date column for convenient grouping by day.

We then generated a summary of missing data for each day. Implicit gaps (intervals where the device should have recorded data but did not) were converted into explicit missing entries using LightLogR's gap-handling functions. We also removed days that had very little data to focus on days with substantial wear time. In our `Clouclip` example, days with <1 hour of recordings were dropped. This threshold should be adjusted based on how much complete days matter for a given analysis at hand. E.g., in circadian science, the metrics of *interdaily stability* and *intradaily variation* require measurements for each hour of the day.

After these preprocessing steps, the `Clouclip` dataset had no irregular timestamps remaining and contained explicit markers for all periods of missing data (e.g., times when the device was off or not worn). The distance and illuminance values were now

ready for metric calculations. Because the device was put in `sleep mode` when not worn, there are no measurements during non-wear times.

The `VEET` illuminance data underwent a similar cleaning procedure. To make the `VEET`'s 2-second illuminance data more comparable to the `Clouclip`'s and to reduce computational load, we aggregated the illuminance time series to 5-second intervals. Aggregation was performed with the arithmetic mean of values in a 5-second bin. We then inserted explicit missing entries for each whole day and removed days with more than one hour of missing illuminance data. After cleaning, six days of VEET illuminance data with good coverage remained for analysis (see Supplement 1 for details).

Finally, for spectral analysis, we imported the `VEET`'s spectral sensor modality, and, for the distance analysis, the time-of-flight modality. This required additional processing: the raw spectral data consists of counts from nine wavelength-specific channels (approximately 415 nm through 940 nm, unequally spaced between 30 and 50 nm, plus one broadband clear channel covering the whole range of individual channels, another broadband channel for flicker detection, and a dark channel) along with a sensor gain setting. We aggregated the spectral data to 5-minute intervals to focus on broader trends and reduce data volume. Each channel's counts were normalized by the appropriate gain. Using a calibration matrix provided by the manufacturer (specific to the spectral sensor model), we reconstructed full spectral power distributions for each 5-minute interval. The end result is a list-column in the dataset where each entry is the estimated spectral irradiance across wavelengths for that time interval. Detailed spectral preprocessing steps, including the calibration and normalization, are provided in the Supplement 1. After spectral reconstruction, the dataset was ready for calculating example spectrum-based metrics.

Similarly, the time-of-flight modality contains 256 values per observation, encoding an 8x8 grid of distance and confidence measurements for up to two objects (8x8 grid, times two objects, times distance + confidence column for each object and grid point -> 256 values). For computational reasons, only the first object was kept. These data were pivoted into a long format, where each row contains the distance and confidence data for a given position in the grid and a given datetime. After

pivoting and converting grid positions into a deviation angle from central view, the dataset was ready to be used for distance analysis.

Because the `VEET` devices record even when not worn, a non-wear detection using the devices' actigraphy modality was implemented. This process used the standard deviation of a linear motion sensor in a 5-minute bin with a visually derived threshold to separate wear from non-wear time. Measurements of illuminance and distance were consequently removed during the calculated non-wear times.

This tutorial will start by importing a `Clouclip` dataset and providing an overview of the data. The `Clouclip` export is considerably simpler compared to the `VEET` export, only containing `Distance` and `Illuminance` measurements. The `VEET` dataset will be imported later for the spectrum related metrics.

Load libraries and preprocessed data

```
1  library(LightLogR)
2  library(tidyverse)
3  library(gt)
4  load("data/cleaned/data.RData")
```

Store coordinates of data collection

```
1  coordinates <- c(29.75, -95.36)
```

Line 1 Coordinates for Houston, Texas; coordinates are important to calculate and visualize photoperiods later

Results

Figure 2 shows an overview of the covered workflows in the results section.

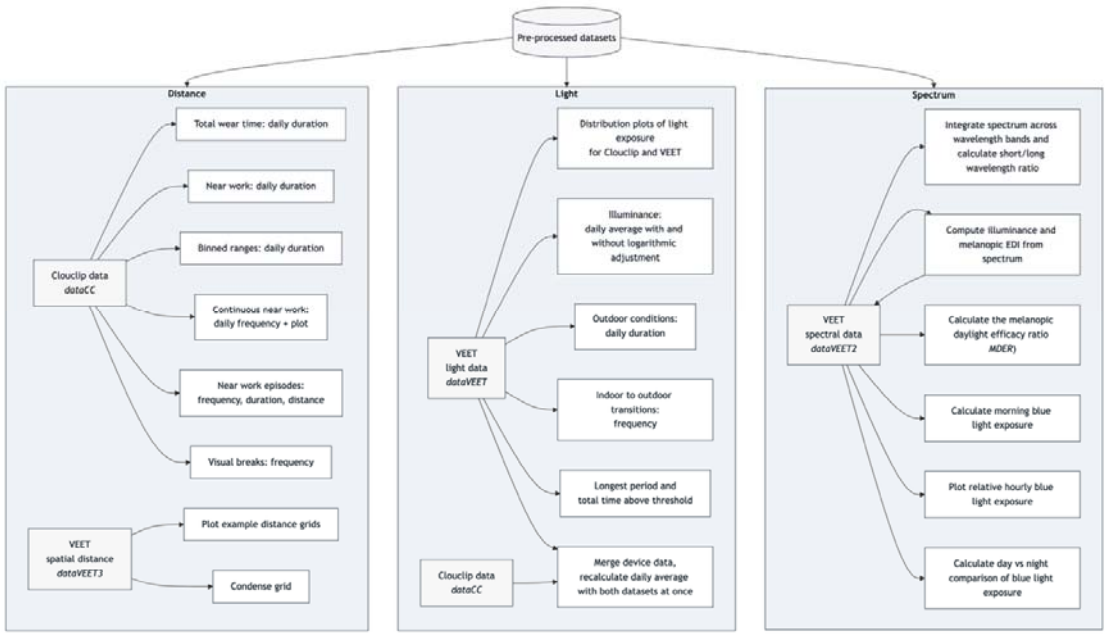


Figure 2: Workflows that are covered in the results section. The pre-processed datasets are covered in detail in the supplement

We first examine metrics related to viewing distance, using the processed `Clouclip` dataset. Many distance-based metrics are computed for each day and then averaged over weekdays, weekends, or across all days. To facilitate this, we define a helper function that will take daily metric values and calculate the mean values for weekdays, weekends, and the overall daily average:

```
Define helper function to_mean_daily()
to_mean_daily <- function(data, prefix = "average_") {
 data |>
    ungroup(Date) |>
    mean_daily(prefix = prefix) |>
    rename_with(.fn = \(x) str_replace_all(x,"_"," ")) |>
    gt()
}
```

Line 3 Ungroup by days
Line 4 Calculate the averages per grouping
Line 5 Remove underscores in names
Line 6 Format as a gt table for display

Total wear time daily

Total wear time daily refers to the amount of time the device was actively collecting distance data each day (i.e. the time the device was worn and operational). We compute this by summing all intervals where a valid distance measurement is present, ignoring periods where data are missing or the device was off. The results are shown in **Table 3**.

```
Calculate total wear time
1  dataCC |>
2    durations(Dis) |>
3    to_mean_daily("Total wear ")
```

Line 2 Calculate total duration of data per day
Line 3 Using the helper function defined above

	Total wear duration
Mean daily	31448s (~8.74 hours)
Weekday	34460s (~9.57 hours)
Weekend	23918s (~6.64 hours)

Table 3: Total wear time per day (average across days)

Duration within distance ranges

Many myopia-relevant metrics concern the time spent at certain viewing distances (e.g., “near work” vs. intermediate or far distances). We calculate the duration of time spent in specific distance ranges. **Table 4** shows the average daily *duration of near work*, defined here as time viewing at 15–60 cm. **Table 5** provides a more detailed breakdown across multiple distance bands.

Duration of near work

```
Calculate daily duration of near work
1  dataCC |>
2    filter(Dis >= 15, Dis < 60) |>
```

```
3    durations(Dis) |>
4    to_mean_daily("Near work ")
```

Line 2 Consider only distances in [15, 60) cm
Line 3 Total duration in that range per day

	Near work duration
Mean daily	17309s (~4.81 hours)
Weekday	21173s (~5.88 hours)
Weekend	7648s (~2.12 hours)

Table 4: Daily duration of near work (15–60 cm viewing distance)

Duration within distance ranges

First, we define a set of distance breakpoints and descriptive labels for each range:

Defining distance ranges (in cm)

```
1  dist_breaks <- c(15, 20, 30, 40, 50, 60, 70, 80, 90, 100, Inf)
2  dist_labels <- c(
3      "Extremely near",          # [15, 20)
4      "Very near",               # [20, 30)
5      "Fairly near",             # [30, 40)
6      "Near",                    # [40, 50)
7      "Moderately near",         # [50, 60)
8      "Near intermediate",       # [60, 70)
9      "Intermediate",            # [70, 80)
10     "Moderately intermediate", # [80, 90)
11     "Far intermediate",        # [90, 100)
12     "Far"                      # [100, Inf)
13  )
```

Now we cut the distance data into these ranges and compute the daily duration spent in each range:

Calculate daily duration within viewing distance range

```
1  dataCC |>
2   mutate(Dis_range =
3             cut(Dis, breaks = dist_breaks, labels = dist_labels)) |>
4    drop_na(Dis_range) |>
5    group_by(Dis_range, .add = TRUE) |>
6    durations(Dis) |>
7    pivot_wider(names_from = Dis_range, values_from = duration) |>
8    ungroup() |>
9    mean_daily(prefix = "") |>
10   pivot_longer(-Date) |>
11   pivot_wider(names_from = Date) |>
12   mutate(name = factor(name, levels = rev(dist_labels))) |>
```

```
13   arrange(name) |>
14   gt() |>
15   fmt_duration(input_units = "seconds", output_units = "minutes")
```

Line 2-3 Categorize distances
Line 4 Remove intervals with no data
Line 5 Group by distance range (in addition to the date)
Line 6 Duration per range per day
Line 7 Pivot data from long to wide format (ranges as columns)
Line 15 Convert seconds to minutes

	Mean daily	Weekday	Weekend
Far	16m	20m	5m
Far intermediate	11m	14m	2m
Moderately intermediate	5m	6m	3m
Intermediate	4m	6m	1m
Near intermediate	7m	7m	8m
Moderately near	13m	16m	5m
Near	27m	36m	5m
Fairly near	46m	60m	12m
Very near	102m	128m	38m
Extremely near	74m	88m	40m

Table 5: Daily duration in each viewing distance range in minutes. For better readability, the table was transposed (compared to the code results)

To visualize this, **Figure 3** illustrates the relative proportion of time spent in each distance range:

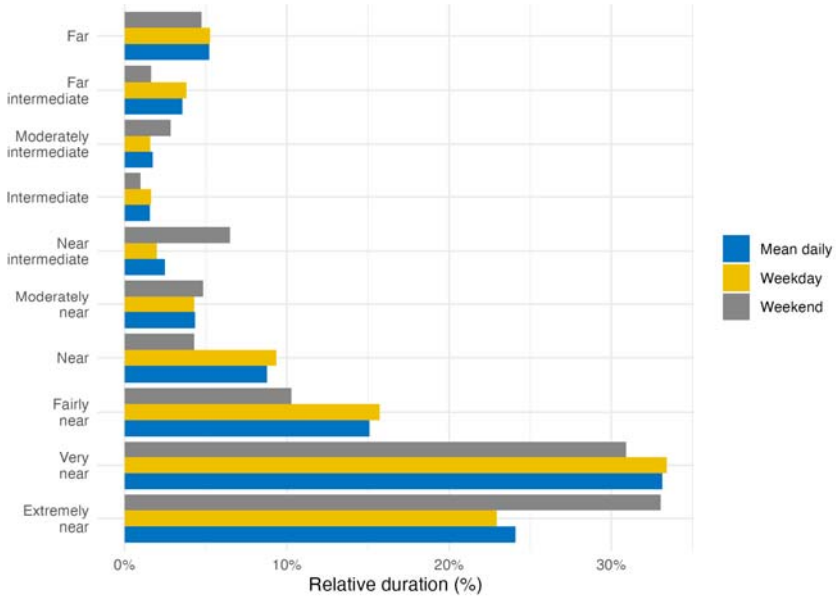


Figure 3: Percentage of total time spent in each viewing distance range for an average day (mean daily), average weekday, or weekend

Frequency of continuous near work

Continuous near-work can be understood as sustained viewing within a near distance for some minimum duration, allowing only brief interruptions. We use LightLogR's `cluster` function to identify episodes of continuous near work. Here, we define a near-work episode as viewing distance between 20 cm and 60 cm that lasts at least 30 minutes, with interruptions of up to 1 minute allowed (meaning short breaks ≤1 min do not end the episode). Using extract_clusters() with those parameters, we count how many such episodes occur per day.

Table 6 summarizes the average frequency of continuous near-work episodes per day, and **Figure 4** provides an example visualization of these episodes on the distance time series.

Calculate the frequency of continuous near-work episodes per day

```
dataCC |>
  extract_clusters(
    Dis >= 20 & Dis < 60,
    cluster.duration = "30 mins",
    interruption.duration = "1 min",
    drop.empty.groups = FALSE
  ) |>
```

```
8    summarize_numeric(remove = c("start", "end", "epoch", "duration"),
9                      add.total.duration = FALSE) |>
10   mean_daily(prefix = "Frequency of ") |>
11   gt() |> fmt_number()
```

Line 3 Condition: near-work distance
Line 4 Minimum duration of a continuous episode
Line 5 Maximum gap allowed within an episode
Line 6 Keep days with zero episodes in output
Line 9 Count number of episodes per day
Line 10 Compute daily mean frequency

	Frequency of episodes
Mean daily	0.86
Weekday	1.20
Weekend	0

Table 6: Frequency of continuous near-work episodes per day

Plot continuous near-work episodes

```
1 dataCC |>
2   add_clusters(
3     Dis >= 20 & Dis < 60,
4     cluster.duration = "30 mins",
5     interruption.duration = "1 min"
6   ) |>
7   gg_day(y.axis = Dis, y.axis.label = "Distance (cm)", geom = "line",
8          y.scale = "identity", y.axis.breaks = seq(0,100, by = 20)) |>
9   gg_photoperiod(coordinates) |>
10  gg_state(state, fill = "red") +
11  geom_hline(yintercept = c(20, 60), col = "red", linetype = "dashed")+
12  coord_cartesian(ylim = c(0,100))
```

Lines 1-6 As in code cell above
Line 9 Add photoperiod information
Line 10 Add state bands

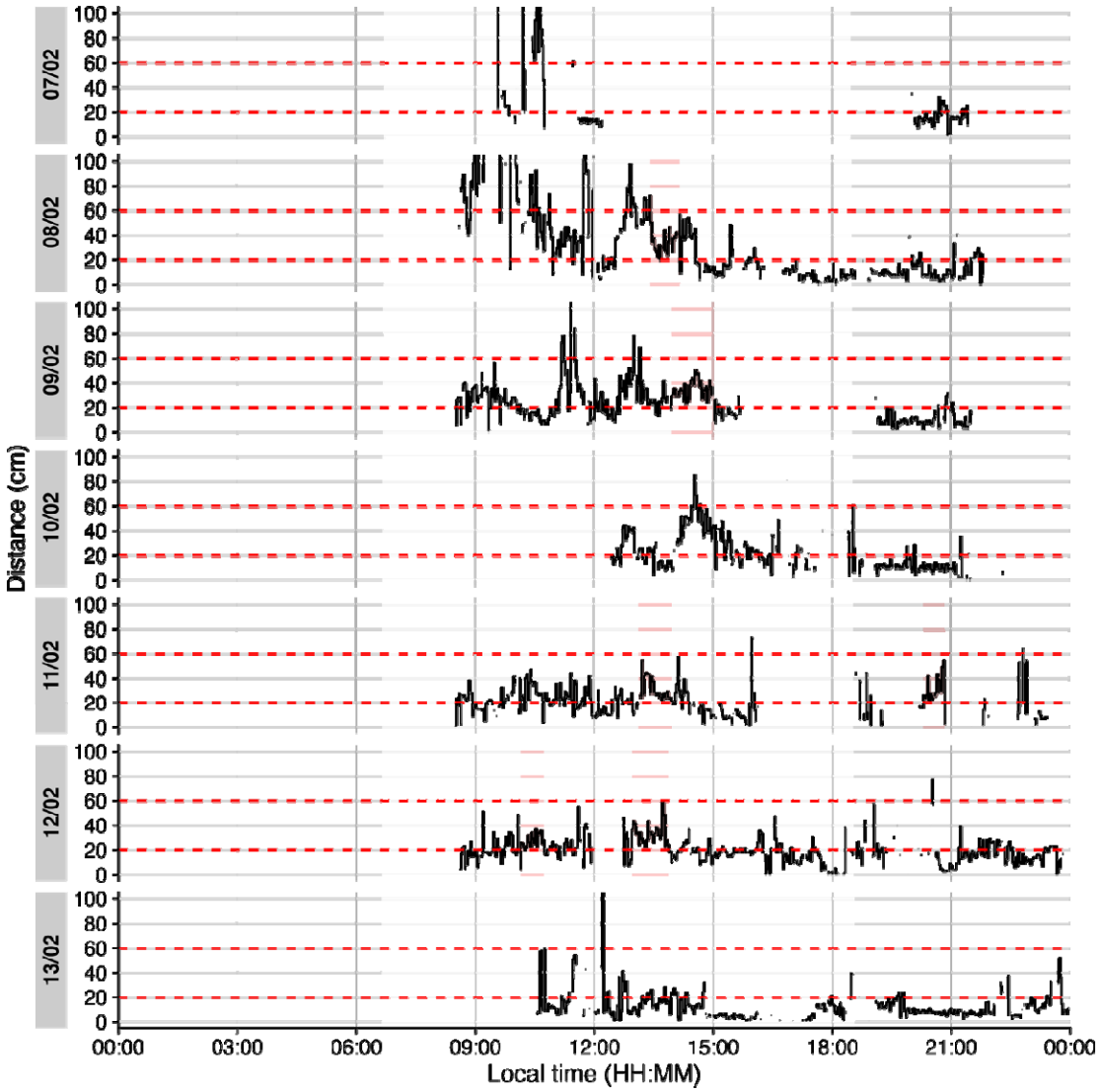


Figure 4: Example of continuous near-work episodes. Red shaded areas indicate periods of continuous near work (20–60 cm for ≥30 min, allowing ≤1 min interruptions). Black trace is viewing distance over time of day; red dashed lines mark the 20 cm and 60 cm boundaries. Grey shaded areas show nighttime between civil dusk and civil dawn

Near-work episodes

Beyond frequency, we can characterize near-work episodes by their duration and typical viewing distance. This section extracts all near-work episodes (using a 5-second minimum duration to capture more routine near-work bouts) and summarizes three aspects:

1. frequency (count of episodes per day),
2. average duration of episodes, and
3. average distance during those episodes.

These results are combined in **Table 7**.

Calculate near-work episodes

```
1 dataCC |>
2   extract_clusters(
3     Dis >= 20 & Dis < 60,
4     cluster.duration = "5 secs",
5     interruption.duration = "20 secs",
6     drop.empty.groups = FALSE
7   ) |>
8   extract_metric(dataCC, distance = mean(Dis, na.rm = TRUE)) |>
9   summarize_numeric(remove = c("start", "end", "epoch"),
10                    prefix = "",
11                    add.total.duration = FALSE) |>
12  mean_daily(prefix = "") |>
13  gt() |>
14  fmt_number(c(distance, episodes), decimals = 0) |>
15  cols_units(distance = "cm")
```

Line 4 Minimal duration to count as an episode (set to interval level of `dataCC`)
Line 8 Calculate mean distance during each episode
Lines 9-11 Calculate averages for all numeric columns per group
Line 12 Daily averages for each metric
Lines 13-15 Table generation

	duration	distance, cm	episodes
Mean daily	233s (~3.88 minutes)	32	57
Weekday	284s (~4.73 minutes)	32	64
Weekend	104s (~1.73 minutes)	32	40

Table 7: Near-work episodes: frequency, mean duration, and mean viewing distance

In the code cell above, `extract_metric(..., distance = mean(Dis, ...))` computes the mean viewing distance during each episode, and the subsequent `summarize_numeric` and `mean_daily` steps derive daily averages of episode count, duration, and distance.

Visual breaks

Visual breaks as defined in this article, require a minimum break-length, and the previous episode is important. This leads to a two step process, where we first

extract instances of `Distance` above 100 cm for at least 20 seconds, before we filter for a previous duration of at maximum 20 minutes. **Table 8** provides the daily frequency of visual breaks and **Figure 5** shows when these occur.

Calculate visual breaks

```
1 dataCC |>
2   extract_clusters(Dis >= 100,
3                    cluster.duration = "20 secs",
4                    return.only.clusters = FALSE,
5                    drop.empty.groups = FALSE
6                    ) |>
7   filter((start - lag(end) <= duration("20 mins")), is.cluster) |>
8   summarize_numeric(
9          remove = c("start", "end", "epoch", "is.cluster", "duration"),
10         prefix = "",
11         add.total.duration = FALSE) |>
12  mean_daily(prefix = "Daily ") |>
13  gt() |>
14  fmt_number(decimals = 1)
```

Line 2 Define the condition, greater 100 cm away
Line 3 Define the minimum duration
Line 4 Return non-clusters as well
Line 5 Keep all days, even without clusters
Line 7 Return only clusters with previous episode lengths of maximum 20 minutes
Lines 8-11 Count the number of episodes
Line 12 Calculate daily means
Lines 13-14 Table generation

	Daily episodes
Mean daily	5.9
Weekday	6.2
Weekend	5.0

Table 8: Frequency of visual breaks (n per day)

Plot visual breaks

```
dataCC |>
    extract_clusters(Dis >= 100,
                     cluster.duration = "20 secs",
                     return.only.clusters = FALSE,
                     drop.empty.groups = FALSE
                     ) |>
  filter((start - lag(end) <= duration("20 mins")), is.cluster) %>%
  add_states(dataCC, ., ) |>
  gg_day(y.axis = Dis, y.axis.label = "Distance (cm)", geom = "line") |>
```

```
10  gg_photoperiod(coordinates) +
11  geom_point(data = \(x) filter(x, is.cluster), col = "red")
```

Lines 1-7 As in the code cell above
Line 8 Add the resulting states
Line 10 Add photoperiod information to the plot

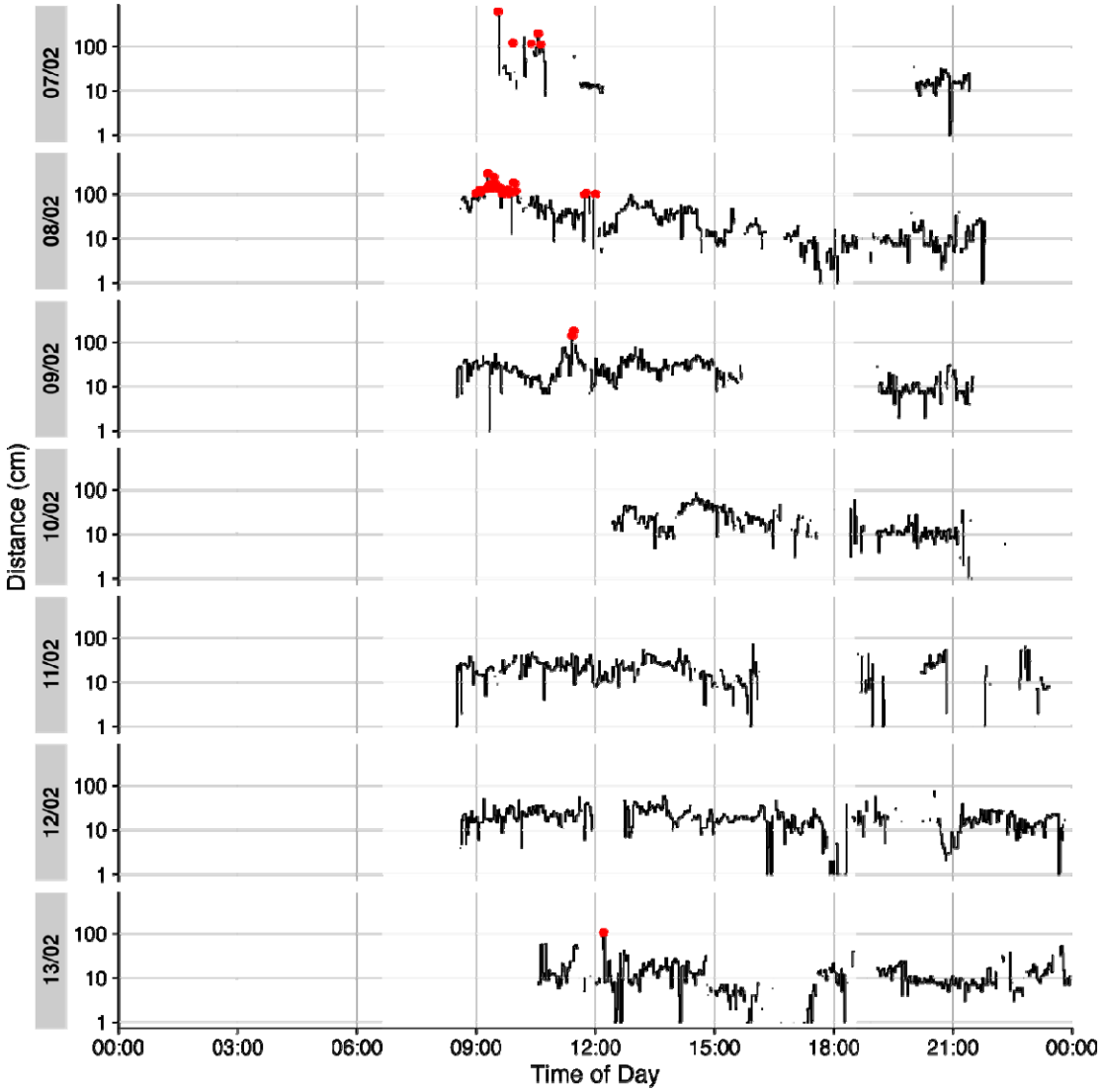


Figure 5: Plot of visual breaks (red dots). Black traces show distance measurement data. Grey shaded areas show nighttime between civil dusk and civil dawn (photoperiod)

Distance with spatial distribution

The `Clouclip` device outputs a singular measure for distance, while the visual environment in natural conditions contains many distances, depending on the solid angle and direction of the measurement. A device like the `VEET` increases the spatial resolution of these measurements, allowing for more in-depth analyses of the size and position of an object within the field of view. In the case of the `VEET`, data are collected from an 8x8 measurement grid, spanning 52° vertically and 41° horizontally. **Figure 6** shows sample observations from six different days at the same time.

Plot spatial distance grids for different days

```
1 dataVEET3 |>
2   filter_Time(start = "13:14:02", length = "00:00:05") |>
3   mutate(dist1 = ifelse(dist1 == 0, Inf, dist1)) |>
4   filter(conf1 >= 0.1 | dist1 == Inf) |>
5   ggplot(aes(x=x.pos, y=y.pos, fill = dist1/10))+
6   extras +
7   facet_wrap(~Datetime)
```

Line 2 Choose a particular observation
Line 3 Replace 0 distances with infinity
Line 4 Remove data that has less than 10% confidence
Line 5-6 Plot the data and add the plot partials from the code cell above
Lines 7 Show one plot per day

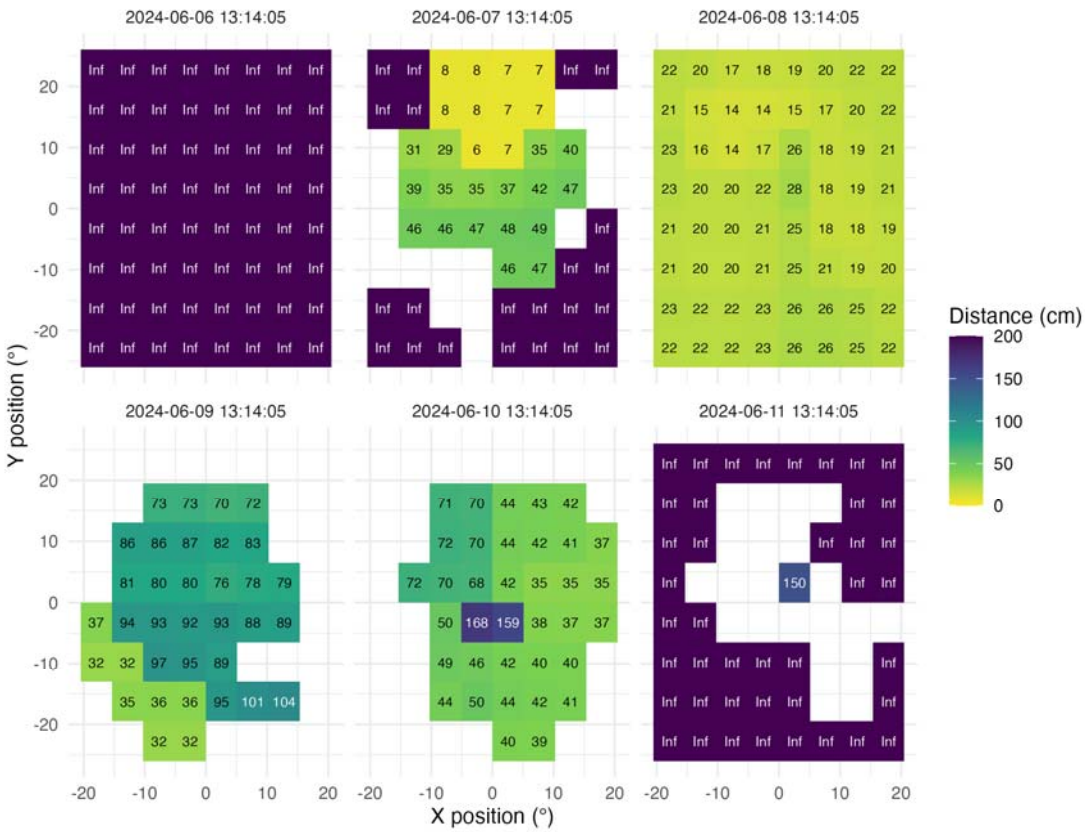


Figure 6: Example observations of the measurement grid at 1:14 p.m. for each measurement day. Text values show distance in cm. Empty grid points show values with low confidence. Zero-distance values were replaced with infinite distance and plotted despite low confidence.

To use these distance data in the framework shown above for the `Clouclip` device, a sensible method to condense the data has to be applied. There are many ways how a spatially resolved distance measure could be utilized for analysis:

- Where in the field of view are objects in close range?
- How large are near objects in the field of view?
- How varied are distances within the field of view?
- How close are objects / is viewing distance in a region of interest within the field of view?

Possible methods include:

- average across all (high confidence) distance values within the grid
- closest (high confidence) distance within the grid

- (high confidence) values at or around a given grid position, e.g., ±10 degrees around the central view (0°)

Many more options are available based on the spatial dataset, e.g., condensation rules based on the number of points in the grid with a given condition, or the variation within the grid.

We will demonstrate these three methods for a single day (`2024-06-10`), all leading to a data structure akin to the `Clouclip`, i.e., to be used for further calculation of visual experience metrics.

Calculating and plotting method results

```
dataVEET3_part <-
dataVEET3 |>
  filter_Date(start = "2024-06-10", length = "1 day")

dataVEET3_condensed <-
dataVEET3_part |>
  group_by(Datetime, .add = TRUE) |>
  filter(conf1 >= 0.1) |>
  summarize(
    distance_mean = mean(dist1),
    distance_min = min(dist1),
    distance_central =
          mean(dist1[between(x.pos, -10,10) & between(y.pos, -10,10)]),
    n = n(),
    .groups = "drop_last"
  )

dataVEET3_condensed |>
  aggregate_Datetime(
     "15 mins", numeric.handler = \(x) mean(x, na.rm = TRUE)) |>
  remove_partial_data(by.date = TRUE) |>
  pivot_longer(contains("distance"),
               names_to = c(".value", "method"),
    names_pattern = "(distance)_(mean|min|max|central)"
    ) |>
  gg_day(y.axis = distance/10,
         geom = "line",
         aes_col = method,
         group = method,
         linewidth = 1,
         alpha = 0.75,
         y.scale = "identity",
         y.axis.breaks = seq(0,150, by = 20),
         y.axis.label = "Distance (cm)"
         ) |>
  gg_photoperiod(coordinates)
```

Lines 1-3 Filter one day
Line 7 Group additionally by every observation
Line 8 Remove data with low confidence
Line 10 Average across all distance values
Line 11 Closest across all distance values
Line 12-13 Central distance
Line 14 Number of (valid) grid points
Line 19-20 Aggregate to 15 minute data
Line 21 Remove data points that fall exactly on midnight of the following day
Lines 22-25 Pivoting the method results from wide to long for plotting
Lines 26-36 Setting up the plot for distance.

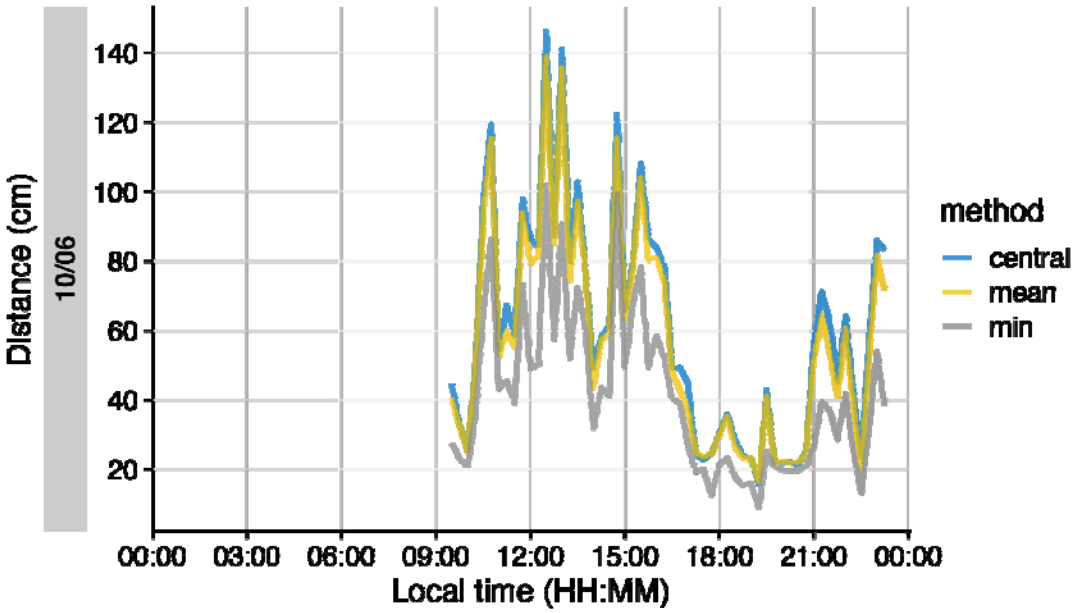


Figure 7: Comparison of condensation methods for spatial grid of distance measurements. The lines represent an average across all data points (yellow), the minimum distance (grey), or the central 10° (blue). Data points with confidence less than 10% were removed prior to calculation.

As can be seen in **Figure 7**, while the overall pattern is similar regardless of the used method, there are notable differences between the methods, which will consequently affect downstream analyses. Most importantly, the process of condensation has to be well documented and reproducible, as shown above. Any of these data could now be used to calculate the `frequency of continuous near work`, `visual breaks`, or `near-work episodes` as described above.

Light

The `Clouclip` illuminance data in our example cover indoor environments and are thus comparatively low, which would make certain daylight exposure summaries trivial or not meaningful. To better illustrate light exposure metrics, we turn to a different dataset, this one taken from the `VEET` device's illuminance data, which capture a broader range of lighting conditions (though both device types are able to capture broadly the same range of illuminance). We import the `VEET` ambient light data (already preprocessed to have regular 5-second intervals as described above) and briefly examine its distribution.

Illuminance distribution: The illuminance values from the `Clouclip` were comparatively low (**Figure 8)**, while the `VEET` (**Figure 9)** data include outdoor exposures up to several thousand lux. The contrast is evident from comparing histograms of the two datasets' lux values (`Clouclip` vs. `VEET`), where the main peak is similarly positioned between 10 and 100 lx, but the tails differ. The `VEET` illuminance histogram (see **Figure 9**) shows a heavily skewed distribution with a considerable number of zero lx values (indicating intervals of complete darkness or the sensor being covered) and a long tail extending to very high lux values. Such zero-inflated and skewed data are common in wearable light measurements (Zauner, Guidolin, and Spitschan, 2025).

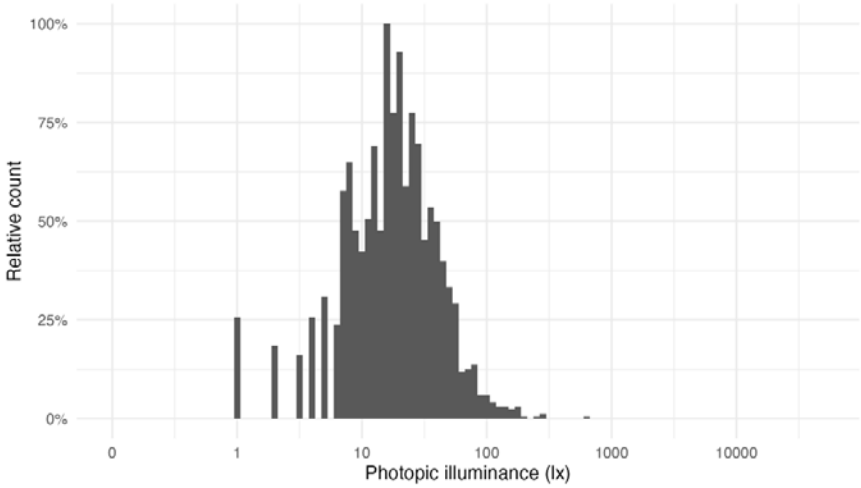


Figure 8: Histogram of illuminance values from the Clouclip dataset (5-second data). The values are typical of indoor conditions.

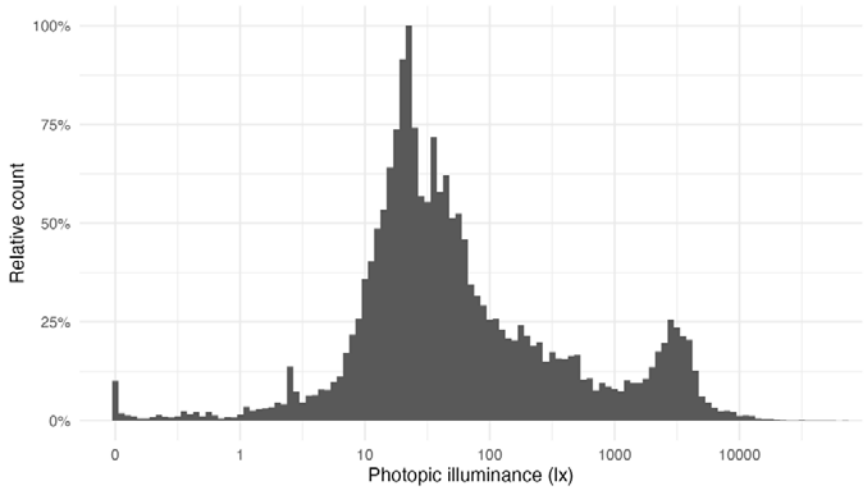


Figure 9: Histogram of illuminance values from the VEET dataset (aggregated to 5 s). Note the logarithmic x-axis: the distribution is highly skewed with many low values (including zeros) and a long tail of high lux readings. Outdoor light exposures in bright conditions are distinguishable around 10^3 to 10^4 lx.

After confirming that the `VEET` data cover a broad dynamic range of lighting, we proceed with calculating light exposure metrics. (The `VEET` data had been cleaned for gaps and irregularities as described earlier, and non-wear times were removed; see Supplement 1 for the details.)

Average light exposure

A basic metric is the average illuminance over the day. **Table 9** shows the mean illuminance (in lux) for weekdays, weekends, and the overall daily mean, calculated directly from the raw lux values.

Calculating mean light exposure per day

```
dataVEET |>
  select(Id, Date, Datetime, Lux) |>
  summarize_numeric(prefix = "mean ", remove = c("Datetime")) |>
  to_mean_daily() |>
  fmt_number(decimals = 1) |>
  cols_hide(`average episodes`) |>
  cols_label(`average mean Lux` = "Mean photopic illuminance (lx)")
```

	Mean photopic illuminance (lx)
Mean daily	481.8

	Mean photopic illuminance (lx)
Weekday	538.1
Weekend	341.1

Table 9: Mean light exposure (illuminance) per day

However, because illuminance data tend to be extremely skewed and contain many zero values (periods of darkness), the arithmetic mean can be misleading. A common approach is to apply a logarithmic transform to illuminance before averaging, which down-weights extreme values and accounts for the multiplicative nature of light intensity effects. LightLogR provides helper functions `log_zero_inflated()` and its inverse `exp_zero_inflated()` to handle log-transformation when zeros are present (by adding a small offset before log, and back-transforming after averaging). Using this approach, we recompute the daily mean illuminance. The results in **Table 10** show that the log-transformed mean (back-transformed to lux) is much lower, reflecting the fact that for much of the time illuminance was near zero. This transformed mean is often more representative of typical exposure for skewed data.

Calculating mean light exposure per day with log transformation

```
1 dataVEET |>
2   select(Id, Date, Datetime, Lux) |>
3   mutate(Lux = Lux |> log_zero_inflated()) |>
4   summarize_numeric(prefix = "mean ", remove = c("Datetime")) |>
5   mean_daily(prefix = "") |>
6   mutate(`mean Lux` = `mean Lux` |> exp_zero_inflated()) |>
7   gt() |> fmt_number(decimals = 1) |> cols_hide(episodes) |>
8   cols_label(`mean Lux` = "Mean photopic illuminance (lx)")
```

Line 3 Log transform with zero handling (base 10)
Line 5 Calculate daily mean of log-lux
Line 6 Back-transform to lux

	Mean photopic illuminance (lx)
Mean daily	57.0
Weekday	70.1
Weekend	33.9

Table 10: Mean light exposure per day (after logarithmic transformation to account for zero inflation and skewness)

Duration in high-light (outdoor) conditions

Another important metric is the amount of time spent under bright light, often used as a proxy for outdoor exposure. We define thresholds corresponding to outdoor light levels (e.g. 1000 lx and above). Here, we categorize each 5-second interval of illuminance into bands: Outdoor bright (≥1000 lx), Outdoor very bright (≥2000 lx), and Outdoor extremely bright (≥3000 lx). We then sum the duration in each category per day.

While daylight levels can far exceed the recorded light levels, those are usually recorded with direct sunlight and without obstruction. Under normal viewing conditions, at eye level, and avoiding glare, daylight levels of a few thousand lux are at the higher end of the distribution (Murukesu, Zauner, and Spitschan 2025). **Figure 9** shows a bimodal distribution, with the right mode representing outdoor lighting conditions. In a 2023 review of light dosimeters to investigate the light-myopia relationship (Hönekopp and Weigelt 2023), 1000 lx was the predominant cutoff value to distinguish indoor vs. outdoor environments. It is not, however, without critique, and both other thresholds (Patterson Gentile et al. 2025) and classification methods are proposed (Tabandeh and Spitschan 2025).

```
Define outdoor illuminance thresholds (in lux)
out_breaks <- c(0, 1e3, 2e3, 3e3, Inf)
out_labels <- c(
    "Indoor",                   # [0, 1000) lx
    "Outdoor bright",           # [1000, 2000) lx
    "Outdoor very bright",      # [2000, 3000) lx
    "Outdoor extremely bright" # [3000, ∞) lx
  )

dataVEET <- dataVEET |>
  mutate(Lux_range = cut(Lux, breaks=out_breaks, labels=out_labels))
```

Now we compute the mean daily duration spent in each of these outdoor light ranges (**Table 11**):

```
Calculate the mean daily duration spent in each light range
dataVEET |>
  drop_na(Lux_range) |>
  group_by(Lux_range, .add = TRUE) |>
```

```
  durations(Lux) |>
  pivot_wider(names_from = Lux_range, values_from = duration) |>
  to_mean_daily("") |>
  fmt_duration(input_units = "seconds", output_units = "minutes")
```

Outdoor	bright	very bright	extremely bright	Indoor
Mean daily	24m	28m	42m	663m
Weekday	29m	35m	48m	688m
Weekend	10m	10m	28m	602m

Table 11: Average daily duration in outdoor-equivalent light conditions (column order is different from code output)

It is also informative to visualize when these high-light conditions occurred. **Figure 10** shows a timeline plot with periods of outdoor-level illuminance highlighted in color. In this example, violet denotes ≥1000 lx, green ≥2000 lx, and yellow ≥3000 lx. Grey shading indicates nighttime (from civil dusk to dawn) for context.

Visualize time spent outdoors

```
1 dataVEET |>
2   aggregate_Datetime("2 mins", type = "floor") |>
3   mutate(Lux_range = fct_recode(Lux_range, NULL = "Indoor")) |>
4   gg_day(y.axis = Lux,
5          y.axis.label = "Photopic illuminance (lx)",
6          geom = "line",
7          jco_color = FALSE) |>
8   gg_state(Lux_range, aes_fill = fct_rev(Lux_range),
9            alpha = 0.75, ymin = 10^3, ymax = 10^4) +
10  scale_fill_viridis_d() +
11  labs(fill = "Illuminance range") +
12  theme(legend.position = "bottom") +
13  coord_cartesian(xlim = c(8, 19.5)*3600)
```

Line 2 Aggregating data to 5-minute bins
Line 3 Removing the indoor condition
Lines 4-7 Setting up the basic plot
Lines 8-9 Adding state information on the illuminance ranges
Line 13 Setting the x-axis limits to cover daytime hours

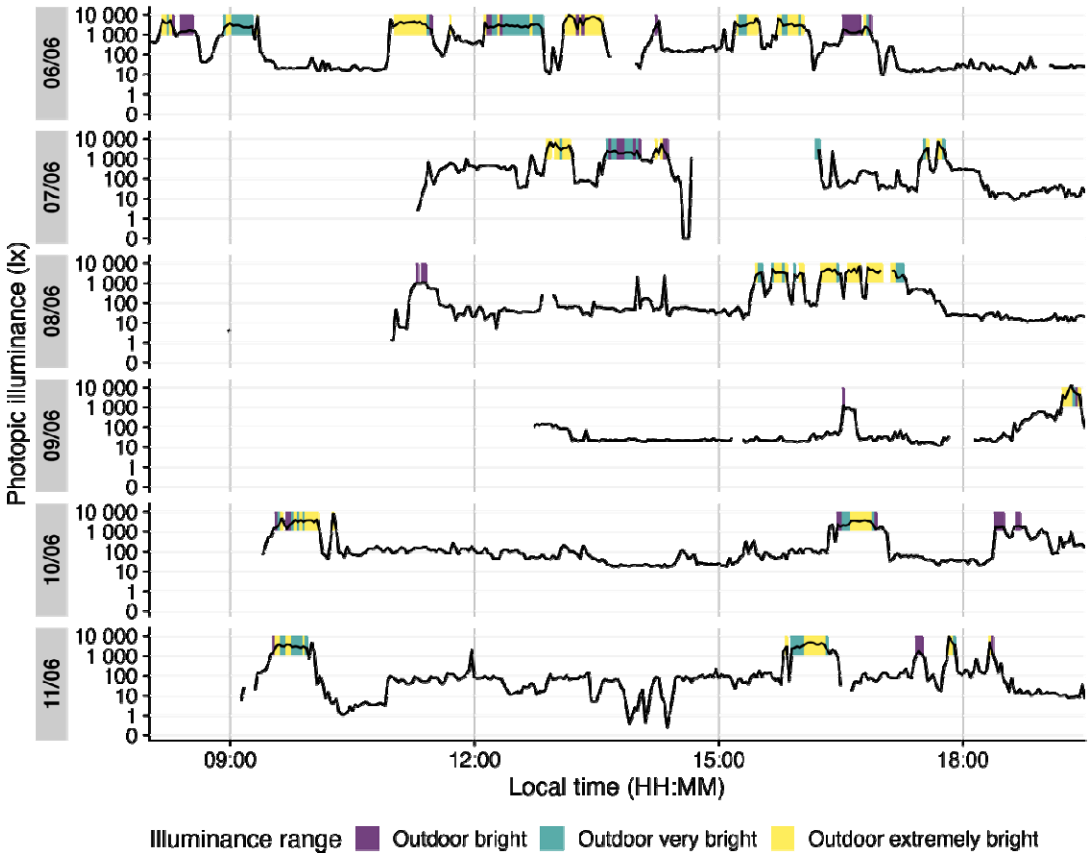


Figure 10: Outdoor light exposure over time with 2-minute interval. Colored bands indicate periods when illuminance exceeded outdoor thresholds for at least half of each interval: violet for ≥1000 lx, green for ≥2000 lx, and yellow for ≥3000 lx. Grey shaded regions denote night (from civil dusk to dawn).

Frequency of transitions from indoor to outdoor light

We next consider how often the subject moved from an indoor light environment to an outdoor-equivalent environment. We operationally define an "outdoor transition" as a change from <1000 lx to ≥1000 lx. Using the cleaned `VEET` data, we extract all instances where illuminance crosses that threshold from below to above.

Table 12 shows the average number of such transitions per day. Note that if data are recorded at a fine temporal resolution (5 s here), very brief excursions above 1000 lx could count as transitions and inflate this number. Indeed, the initial count is

fairly high, reflecting fleeting spikes above 1000 lx that might not represent meaningful outdoor exposures.

```
Calculate the number of transitions from indoor to outdoor
1 dataVEET |>
2   extract_states(Outdoor, Lux >= 1000, group.by.state = FALSE) |>
3   filter(!lead(Outdoor) & Outdoor) |>
4   summarize_numeric(prefix = "mean ",
5     remove = c("Datetime", "Outdoor", "start", "end", "duration"),
6     add.total.duration = FALSE) |>
7   mean_daily(prefix = "") |>
8   gt() |>
9   fmt_number(episodes, decimals = 0) |>
10  fmt_duration(
11   `mean epoch`, input_units = "seconds", output_units = "seconds"
12  )
```

Line 2 Label each interval as Outdoor (Lux≥1000) or not
Line 3 Find instances where the previous interval was “indoor” and current is “outdoor”

	mean epoch	episodes
Mean daily	5s	64
Weekday	5s	72
Weekend	5s	46

Table 12: Average daily count of transitions from indoor (<1000 lx) to outdoor (≥1000 lx) lighting when looking at 5-second epochs

To obtain a more meaningful measure, we can require that the outdoor state persists for some minimum duration to count as a true transition (filtering out momentary fluctuations around the 1000 lx mark). For example, we can require that once ≥1000 lx is reached, it continues for at least 5 minutes (allowing short interruptions up to 20 s). **Table 13** applies this criterion, resulting in a lower, more plausible transition count.

```
Calculate the number of transitions from indoor to outdoor with clusters
1 dataVEET |>
2   extract_clusters(Lux >= 1000,
3                    cluster.duration = "5 min",
4                    interruption.duration = "20 secs",
5                    return.only.clusters = FALSE,
6                    drop.empty.groups = FALSE) |>
```

```
  filter(!lead(is.cluster) & is.cluster) |>
  summarize_numeric(prefix = "mean ",
    remove = c("Datetime", "start", "end", "duration"),
    add.total.duration = FALSE) |>
  mean_daily(prefix = "") |>
  gt() |> fmt_number(episodes, decimals = 0)
```

	mean epoch	episodes
Mean daily	5s	5
Weekday	5s	6
Weekend	5s	4

Table 13: Daily indoor-to-outdoor transition count (requiring ≥5 min duration of ≥1000 lx to count)

Longest sustained bright-light period

The final light exposure metric we illustrate is the longest continuous period above a certain illuminance threshold (often termed longest period above threshold, e.g. PAT1000 for 1000 lx). This gives us a sense of the longest outdoor exposure in a day. Along with it, one might report the total duration above that threshold in the day (TAT1000). While we could derive these from the earlier analyses, LightLogR provides dedicated metric functions for such calculations, which can compute multiple related metrics at once.

Using the function period_above_threshold() for PAT and duration_above_threshold() for TAT, we calculate both metrics for the 1000 lx threshold. **Table 14** shows the mean of these metrics across days (i.e., average longest bright period and average total bright time per day).

Calculate PAT1000 and TAT1000

```
dataVEET |>
  summarize(
    period_above_threshold(
      Lux, Datetime, threshold = 1000, na.rm = TRUE, as.df = TRUE),
    duration_above_threshold(
      Lux, Datetime, threshold = 1000, na.rm = TRUE, as.df = TRUE),
    .groups = "keep"
  ) |>
  to_mean_daily("")
```

	period above 1000	duration above 1000
Mean daily	1208s (~20.13 minutes)	5703s (~1.58 hours)
Weekday	1469s (~24.48 minutes)	6815s (~1.89 hours)
Weekend	555s (~9.25 minutes)	2922s (~48.7 minutes)

Table 14: Longest period and total duration above 1000 lx (PAT1000 and TAT1000)

Merging data streams

Note that while imports from different devices can be merged, devices differ in their sensors, electronics, housing or diffuser form factors, and on-device data-processing pipelines. All of these factors affect the comparability of measurements, even when devices output the same variable (e.g., illuminance or distance). If data from different devices with the same measurement variable are to be merged, the corresponding variable names should be standardized beforehand - for example, renaming `Lux` and `LIGHT` to `illuminance`. If we wanted to analyse the `VEET` data together with the `Clouclip` data, for example, we would not have to rename anything, as both carry their illuminance measurements in the variable `Lux`. The following example shows how the combination of datasets would lead to a combined dataset, and how that would affect analysis outcomes. It is the responsibility of the researcher to perform device calibration and/or checks for a similar measurement fidelity.

```
Merge Clouclip and VEET data
1 data <- join_datasets(dataCC, dataVEET)
2 data |> summary_overview(Lux, threshold.missing = 0.5) |> gt()
```

	mean	min	max
Participants (n)	2		
Participant-days (n)	13	6	7
Days ≥50% complete (n)	6	3	3
Missing/Irregular (%)	52%	38%	86%

Table 15: Overview of the merged dataset

We will reuse the example from Section 3.2.1, but instead of one participant, we now have data from two devices and participants

```
Re-calculate mean photopic illuminance with the merged data
1 data |>
2   select(Id, Date, Datetime, Lux) |>
3   mutate(Lux = Lux |> log_zero_inflated()) |>
4   summarize_numeric(prefix = "mean ", remove = c("Datetime")) |>
5   mean_daily(prefix = "") |>
6   mutate(`mean Lux` = `mean Lux` |> exp_zero_inflated()) |>
7   gt() |> fmt_number(decimals = 1) |> cols_hide(episodes) |>
8   cols_label(`mean Lux` = "Mean photopic illuminance (lx)")
```

Line 1 Instead of `dataVEET` we now supply the merged `data` object
Lines 2-8 Verbatim from Section 3.2.1

	Mean photopic illuminance (lx)
Clouclip	
Mean daily	17.6
Weekday	18.6
Weekend	15.2
VEET	
Mean daily	57.0
Weekday	70.1
Weekend	33.9

Table 16: Recalculation of the mean light exposure per day (after logarithmic transformation to account for zero inflation and skewness) with the merged dataset

Spectrum

The `VEET` device's spectral sensor provides multimodal data beyond simple lux values, but it requires reconstruction of the actual light spectrum from raw sensor counts. We processed the spectral sensor data in order to compute two example spectrum-based metrics. Detailed data import, normalization, and spectral reconstruction steps are given in Supplement 1; here we present the resulting metrics. Briefly, the `VEET`'s spectral sensor recorded counts in nine wavelength bands (roughly 415 nm to 910 nm), plus a `Dark`, a `Clear`, and a flicker detection

channel[1]. After normalizing by sensor gain and applying the calibration matrix, we obtained an estimated spectral irradiance distribution for each 5-minute interval in the recording. With these reconstructed spectra, we can derive novel metrics that consider spectral content of the light.

Spectrum-based metrics in wearable data are relatively new and less established compared to distance or broadband light metrics. The following examples illustrate potential uses of spectral data in a theoretical sense, which can be adapted as needed for specific research questions.

Ratio of short- vs. long-wavelength light

Our first spectral metric is the ratio of short-wavelength light to long-wavelength light, which is relevant, for example, in assessing the blue-light content of exposure. We define “short” wavelengths as 400–500 nm and “long” as 600–700 nm (which are not standardized thresholds and can be freely adjusted). Using the list-column of spectra in our dataset, we integrate each spectrum over these ranges (using `spectral_integration()`), and then compute the ratio short/long for each time interval. We then summarize these ratios per day.

Extract wavelength sections and integrate over them

```
dataVEET <- dataVEET2 |>
  select(Id, Date, Datetime, Spectrum) |>
  mutate(
    short =
      Spectrum |>
      map_dbl(spectral_integration, wavelength.range = c(400, 500)),
    long  =
      Spectrum |>
      map_dbl(spectral_integration, wavelength.range = c(600, 700)),
    `sl ratio` = ifelse(is.nan(short / long), NA, short / long)
  )
```

---

[1] Note that older firmware versions of the `VEET` prior to *2.1.7* contained two `Clear` channels and the highest spectral channel was indicated as 940 nm. Data collected with this early firmware version are not suitable for spectral reconstruction in the context of research projects.

Line 2 Focus on ID, date, time, and spectrum
Line 10 Compute short-to-long wavelength ratio

Table 17 shows the average short/long wavelength ratio, averaged over each day (and then as weekday/weekend means if applicable). In this dataset, the values give an indication of the spectral balance of the light the individual was exposed to (higher values mean relatively more short-wavelength content).

```
Calculate daily average values of short and long wavelength content
1 dataVEET |>
2   summarize_numeric(prefix = "", remove = c("Datetime", "Spectrum")) |>
3   gt() |>
4   fmt_number(decimals = 1, scale_by = 1000) |>
5   fmt_number(`sl ratio`, decimals = 3) |>
6   cols_hide(episodes)
```

date	short	long	sl-ratio
2025-06-18	83.7	71.7	0.610
2025-06-20	114.0	81.6	0.372

Table 17: Average (mW/m²) and ratio of short-wavelength (400–500 nm) to long-wavelength (600–700 nm) light

Melanopic daylight efficacy ratio (MDER)

The same idea is behind calculating the melanopic daylight efficacy ratio (or MDER), which is defined by the CIE ("CIE System for Metrology of Optical Radiation for ipRGC-Influenced Responses to Light" 2018) as the melanopic EDI divided by the photopic illuminance (Hartmeyer and Andersen 2023). Results are shown in **Table 18**. In this case, instead of a simple integration over a wavelength band, we apply an action spectrum to the spectral power distribution (SPD), integrate over the weighted SPD, and apply a correction factor. All alphaopic action spectra are implemented in the `spectral_integration()` function. These will result in photopic illuminance and melanopic equivalent daylight illuminance (melEDI)

```
Calculate melEDI and illumiance
1  dataVEET <-
2    dataVEET |>
3    select(Id, Date, Datetime, Spectrum, short, long, `sl ratio`) |>
4    mutate(
5      melEDI =
6       Spectrum |>
```

```
7       map_dbl(spectral_integration, action.spectrum = "melanopic"),
8      illuminance  =
9       Spectrum |>
10      map_dbl(spectral_integration, action.spectrum = "photopic")
11  )
```

Lines 5-7 Calculate melanopic EDI by applying the $s_{mel(\lambda)}$ action spectrum, integrating, and weighing
Lines 8-10 Calculate photopic illuminance by applying the $V_{(\lambda)}$ action spectrum, integrating, and weighing

Calculate MDER

```
1  dataVEET  |>
2    summarize_numeric(prefix = "", remove = c("Datetime", "Spectrum")) |>
3    mutate(MDER = melEDI / illuminance) |>
4    gt() |>
5    fmt_number(-`sl ratio`, decimals = 1, scale_by = 1000) |>
6    fmt_number(c(MDER, `sl ratio`), decimals = 3) |>
7    cols_hide(episodes)
```

Date	melEDI	Illuminance	MDER
2025-06-18	80.6	103.7	0.777
2025-06-20	108.0	123.9	0.871

Table 18: Average melanopic daylight efficacy ratio (MDER)

Short-wavelength light at specific times of day

The third spectral example examines short-wavelength light exposure as a function of time of day. Certain studies might be interested in, for instance, blue-light exposure during midday versus morning or night. We demonstrate three approaches: (a) filtering the data to a specific local time window, and (b) aggregating by hour of day to see a daily profile of short-wavelength exposure. Additionally, we (c) look at differences between day and night periods.

Local morning exposure

Table 19 isolates the time window between 7:00 and 11:00 each day and computes the average short-wavelength irradiance in that interval. This represents a straightforward query: “How much blue light does the subject get in the morning on average?”

```
Calculate short-wavelength light before noon
1 dataVEET |>
2   filter_Time(start = "7:00:00", end = "11:00:00") |>
3   select(c(Id, Date, short)) |>
4   summarize_numeric(prefix = "") |>
5   gt() |>
6   fmt_number(short, scale_by = 1000) |>
7   cols_label(short = "Short-wavelength irradiance (mW/m²)") |>
8   cols_hide(episodes)
```

Line 2 Filter data to local 7am–11am

Date	Short-wavelength irradiance (mW/m²)
2025-06-18	5.44
2025-06-20	0.95

Table 19: Average short-wavelength light (400–500 nm) exposure between 7:00 and 11:00 each day

Hourly profile across the day

To visualize short-wavelength exposure over the course of a day, we aggregate the data into hourly bins. We cut the timeline into 1-hour segments (using local time), compute the mean short-wavelength irradiance in each hour for each day. **Figure 11** shows the resulting diurnal profile, with short-wavelength exposure expressed as a fraction of the daily maximum for easier comparison.

```
Plot a diurnal profile
1  dataVEETtime <- dataVEET |>
2    cut_Datetime(unit = "1 hour", type = "floor", group_by = TRUE) |>
3    select(-c(Spectrum, long, Datetime)) |>
4    summarize_numeric(prefix = "") |>
5    add_Time_col(Datetime.rounded)  |>
6    mutate(rel_short = short / max(short))
7
8  dataVEETtime |>
9    ggplot(aes(x=Time, y = rel_short)) +
10   geom_col(aes(fill = factor(Date)), position = "dodge") +
11   ggsci::scale_fill_jco() +
12   theme_minimal() +
13   labs(y = "Normalized short-wavelength irradiance",
14        x = "Local time (HH:MM)",
15        fill = "Date") +
```

```
16   scale_y_continuous(labels = scales::label_percent()) +
17   scale_x_time(labels = scales::label_time(format = "%H:%M"))
```

Line 1 Prepare hourly binned data
Line 2 Bin timestamps by hour
Line 5 Add a Time column (hour of day)
Line 8-17 Creating the plot

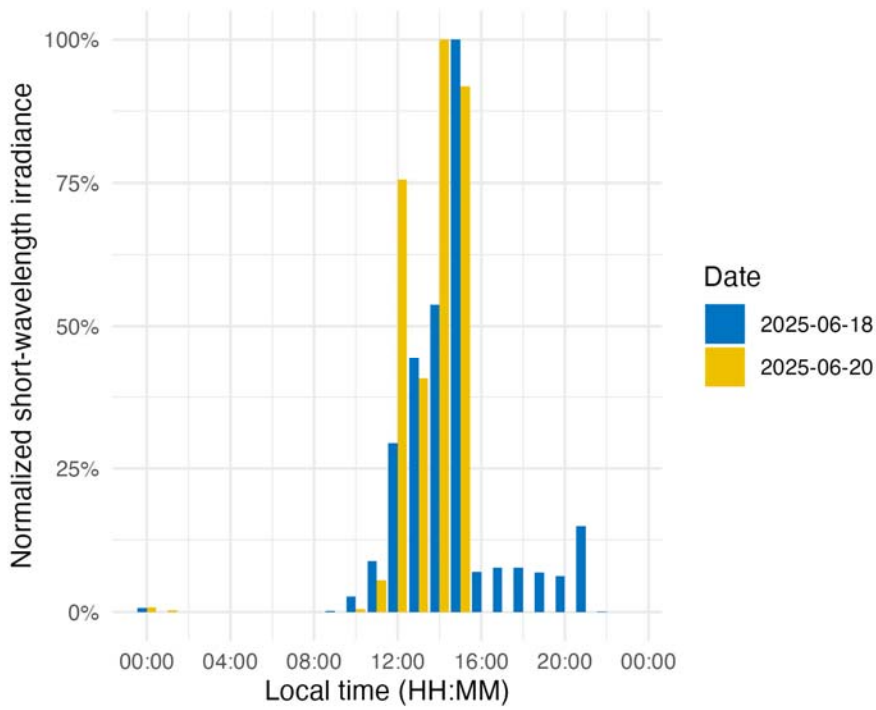


Figure 11: Diurnal profile of short-wavelength light exposure. Each bar represents the average short-wavelength irradiance at that hour of the day (0–23 h), normalized to the daily maximum.

Day vs. night (photoperiod)

Finally, we compare short-wavelength exposure during daytime vs. nighttime. Using civil dawn and dusk information (based on geographic coordinates, here set for Houston, TX, USA), we label each measurement as day or night and then compute the total short-wavelength exposure in each period. **Table 20** summarizes the daily short-wavelength dose received during the day vs. during the night.

```
Calculate photoperiod dependent measures
1  dataVEET |>
2    select(-c(Spectrum, long, `sl ratio`, melEDI, illuminance)) |>
```

```
  add_photoperiod(coordinates) |>
  group_by(photoperiod.state, .add = TRUE) |>
  summarize_numeric(prefix = "",
                    remove =
                      c("dawn", "dusk", "photoperiod", "Datetime")) |>
  group_by(Id, photoperiod.state) |>
 select(-episodes) |>
  pivot_wider(names_from =photoperiod.state, values_from = short) |>
  gt() |>
  fmt_number(scale_by = 1000, decimals = 1)
```

date	day	night
2025-06-18	126.5	12.3
2025-06-20	181.8	1.0

Table 20: Short wavelength light exposure (mW/m²) during the day and at night

In the code cell above, `add_photoperiod(coordinates)` is used as a convenient way to add columns to the data frame, indicating for each timestamp whether it was day or night, given the latitude/longitude.

Discussion and conclusion

This tutorial demonstrates a standardized, step-by-step pipeline to calculate a variety of visual experience metrics. We illustrated how a combination of LightLogR functions and tidyverse workflows can yield clear and reproducible analyses for wearable device data. While the full pipeline is detailed, each metric is computed through a dedicated sequence of well-documented steps, yet remains configurable to realize different metric definitions or thresholds.

By leveraging LightLogR's framework alongside common data analysis approaches, the process remains transparent and relatively easy to follow. The overall goal is to make analysis transparent (with open-source functions), accessible (through thorough documentation, tutorials, and human-readable function naming, all under an MIT license), robust (the package includes >900 unit tests and continuous integration with bug tracking on GitHub), and community-driven (open feature requests and contributions via GitHub).

Even with standardized pipelines, researchers must still make and document many decisions during data cleaning, time-series handling, and metric calculations — especially for complex metrics that involve grouping data in multiple ways (for example, grouping by distance range as well as by duration for cluster metrics). We have highlighted these decision points in the tutorial (such as how to handle irregular intervals, choosing thresholds for “near” distances or “outdoor” light, and deciding on minimum durations for sustained events). Explicitly considering and reporting these choices is important for reproducibility and for comparing results across studies.

The broad set of features in LightLogR — ranging from data import and cleaning tools (for handling time gaps and irregularities) to visualization functions and metric calculators — make it a powerful toolkit for visual experience research. Our examples spanned circadian-light metrics and myopia-related metrics, demonstrating the versatility of a unified analysis approach. By using community-supported tools and workflows, researchers in vision science, chronobiology, myopia, and related fields can reduce time spent on low-level data wrangling and focus more on interpreting results and advancing scientific understanding.

Statements

Acknowledgement

We thank George Hatoun and David Sullivan (Reality Labs Research) for reviewing a draft of the tutorial manuscript and providing sample data for the VEET device for spectral analyses.

Data availability statement

All data and code in this tutorial and Supplement 1 are available from the GitHub repository: https://github.com/tscnlab/ZaunerEtAl_JVis_2026/, archived on Zenodo: https://doi.org/10.5281/zenodo.16566014 under a MIT license (data under CC-BY license).

Funding statement

JZ’s position is funded by the MeLiDos project. The project has received funding from the European Partnership on Metrology (22NRM05 MeLiDos), co-financed from

the European Union's Horizon Europe Research and Innovation Programme and by the Participating States. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or EURAMET. Neither the European Union nor the granting authority can be held responsible for them. JZ, LAO, and MS received research funding from Reality Labs Research. The funders had no role in study design, data collection and analysis, decision to publish or preparation of the manuscript.

Conflict of interest statement

JZ declares the following potential conflict of interest in the past five years (2021-2025). Funding: Received research funding from Reality Labs Research.

AN declares the following potential conflicts of interest in the past five years (2021-2025). none

LAO declares the following potential conflict of interest in the past five years (2021-2025). Consultancy: Zeiss, Alcon, EssilorLuxottica; Research support: Topcon, Meta, LLC; Patents: US 11375890 B2

MS declares the following potential conflicts of interest in the past five years (2021–2025). Academic roles: Member of the Board of Directors, Society of Light, Rhythms, and Circadian Health (SLRCH); Chair of Joint Technical Committee 20 (JTC20) of the International Commission on Illumination (CIE); Member of the Daylight Academy; Chair of Research Data Alliance Working Group Optical Radiation and Visual Experience Data. Remunerated roles: Speaker of the Steering Committee of the Daylight Academy; Ad-hoc reviewer for the Health and Digital Executive Agency of the European Commission; Ad-hoc reviewer for the Swedish Research Council; Associate Editor for LEUKOS, journal of the Illuminating Engineering Society; Examiner, University of Manchester; Examiner, Flinders University; Examiner, University of Southern Norway. Funding: Received research funding and support from the Max Planck Society, Max Planck Foundation, Max Planck Innovation, Technical University of Munich, Wellcome Trust, National Research Foundation Singapore, European Partnership on Metrology, VELUX Foundation, Bayerisch-Tschechische Hochschulagentur (BTHA), BayFrance (Bayerisch-Französisches Hochschulzentrum), BayFOR (Bayerische Forschungsallianz), and Reality Labs

Research. Honoraria for talks: Received honoraria from the ISGlobal, Research Foundation of the City University of New York and the Stadt Ebersberg, Museum Wald und Umwelt. Travel reimbursements: Daimler und Benz Stiftung. Patents: Named on European Patent Application EP23159999.4A ("System and method for corneal-plane physiologically-relevant light logging with an application to personalized light interventions related to health and well-being"). With the exception of the funding source supporting this work, M.S. declares no influence of the disclosed roles or relationships on the work presented herein.

Statement of generative AI and AI-assisted technologies in the writing process

The authors used ChatGPT during the preparation of this work. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Use of AI in contributor roles [2]: Conceptualization: no Data curation: no Formal analysis: bug fixing Methodology: no Software: bug fixing Validation: no Visualization: tweaking of options Writing – original draft: abstract refinement Writing – review & editing: improve readability and language

References

Bhandari, Khob R, and Lisa A Ostrin. 2020. "Validation of the Clouclip and Utility in Measuring Viewing Distance in Adults." *Ophthalmic and Physiological Optics* 40 (6): 801–14. https://doi.org/https://doi.org/10.1111/opo.12735.

[2] Based on the CRediT taxonomy. Funding acquisition, investigation, project administration, resources, and supervision were deemed irrelevant in this context and thus removed.

Biller, A. M., P. Balakrishnan, and M. Spitschan. 2024. “Behavioural Determinants of Physiologically-Relevant Light Exposure.” Journal Article. *Commun Psychol* 2 (1): 114. https://doi.org/10.1038/s44271-024-00159-5.

Blume, C., C. Garbazza, and M. Spitschan. 2019. “Effects of Light on Human Circadian Rhythms, Sleep and Mood.” Journal Article. *Somnologie (Berl)* 23 (3): 147–56. https://doi.org/10.1007/s11818-019-00215-x.

Brown, T. M., G. C. Brainard, C. Cajochen, C. A. Czeisler, J. P. Hanifin, S. W. Lockley, R. J. Lucas, et al. 2022. “Recommendations for Daytime, Evening, and Nighttime Indoor Light Exposure to Best Support Physiology, Sleep, and Wakefulness in Healthy Adults.” Journal Article. *PLoS Biol* 20 (3): e3001571. https://doi.org/10.1371/journal.pbio.3001571.

“CIE System for Metrology of Optical Radiation for ipRGC-Influenced Responses to Light.” 2018, no. CIE S 026/E:2018. https://doi.org/10.25039/S026.2018.

Dahlmann-Noor, A. H., D. Bokre, M. Khazova, and L. L. A. Price. 2025. “Measuring the Visual Environment of Children and Young People at Risk of Myopia: A Scoping Review.” Journal Article. *Graefes Arch Clin Exp Ophthalmol*. https://doi.org/10.1007/s00417-024-06719-z.

Gibaldi, A., E. N. Harb, C. F. Wildsoet, and M. S. Banks. 2024. “A Child-Friendly Wearable Device for Quantifying Environmental Risk Factors for Myopia.” Journal Article. *Transl Vis Sci Technol* 13 (10): 28. https://doi.org/10.1167/tvst.13.10.28.

Guidolin, Carolina, Johannes Zauner, Steffen Lutz Hartmeyer, and Manuel Spitschan. 2024. “Collecting, Detecting and Handling Non-Wear Intervals in

Longitudinal Light Exposure Data." *bioRxiv*. https://doi.org/10.1101/2024.12.23.627604.

Hartmeyer, S. L., and M. Andersen. 2023. "Towards a Framework for Light-Dosimetry Studies: Quantification Metrics." Journal Article. *Lighting Research & Technology* 56 (4): 337–65. https://doi.org/10.1177/14771535231170500.

Hartmeyer, S. L., F. S. Webler, and M. Andersen. 2022. "Towards a Framework for Light-Dosimetry Studies: Methodological Considerations." Journal Article. *Lighting Research & Technology* 55 (4-5): 377–99. https://doi.org/10.1177/14771535221103258.

Hönekopp, A., and S. Weigelt. 2023. "Using Light Meters to Investigate the Light-Myopia Association - a Literature Review of Devices and Research Methods." Journal Article. *Clin Ophthalmol* 17: 2737–60. https://doi.org/10.2147/OPTH.S420631.

Mohamed, A., V. Kalavally, S. W. Cain, A. J. K. Phillips, E. M. McGlashan, and C. P. Tan. 2021. "Wearable Light Spectral Sensor Optimized for Measuring Daily Alpha-Opic Light Exposure." Journal Article. *Opt Express* 29 (17): 27612–27. https://doi.org/10.1364/OE.431373.

Murukesu, Resshaya Roobini, Johannes Zauner, and Manuel Spitschan. 2025. "A Day in Daylight," November. https://doi.org/10.5281/zenodo.17513996.

Okudaira, N., D. F. Kripke, and J. B. Webster. 1983. "Naturalistic Studies of Human Light Exposure." Journal Article. *Am J Physiol* 245 (4): R613–5. https://doi.org/10.1152/ajpregu.1983.245.4.R613.

Patterson Gentile, Carlyn, Ryan Shah, Blanca Marquez De Prado, Nichelle Raj, Christina L. Szperka, Andrew D. Hershey, and Geoffrey K. Aguirre. 2025. "Daily Light Exposure Habits of Youth with Migraine: A Prospective Exploratory Study." *Npj Biological Timing and Sleep* 2 (1). https://doi.org/10.1038/s44323-025-00056-y.

Sah, Raman Prasad, Pavan Kalyan Narra, and Lisa A. Ostrin. 2025. "A Novel Wearable Sensor for Objective Measurement of Distance and Illumination." *Ophthalmic and Physiological Optics* 00 (n/a): 1–13. https://doi.org/https://doi.org/10.1111/opo.13523.

Sullivan, David, Aaron Nicholls, George Hatoun, Samuel Thompson, Cory Schwarzmiller, Fathollah Memarzanjany, Alyssa Gunderson, et al. 2024. "The Visual Experience Evaluation Tool: A Myopia Research Instrument for Quantifying Visual Experience." *bioRxiv*. https://doi.org/10.1101/2024.09.20.614212.

Tabandeh, Niloufar, and Manuel Spitschan. 2025. "Photoreceptor-Specific Scene Statistics Reveal Melanopic Structure in Natural Environments," November. http://dx.doi.org/10.1101/2025.11.10.687567.

van Duijnhoven, J., S. L. Hartmeyer, A. Didikoglu, O. Stefani, K. W. Houser, V. Kalavally, and M. Spitschan. 2025. "Measuring Light Exposure in Daily Life: A Review of Wearable Light Loggers." Journal Article. *Build Environ* 274. https://doi.org/10.1016/j.buildenv.2025.112771.

Webler, F. S., M. Spitschan, R. G. Foster, M. Andersen, and S. N. Peirson. 2019. "What Is the 'Spectral Diet' of Humans?" Journal Article. *Curr Opin Behav Sci* 30: 80–86. https://doi.org/10.1016/j.cobeha.2019.06.006.

Wen, Longbo, Yingpin Cao, Qian Cheng, Xiaoning Li, Lun Pan, Lei Li, HaoGang Zhu, Weizhong Lan, and Zhikuan Yang. 2020. “Objectively Measured Near Work, Outdoor Exposure and Myopia in Children.” *British Journal of Ophthalmology* 104 (11): 1542–47. https://doi.org/10.1136/bjophthalmol-2019-315258.

Wen, Longbo, Qian Cheng, Yingpin Cao, Xiaoning Li, Lun Pan, Lei Li, Haogang Zhu, Ian Mogran, Weizhong Lan, and Zhikuan Yang. 2021. “The Clouclip, a Wearable Device for Measuring Near-Work and Outdoor Time: Validation and Comparison of Objective Measures with Questionnaire Estimates.” *Acta Ophthalmologica* 99 (7): e1222–35. https://doi.org/https://doi.org/10.1111/aos.14785.

Wen, Longbo, Qian Cheng, Weizhong Lan, Yingpin Cao, Xiaoning Li, Yiqiu Lu, Zhenghua Lin, Lun Pan, Haogang Zhu, and Zhikuan Yang. 2019. “An Objective Comparison of Light Intensity and Near-Visual Tasks Between Rural and Urban School Children in China by a Wearable Device Clouclip.” *Translational Vision Science & Technology* 8 (6): 15–15. https://doi.org/10.1167/tvst.8.6.15.

Williams, Rachel, Suyash Bakshi, Edwin J Ostrin, and Lisa A Ostrin. 2019. “Continuous Objective Assessment of Near Work.” *Scientific Reports* 9 (1): 6901.

Zauner, J., C. Guidolin, and M. Spitschan. 2025. “How to Deal with Darkness: Modeling and Visualization of Zero-Inflated Personal Light Exposure Data on a Logarithmic Scale.” *Journal of Biological Rhythms* 0 (0): 07487304251336624. https://doi.org/10.1177/07487304251336624.

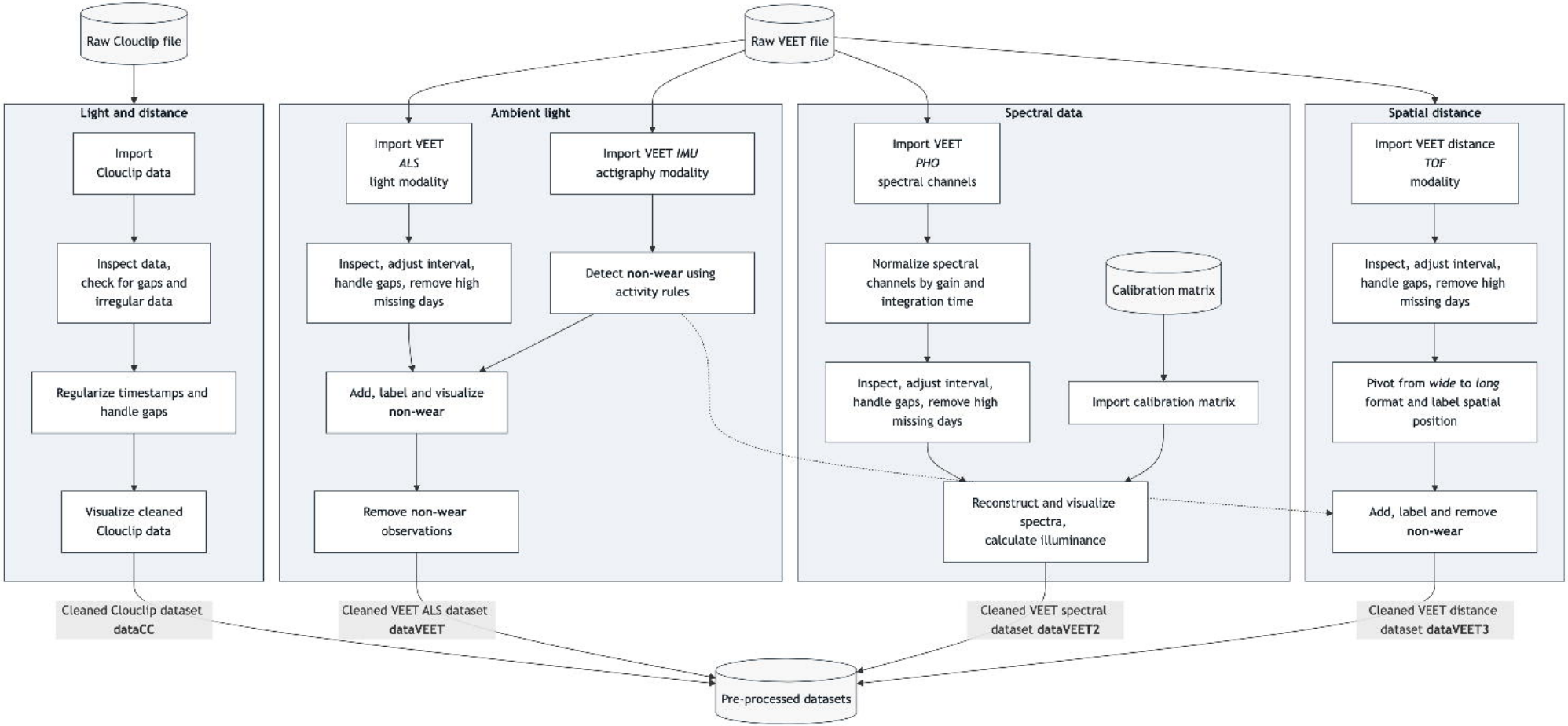
Zauner, J., S. Hartmeyer, and M. Spitschan. 2025. “LightLogR: Reproducible Analysis of Personal Light Exposure Data.” Journal Article. *J Open Source Softw* 10 (107): 7601. https://doi.org/10.21105/joss.07601.

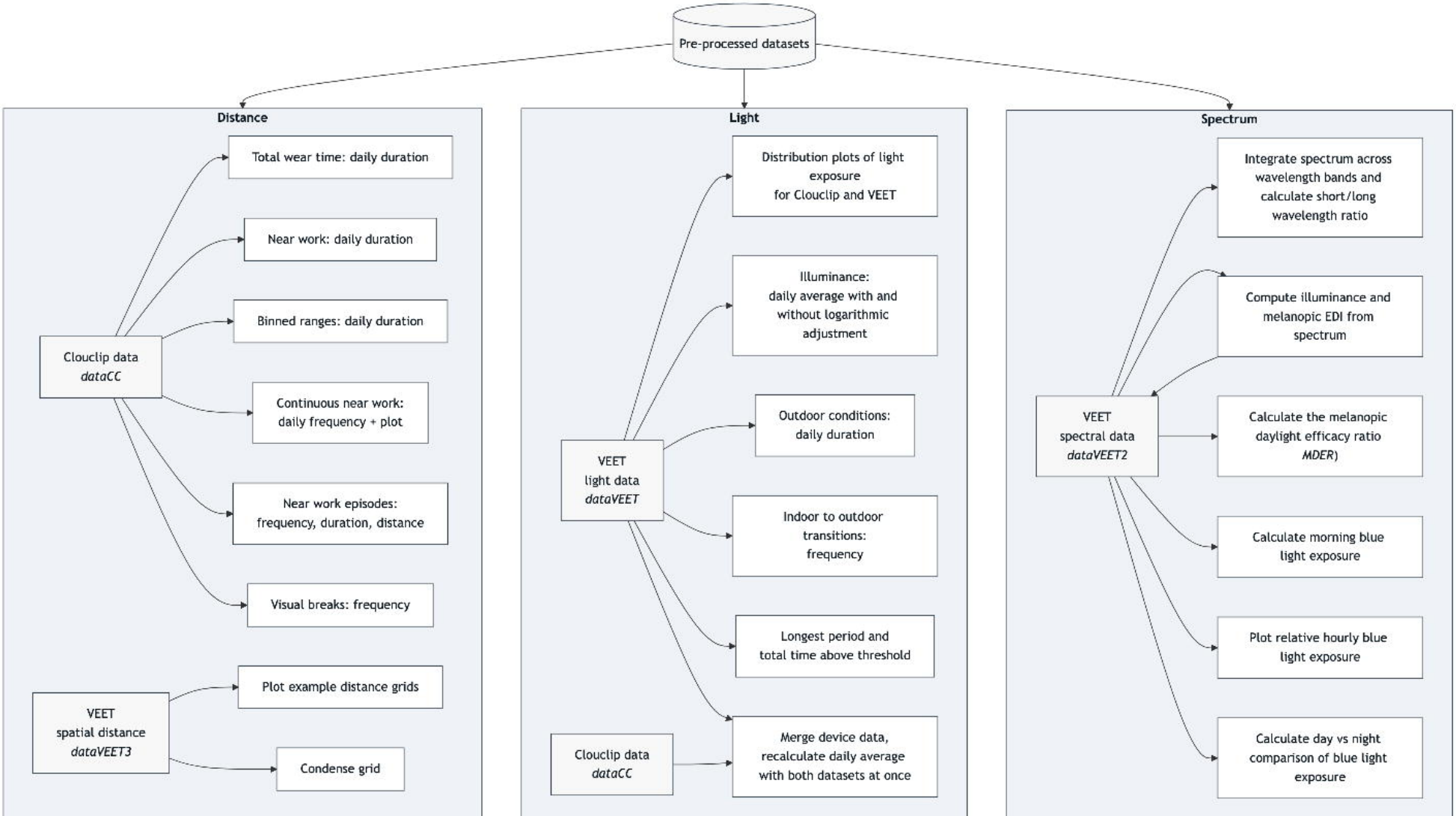
Zauner, Johannes, Oliver Stefani, Anna M. Biller, Carolina Guidolin, and Manuel Spitschan. 2025a. “A Web-Based Specification Tool for Wearable Light Loggers and Optical Radiation Dosimeters.” *Preprints*. https://doi.org/10.20944/preprints202511.0644.v1.

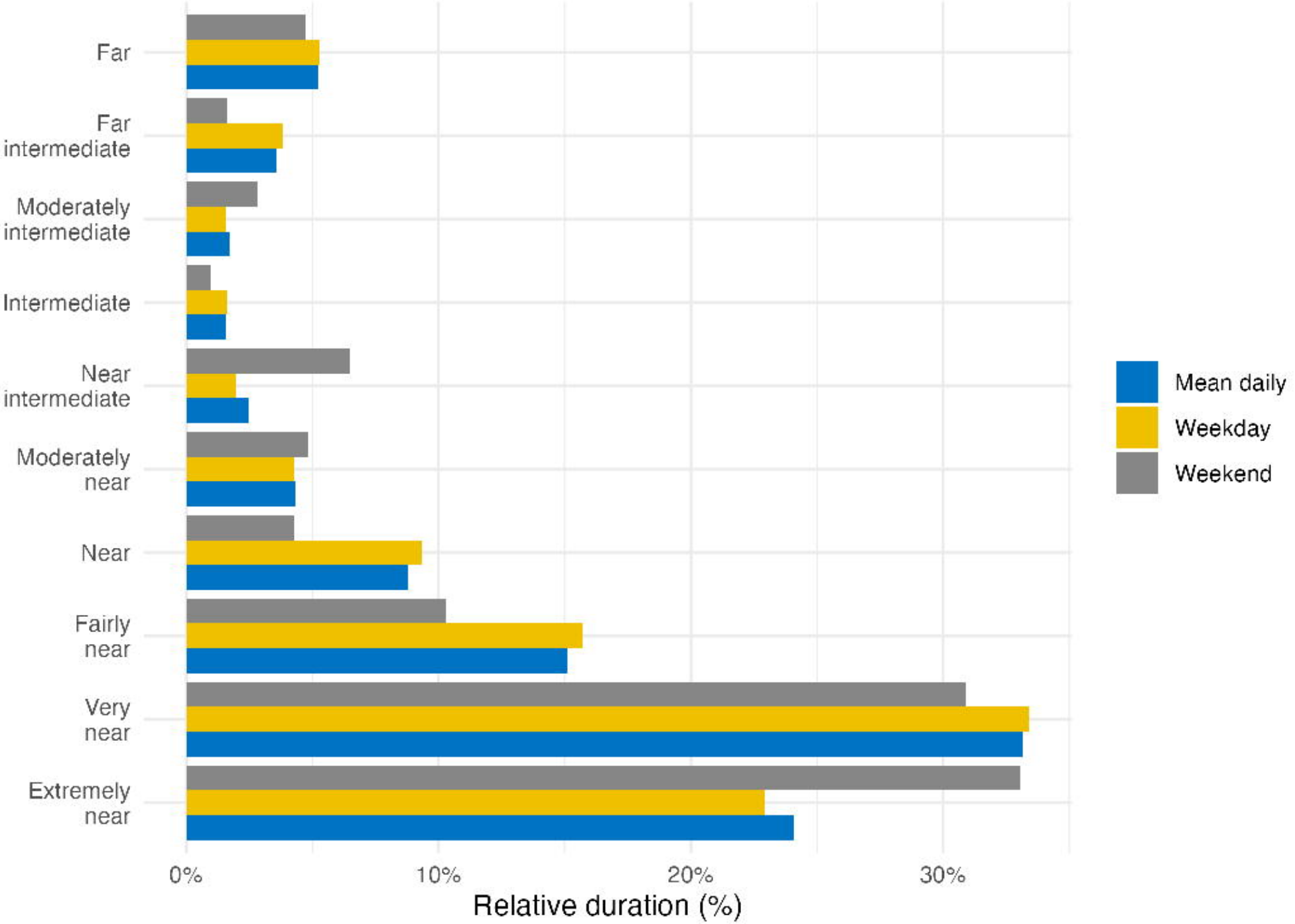
Zauner, J., O. Stefani, G. Bocanegra, C. Guidolin, B. Schrader, L. Udovicic, and M. Spitschan. 2025b. “Auxiliary Data, Quality Assurance and Quality Control for Wearable Light Loggers and Optical Radiation Dosimeters.” *bioRxiv*. https://doi.org/10.1101/2025.09.11.675633.

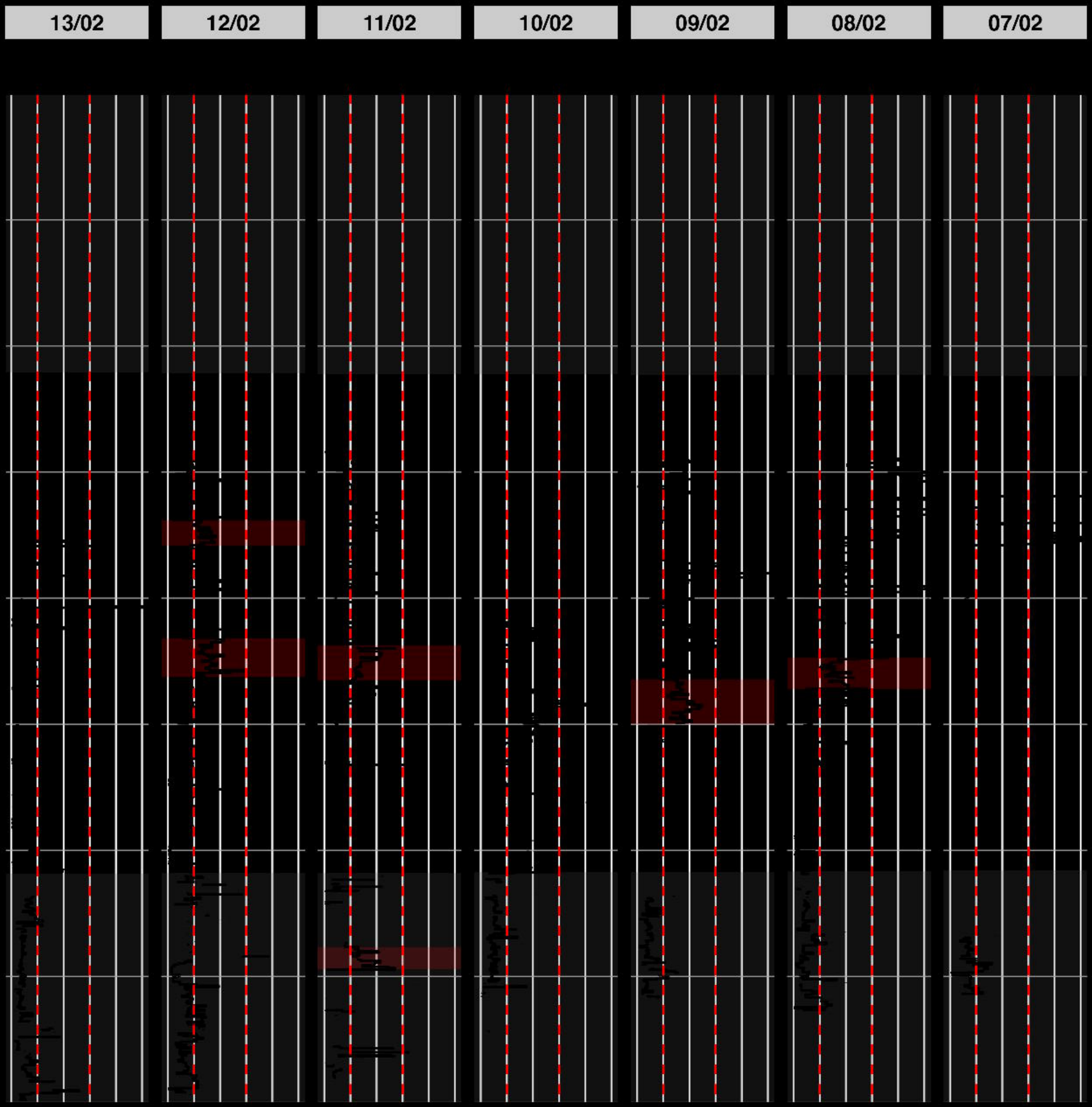
Zauner, J., L. Udovicic, and M. Spitschan. 2024. “Power Analysis for Personal Light Exposure Measurements and Interventions.” *PLOS ONE* 19 (12): 1–15. https://doi.org/10.1371/journal.pone.0308768.

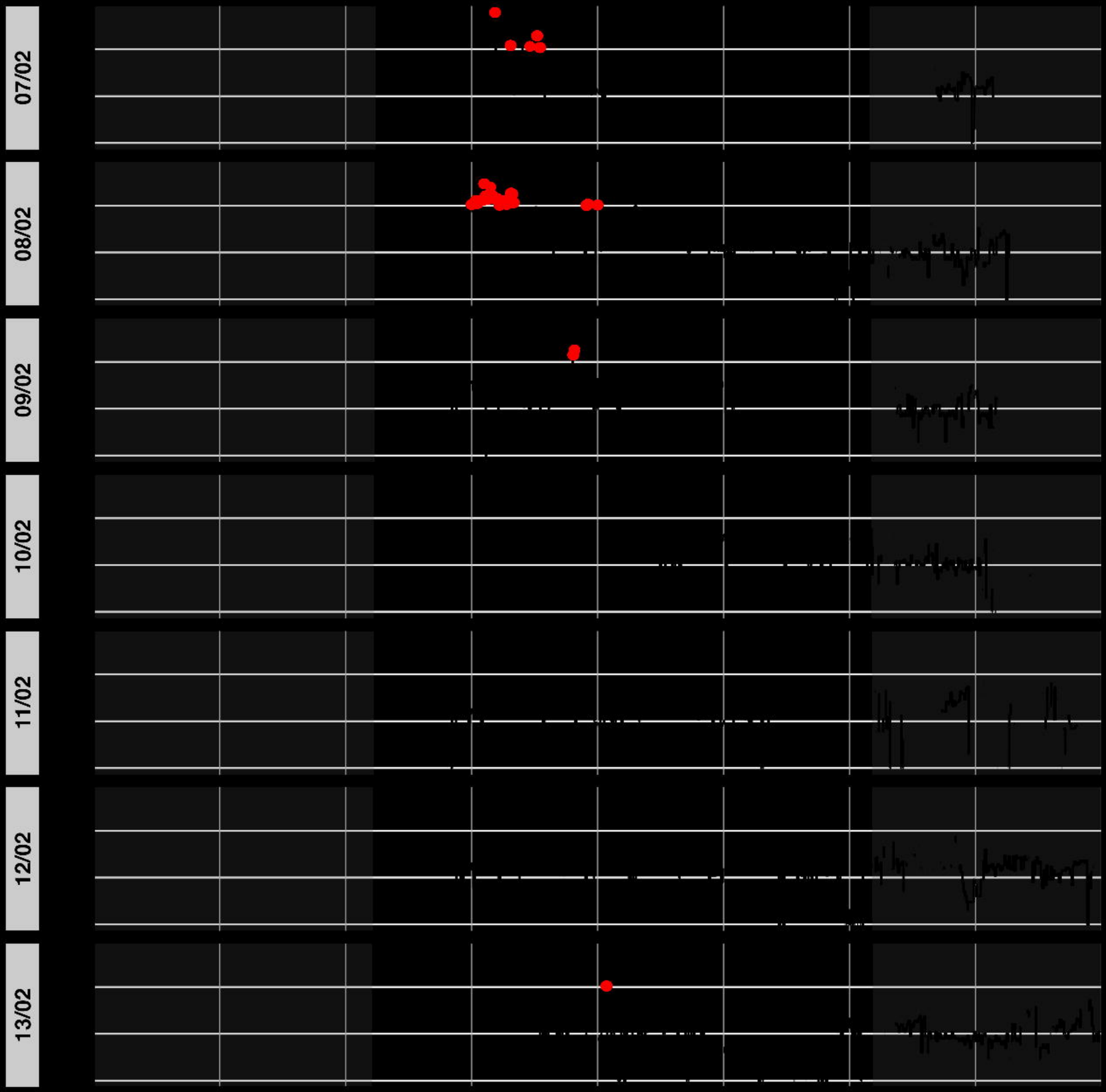
Zauner, Johannes, Rigmor C. Baraas, Elise N. Harb, Ali Heshmati, Francisco Imai, Pauline Kang, Raymond P. Najjar, Aaron Nicholls, Lisa Ostrin, Erik Page, and Manuel Spitschan. 2025c. “Technical Guide for Wearable Optical Radiation Dosimetry and Visual Experience Assessment.” https://doi.org/10.17617/6XCA-CG59.

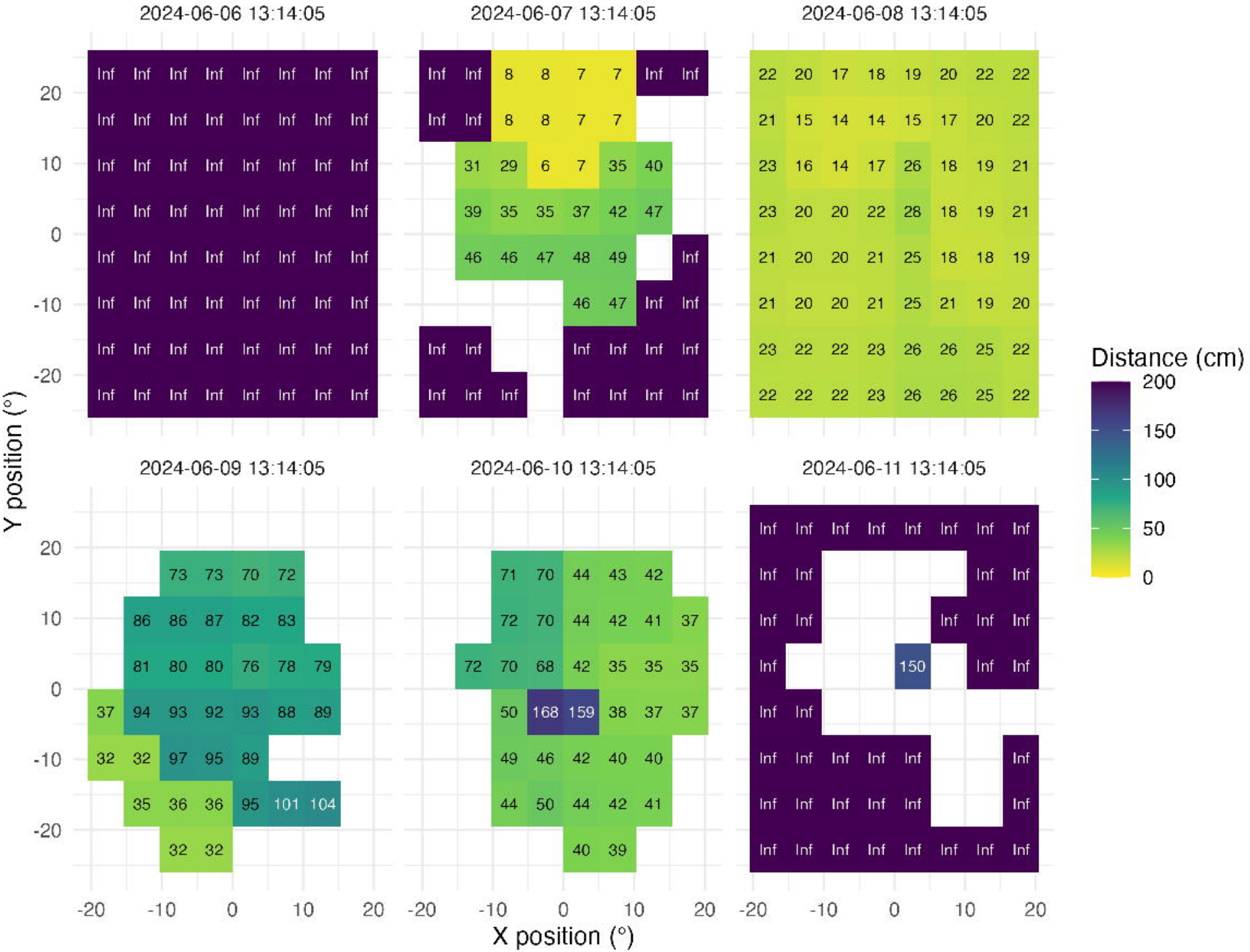

Raw Clouclip file
Raw VEET file
Light and distance
Import Clouclip data
Inspect data, check for gaps and irregular data
Regularize timestamps and handle gaps
Visualize cleaned Clouclip data
Ambient light
Import VEET *ALS* light modality
Import VEET *IMU* actigraphy modality
Inspect, adjust interval, handle gaps, remove high missing days
Detect **non-wear** using activity rules
Add, label and visualize **non-wear**
Remove **non-wear** observations
Spectral data
Import VEET *PHO* spectral channels
Normalize spectral channels by gain and integration time
Calibration matrix
Inspect, adjust interval, handle gaps, remove high missing days
Import calibration matrix
Reconstruct and visualize spectra, calculate illuminance
Spatial distance
Import VEET distance *TOF* modality
Inspect, adjust interval, handle gaps, remove high missing days
Pivot from *wide* to *long* format and label spatial position
Add, label and remove **non-wear**
Cleaned Clouclip dataset **dataCC**
Cleaned VEET ALS dataset **dataVEET**
Cleaned VEET spectral dataset **dataVEET2**
Cleaned VEET distance dataset **dataVEET3**
Pre-processed datasets


Pre-processed datasets
Distance
Clouclip data
dataCC
Total wear time: daily duration
Near work: daily duration
Binned ranges: daily duration
Continuous near work:
daily frequency + plot
Near work episodes:
frequency, duration, distance
Visual breaks: frequency
VEET
spatial distance
dataVEET3
Plot example distance grids
Condense grid
Light
VEET
light data
dataVEET
Distribution plots of light
exposure
for Clouclip and VEET
Illuminance:
daily average with and
without logarithmic
adjustment
Outdoor conditions:
daily duration
Indoor to outdoor
transitions:
frequency
Longest period and
total time above threshold
Clouclip data
dataCC
Merge device data,
recalculate daily average
with both datasets at once
Spectrum
VEET
spectral data
dataVEET2
Integrate spectrum across
wavelength bands and
calculate short/long
wavelength ratio
Compute illuminance and
melanopic EDI from
spectrum
Calculate the melanopic
daylight efficacy ratio
MDER)
Calculate morning blue
light exposure
Plot relative hourly blue
light exposure
Calculate day vs night
comparison of blue light
exposure

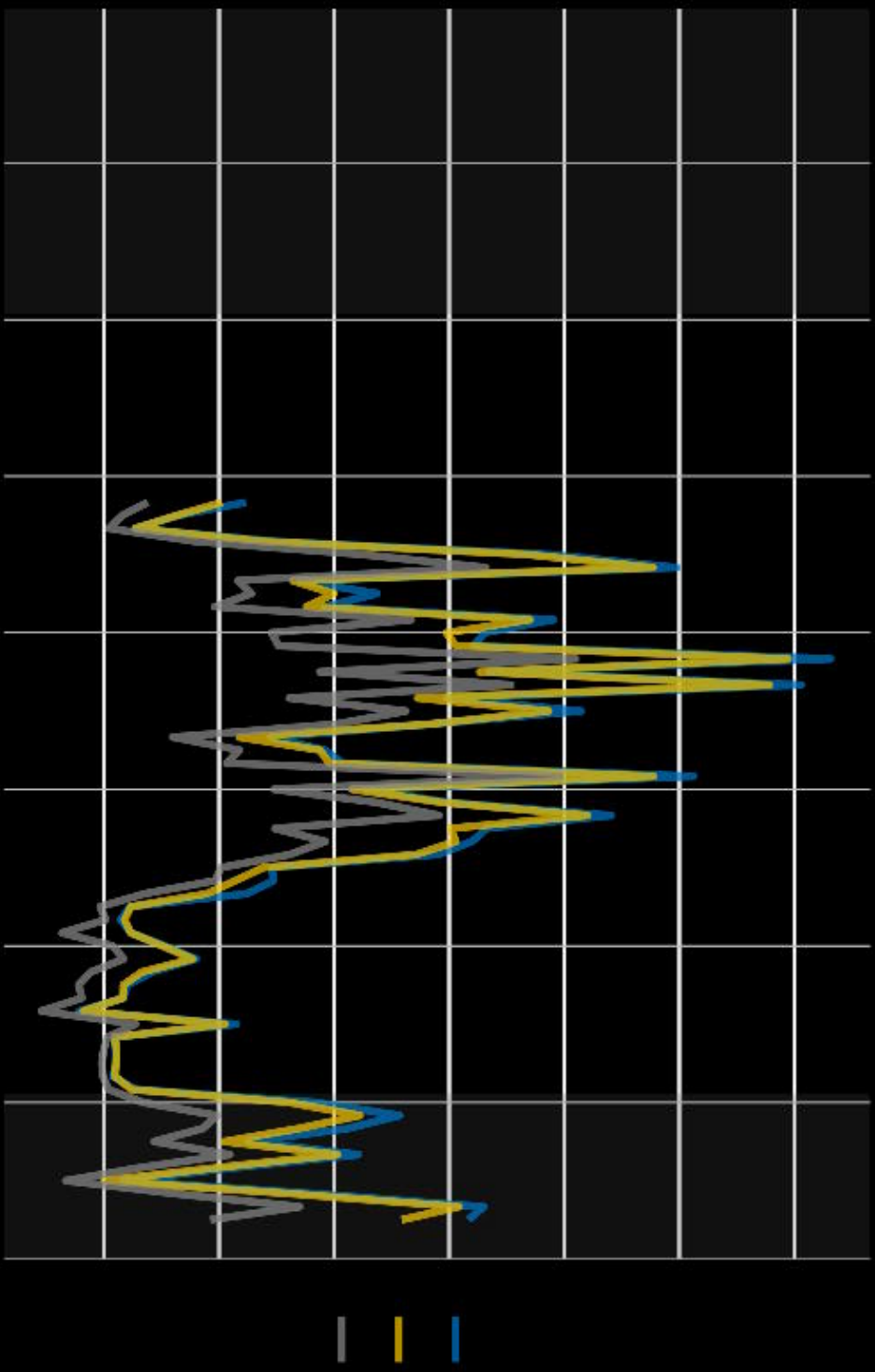

Far
Far intermediate
Moderately intermediate
Intermediate
Near intermediate
Moderately near
Near
Fairly near
Very near
Extremely near
0%
10%
20%
30%
Relative duration (%)
Mean daily
Weekday
Weekend

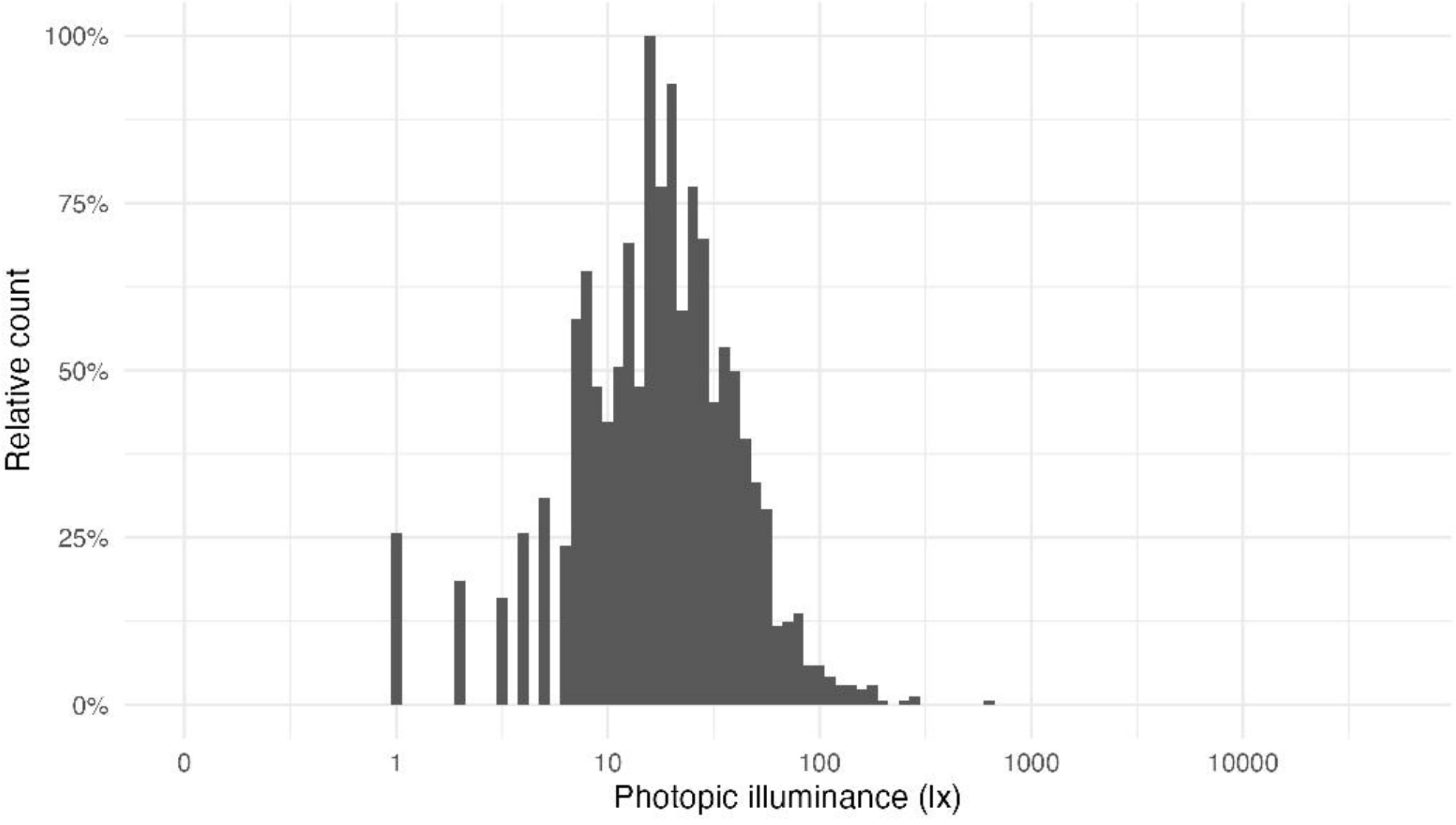

13/02
12/02
11/02
10/02
09/02
08/02
07/02

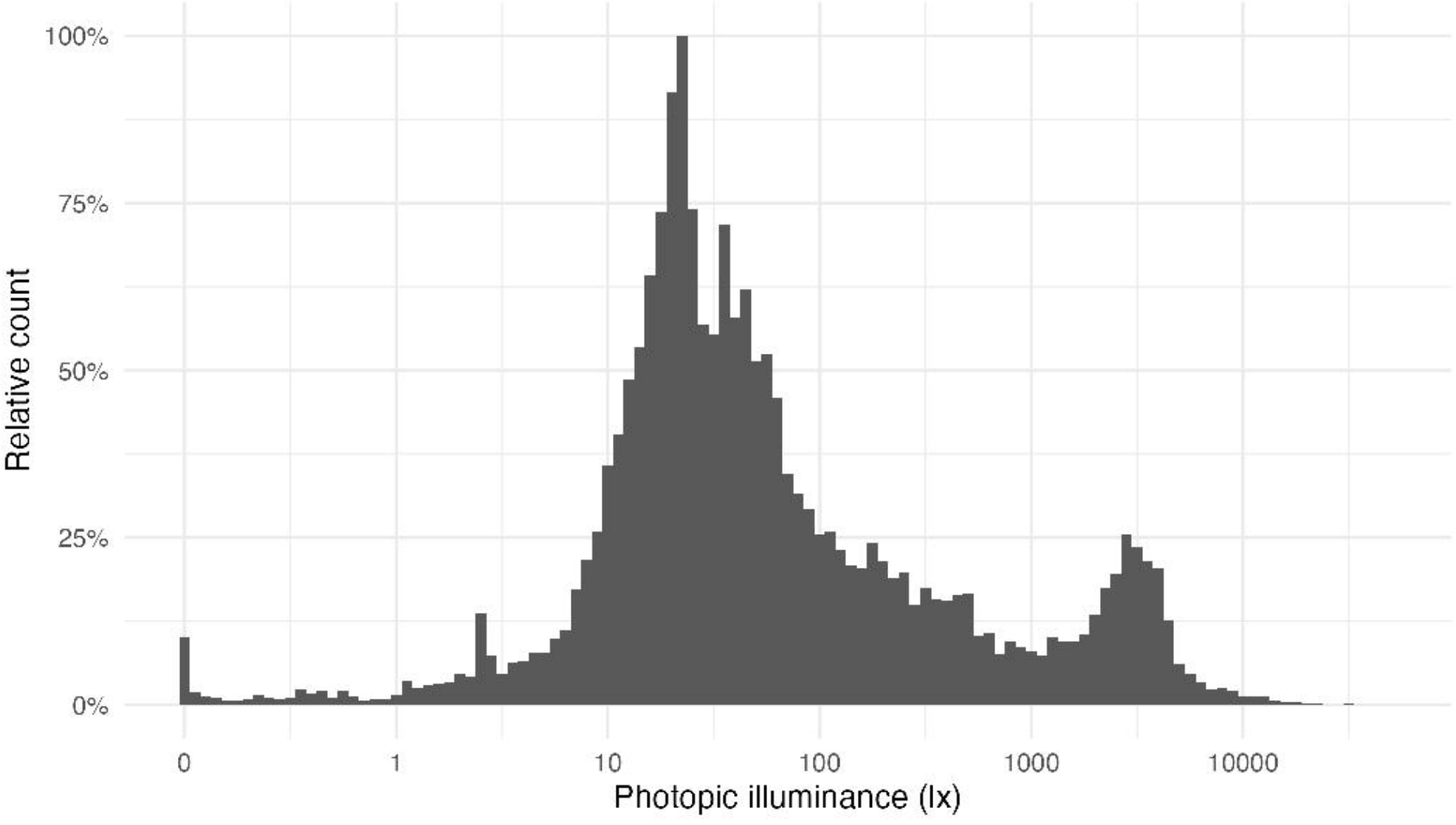

07/02
08/02
09/02
10/02
11/02
12/02
13/02

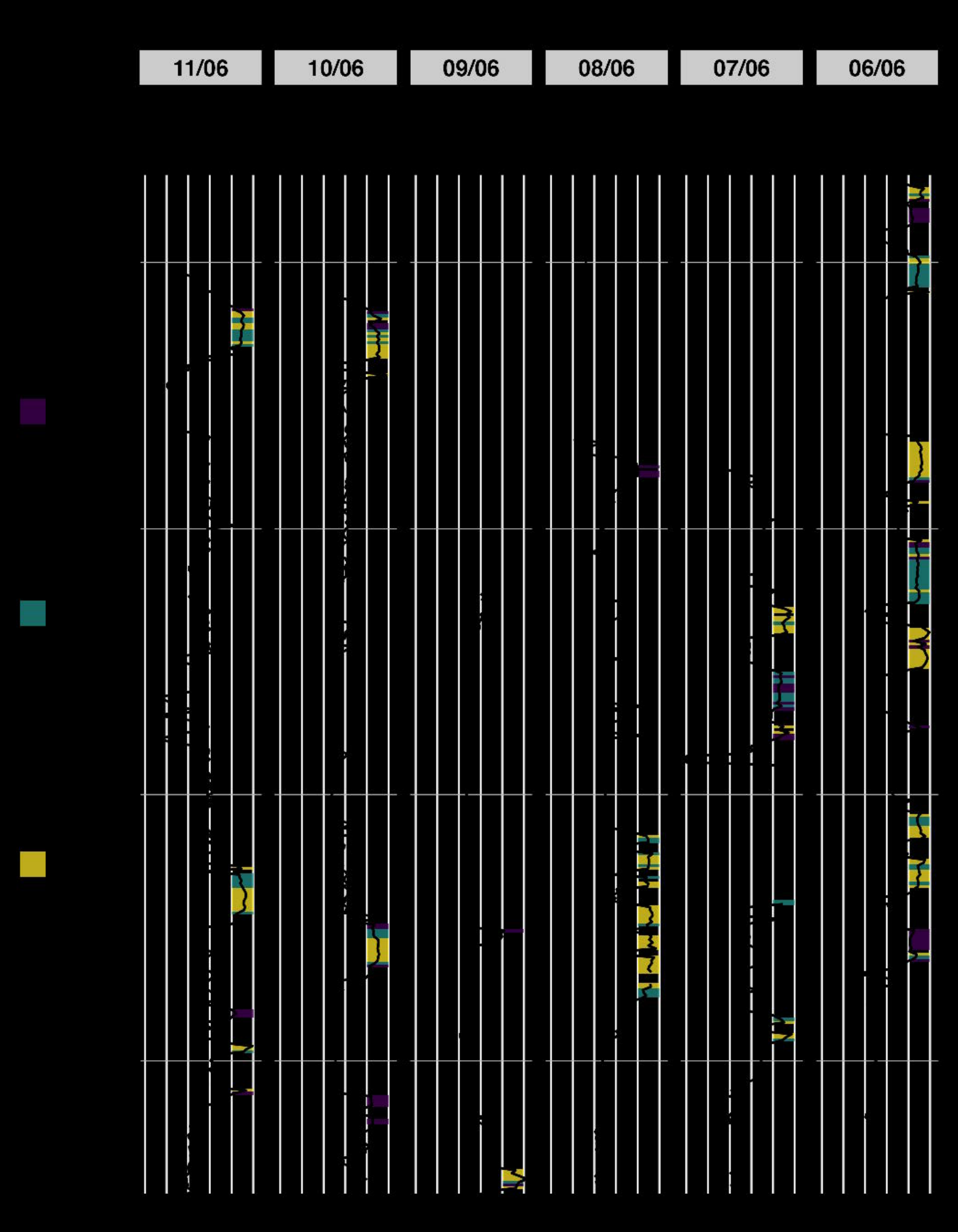

2024-06-06 13:14:05
2024-06-07 13:14:05
2024-06-08 13:14:05
2024-06-09 13:14:05
2024-06-10 13:14:05
2024-06-11 13:14:05
Y position (°)
X position (°)
Distance (cm)
200
150
100
50
0

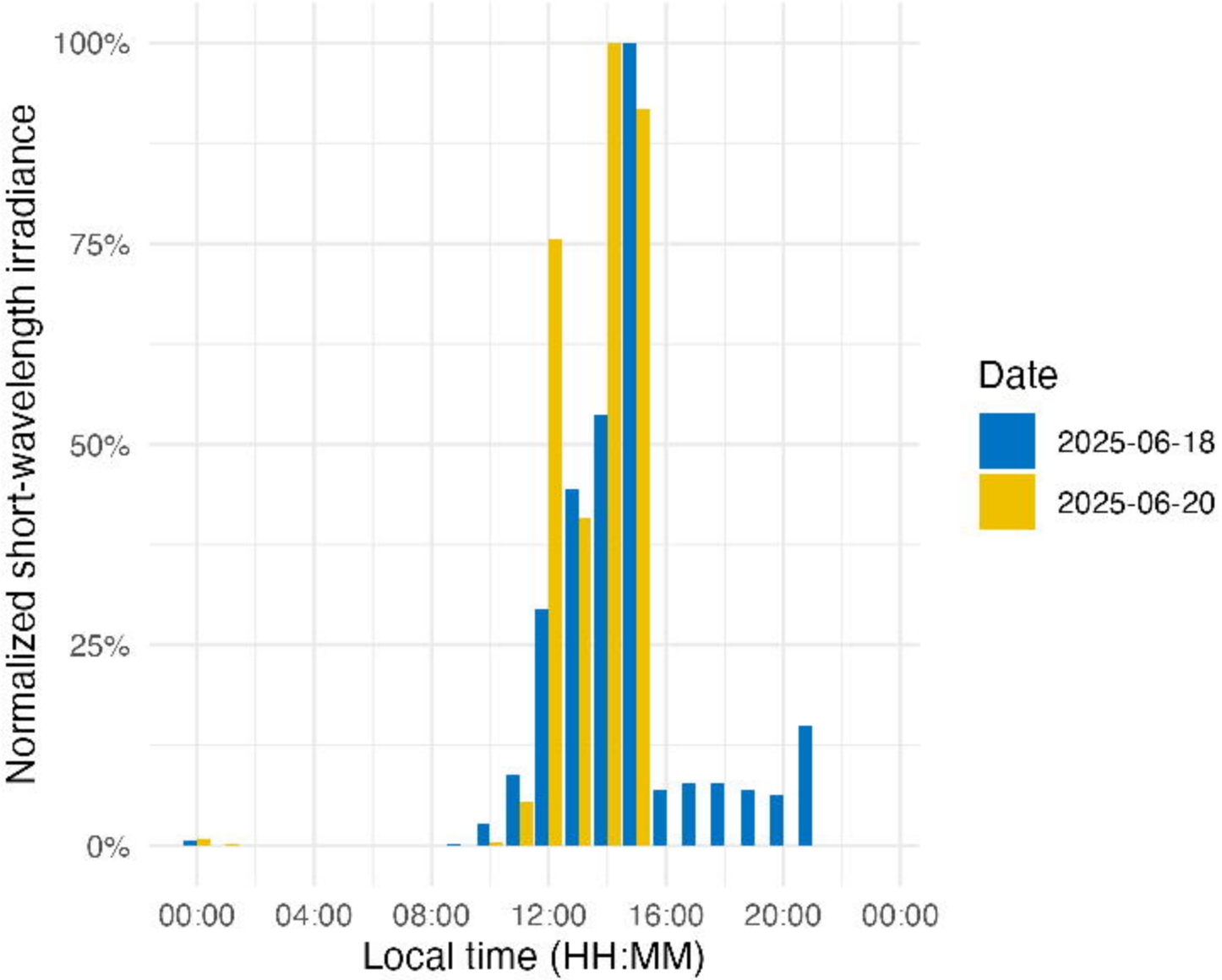
10/06




100%
75%
50%
25%
0%
Relative count
0
1
10
100
1000
10000
Photopic illuminance (lx)


100%
75%
50%
25%
0%
Relative count
0
1
10
100
1000
10000
Photopic illuminance (lx)


11/06
10/06
09/06
08/06
07/06
06/06


100%
75%
50%
25%
0%
Normalized short-wavelength irradiance
00:00
04:00
08:00
12:00
16:00
20:00
00:00
Local time (HH:MM)
Date
2025-06-18
2025-06-20