CODECHECK certificate 2026-019

https://doi.org/10.5281/zenodo.21238767

Table 1: CODECHECK summary

Item	Value
Title	*Photoreceptor-specific scene statistics reveal melanopic structure in natural environments*
Author(s)	Niloufar Tabandeh, Manuel Spitschan
Reference	https://doi.org/10.1016/j.isci.2026.116661
Repository	https://github.com/tscnlab/TabandehEtAl_iScience_2026
Codechecker(s)	Jan Haacker
Date of check	2026-07-07
Summary	The checked repository includes code to reproduce all figures shown in the associated manuscript titled "Photoreceptor-specific scene statistics reveal melanopic structure in natural environments" including one supplementary figure. The workflow is instructed by a detailed recipe in the file **`README.md`** and structured into three main parts and small alignment steps. The main parts are downloading the input data and running two notebooks consecutively. Following the instructions successfully produces the manuscript and supplement figures; however, in Figures 1 to 4 structural changes can occur, probably due to a non-deterministic order when processing input files. The code is entirely written in Python allowing to understand the mechanics by reading and uses a transparent repository structure.

Table 2: Summary of output files generated

File	Comment	Size (b)
`merged_wp_jeti_metadata_df.csv`	Merged WP, JETI, and metadata	949043
`average_pixels_correlation_df.csv`	Pixel-level alpha-opic correlations	3865
`amplitude_spectra_regression_view_df.csv`	Amplitude spectra regressions by view	67183
`jeti_statistics_overall.csv`	Overall JETI central statistics	712
`jeti_statistics_per_view.csv`	JETI central statistics by view	2843
`wp_statistics_overall.csv`	Overall WP central statistics	1052
`wp_statistics_per_view.csv`	WP central statistics by view	2662
`overexposed_percentage_per_view.csv`	Overexposed pixels by view	236

File	Comment	Size (b)
`broken_bar_plot_mEDI_logscale_location.png`	Figure 1 component, mEDI by location	430489
`Overexposed_pixels_per_view.png`	Figure 1 component, overexposed pixels	195797
`Figure1.png`	Figure 1	740872
`Figure1.pdf`	Figure 1	1523661
`MelEDI_vr_illuminance_logscale_KDE_jeti.png`	Figure 2 component, melanopic EDI vs illuminance	701861
`Wp_vr_jeti_KDE_logscale.png`	Figure 2 component, WP vs JETI	956949
`Correlation_matrix_a-opics_radiance_per_view.png`	Figure 2 component, JETI alpha-opic correlations	467004
`Figure2.png`	Figure 2	1602173
`Figure2.pdf`	Figure 2	3374518
`Correlation_matrix_a-opics_average_radiances_per_view.png`	Figure 3 component, average radiance correlations	727844
`Correlation_matrix_a-opics_pixel_by_pixel_per_view.png`	Figure 3 component, pixel-level correlations	478519
`Figure3.png`	Figure 3	663369
`Figure3.pdf`	Figure 3	1594981
`Violin_plot_rms_contrast_per_view_flip.png`	Figure 4 component, RMS contrast by view	418220
`RMS_vr_luminance_KDE_logscale_WP.png`	Figure 4 component, RMS contrast vs luminance	869423
`Figure4.png`	Figure 4	1012571
`Figure4.pdf`	Figure 4	2161711
`Figure5.png`	Figure 5	1154054
`Figure5.pdf`	Figure 5	510902
`Min_slope.png`	Figure 6 component, low-slope example	4245732
`Max_slope.png`	Figure 6 component, high-slope example	3571378
`Figure6.png`	Figure 6	2466807
`Figure6.pdf`	Figure 6	6148094
`Figure7.png`	Figure 7	95869
`Figure7.pdf`	Figure 7	18288
`Figure_Sup.png`	Supplementary figure	530244
`Figure_sup.pdf`	Supplementary figure	26619

Summary

The checked repository includes code to reproduce all figures shown in the associated manuscript titled "Photoreceptor-specific scene statistics reveal melanopic structure in natural environments" including one supplementary figure. The workflow is instructed by a detailed recipe in the file `README.md` and structured into three main parts and small alignment steps. The main parts are downloading the input data and running two notebooks consecutively. Following the instructions successfully produces the manuscript and supplement figures; however, in Figures 1 to 4 structural changes can occur, probably due to a non-deterministic order when processing input files. The code is entirely written in Python allowing to understand the mechanics by reading and uses a transparent repository structure.

CODECHECKER notes

Environment setup

The checked repository instructs to setup the environment using `conda` with the provided `environment.yml`. ~~However, this requires a `win-64` platform. The list of dependencies as read from the well structured notebooks with all imports at their headers allowed to assemble the following list:~~

- ~~cmcrameri~~
- ~~matplotlib~~
- ~~numpy~~
- ~~pandas~~
- ~~scipy~~
- ~~seaborn~~

The environment was revised to be platform independent. This was smoke-tested by having **`pixi`**[1] successfully resolve the dependencies for the platforms **`linux-64`**, **`osx-64`**, **`osx-arm64`**, and **`win-64`**.

[1] Arts, R., Zalmstra, B., Vollprecht, W., de Jager, T., Morcotilo, N., & Hofer, J. (2026). Pixi [Rust]. https://github.com/prefix-dev/pixi/releases/tag/v0.71.0 (Original work published 2023)

Raw data download

In the next step of the instructions, raw data should be downloaded. While not mentioned, the compressed raw data amount to about 180 Gb which deflate to about 320 Gb. Personally, this forces me to work on a remote machine. I registered at the data repository to obtain an API token which then allowed me to download the data on a headless machine which would otherwise require either GUI forwarding or programmatic session management. The code used for the download is based on Snakemake[2] in **`../workflow`** (at commit f00dd77; the workflow and environment are extended later). The archives contain the lowest-level directories and should be directly extracted into **`dataset_dir/derivatives`** following the instructions naming convention. A provided list of checksums enables verifying the integrity of the downloaded files.

[2] Mölder F, Jablonski KP, Letcher B et al. Sustainable data analysis with Snakemake [version 3; peer review: 2 approved]. F1000Research 2025, 10:33. https://doi.org/10.12688/f1000research.29032.3

Running the notebooks

The instructions helpfully specify that running the notebooks may take several hours. Since I already have a Snakemake workflow for downloading, I run the notebooks through **`ipykernel`**-based **`papermill`**, having set the paths as instructed. Note that the Snakemake rules are implemented just to help me run the notebooks and do nothing more.

The **`WP_derivates.ipynb`** ran smoothly.

`SCENES_spatial_analysis_paper.ipynb`

~~There were a couple of mild problems for which I filed GitHub issues. In essence, these were mismatches in paths, dictionary keys, and letter casing. I patched the immediate blockers to continue the check and conclude that, if the issues get fixed, the notebook will run completely.~~

The **`SCENES_spatial_analysis_paper.ipynb`** ran smoothly after merging the patch provided by the authors.

Recommendations to the authors

Add limitations to README

Users would benefit from an overview of system requirements by knowing what systems can be expected to reproduce the results without further ado. Stating the characteristics of the system used would already fulfill this need. Ideally, narrower requirements constraints could allow allocating just the required resources.

~~Offer a platform-independent environment definition~~

~~While the provided environment is helpful in ensuring the environment can be resolved, the instructions to install the environment can only be followed on a **`win-64`** platform. If possible, offer an alternative that is platform-independent. Else, include the platform dependence in the workflow limitations (see above).~~ *Addressed by the authors and obsolete.*

Manifest files

CSV files

`results/tables/merged_wp_jeti_metadata_df.csv`
Author comment: *Merged WP, JETI, and metadata*

Column summary statistics:

	count	mean	std	min	25%	50%	75%	max
0	671	335.0000	193.8453	0.0000	167.5000	335.0000	502.5000	670.0000

`results/tables/average_pixels_correlation_df.csv`
Author comment: *Pixel-level alpha-opic correlations*

Column summary statistics:

	count	unique	top	freq
0	76	6	L cones	15
1	76	6	L cones	15
2	76	45	1.0	15
3	76	4	Outdoor view	25

`results/tables/amplitude_spectra_regression_view_df.csv`
Author comment: *Amplitude spectra regressions by view*

Column summary statistics:

	count	unique	top	freq
0	363	362	-1.044715336945301	2
1	363	362	-0.9535159056919448	2
2	363	362	1.1851630285153683e-26	2
3	363	362	0.04765517540560919	2
4	363	362	20230210T1400	2
5	116	116	slope__Indoor w/o window view	1
6	116	116	r__value__Indoor w/o window view	1
7	116	116	p__value__Indoor w/o window view	1
8	116	116	std__err__Indoor w/o window view	1
9	116	116	timestamp__Indoor w/o window view	1
10	195	194	-1.0241379552953915	2
11	195	194	-0.986520520756667	2
12	195	194	2.1433887205133286e-39	2
13	195	194	0.024519680365982663	2
14	195	195	timestamp__Indoor w/ window view	1

results/tables/jeti_statistics_overall.csv
Author comment: *Overall JETI central statistics*

Column summary statistics:

	count	unique	top	freq
0	9	9	index	1
1	9	9	overall__mean	1
2	9	9	overall__std	1
3	9	8	2.1	2
4	9	3	0.0	6
5	9	9	overall__max	1
6	9	9	overall__IQR	1

results/tables/jeti_statistics_per_view.csv
Author comment: *JETI central statistics by view*

Column summary statistics:

	count	unique	top	freq
0	4	4	view	1
1	5	5	Correlated Color Temp. (CCT) [K]	1
2	5	5	Correlated Color Temp. (CCT) [K]	1
3	5	5	Correlated Color Temp. (CCT) [K]	1
4	5	5	Correlated Color Temp. (CCT) [K]	1
5	5	5	Correlated Color Temp. (CCT) [K]	1
6	5	5	Correlated Color Temp. (CCT) [K]	1
7	5	5	Elc [W/sqm]	1
8	5	5	Elc [W/sqm]	1
9	5	5	Elc [W/sqm]	1
10	5	5	Elc [W/sqm]	1
11	5	3	0.0	3
12	5	5	Elc [W/sqm]	1
13	5	5	Emc [W/sqm]	1
14	5	5	Emc [W/sqm]	1
15	5	5	Emc [W/sqm]	1
16	5	5	Emc [W/sqm]	1
17	5	3	0.0	3
18	5	5	Emc [W/sqm]	1
19	5	5	Emel [W/sqm]	1
20	5	5	Emel [W/sqm]	1
21	5	5	Emel [W/sqm]	1

	count	unique	top	freq
22	5	5	Emel [W/sqm]	1
23	5	3	0.0	3
24	5	5	Emel [W/sqm]	1
25	5	5	Erh [W/sqm]	1
26	5	5	Erh [W/sqm]	1
27	5	5	Erh [W/sqm]	1
28	5	5	Erh [W/sqm]	1
29	5	3	0.0	3
30	5	5	Erh [W/sqm]	1
31	5	5	Esc [W/sqm]	1
32	5	5	Esc [W/sqm]	1
33	5	5	Esc [W/sqm]	1
34	5	5	Esc [W/sqm]	1
35	5	3	0.0	3
36	5	5	Esc [W/sqm]	1
37	5	5	Illuminance [lx] (CIE1931 2°)	1
38	5	5	Illuminance [lx] (CIE1931 2°)	1
39	5	5	Illuminance [lx] (CIE1931 2°)	1
40	5	5	Illuminance [lx] (CIE1931 2°)	1
41	5	5	Illuminance [lx] (CIE1931 2°)	1
42	5	5	Illuminance [lx] (CIE1931 2°)	1
43	5	5	X v,mel,D65 (MEDI) [lx]	1
44	5	5	X v,mel,D65 (MEDI) [lx]	1
45	5	5	X v,mel,D65 (MEDI) [lx]	1
46	5	5	X v,mel,D65 (MEDI) [lx]	1
47	5	5	X v,mel,D65 (MEDI) [lx]	1
48	5	5	X v,mel,D65 (MEDI) [lx]	1

`results/tables/wp_statistics_overall.csv`

Author comment: *Overall WP central statistics*

Column summary statistics:

	count	mean	std	min	25%	50%	75%	max
0	10	4.5000	3.0277	0.0000	2.2500	4.5000	6.7500	9.0000

results/tables/wp_statistics_per_view.csv

Author comment: *WP central statistics by view*

Column summary statistics:

	count	unique	top	freq
0	16	16	index	1
1	17	17	Indoor w/ window view	1
2	17	17	Indoor w/ window view	1
3	17	17	Indoor w/ window view	1
4	17	17	Indoor w/ window view	1
5	17	17	Indoor w/ window view	1
6	17	17	Indoor w/ window view	1
7	17	16	0.125	2
8	17	17	Indoor w/o window view	1
9	17	17	Indoor w/o window view	1
10	17	14	0.0048	2
11	17	16	0.0061	2
12	17	9	0.0002	4
13	17	17	Outdoor view	1
14	17	17	Outdoor view	1
15	17	17	Outdoor view	1
16	17	17	Outdoor view	1
17	17	17	Outdoor view	1
18	17	16	0.1334	2

results/tables/overexposed_percentage_per_view.csv

Author comment: *Overexposed pixels by view*

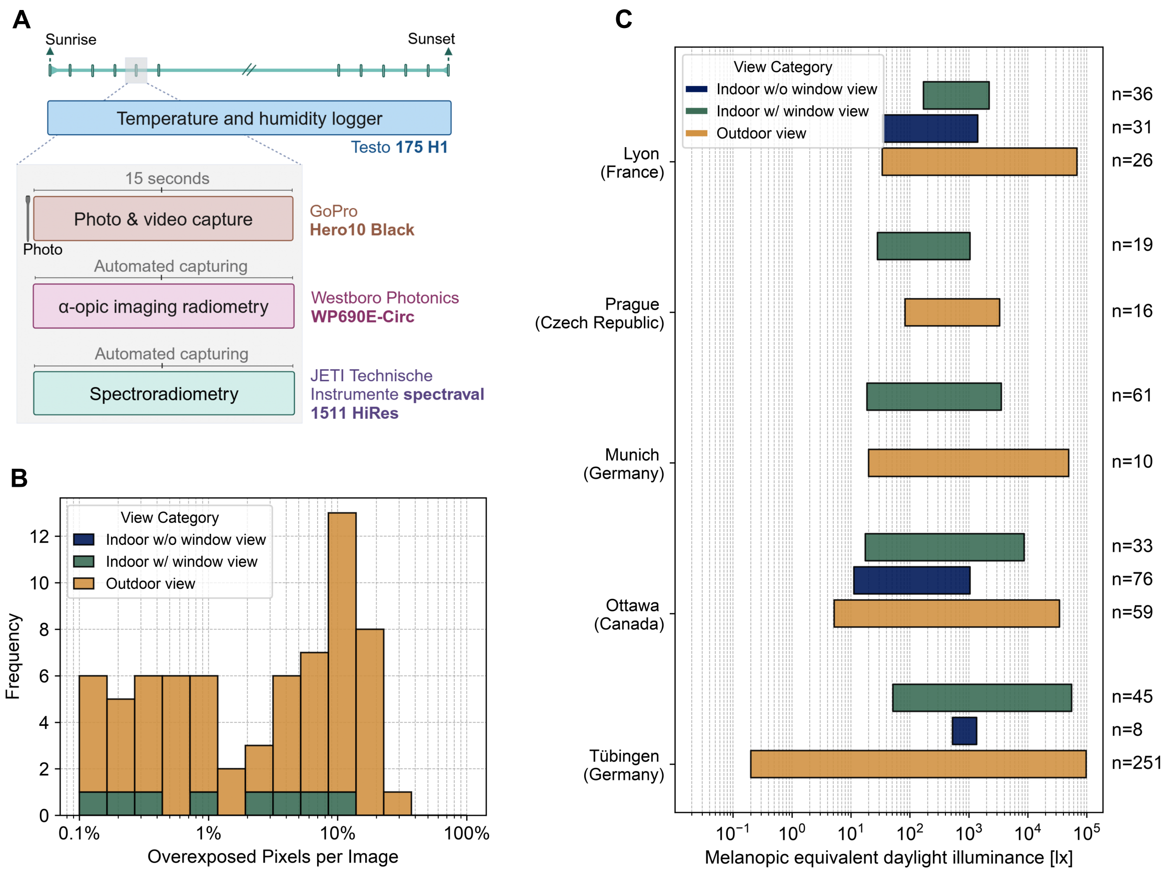
Column summary statistics:

	count	mean	std	min	25%	50%	75%	max
0	3	1.0000	1.0000	0.0000	0.5000	1.0000	1.5000	2.0000

Figures

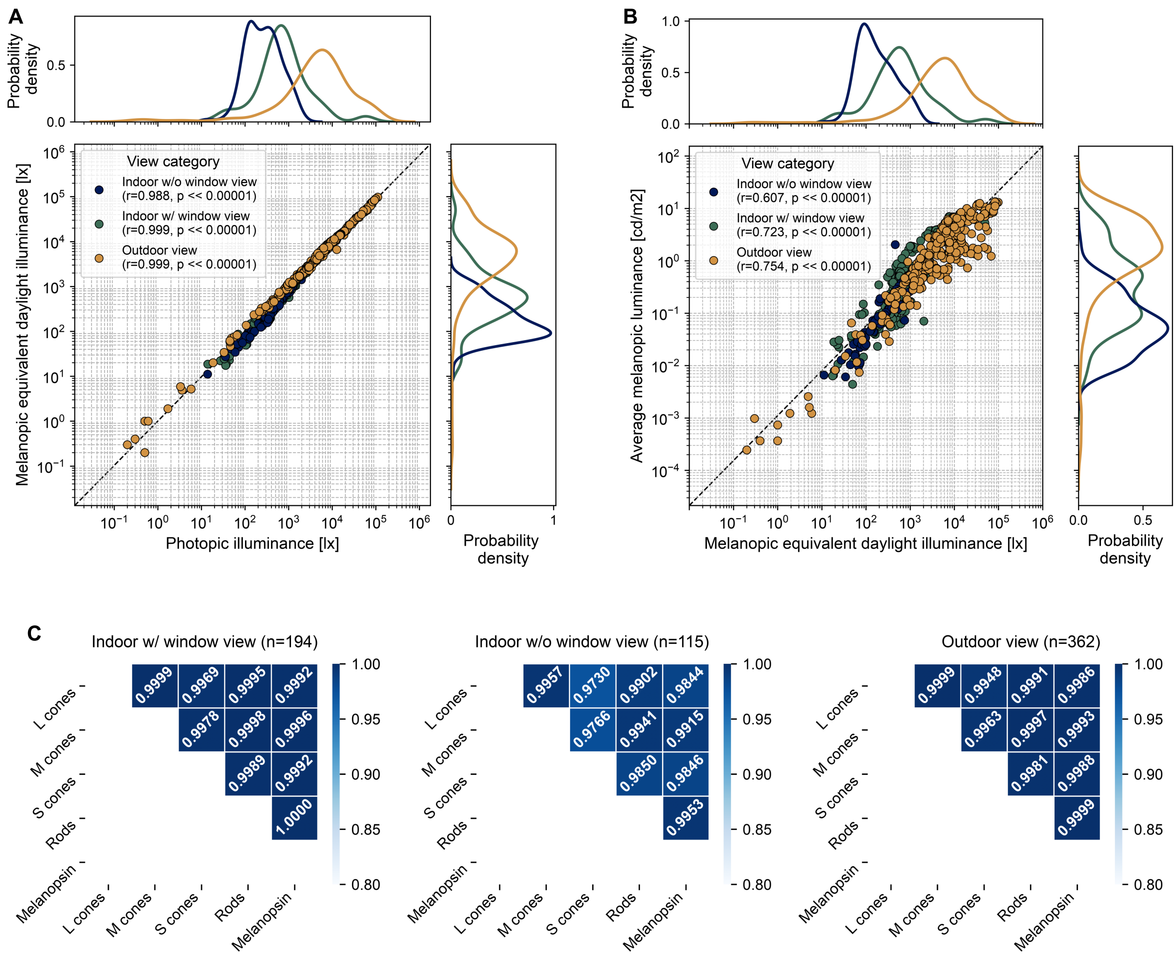
results/figures/Figure1.pdf

Author comment: *Figure 1*



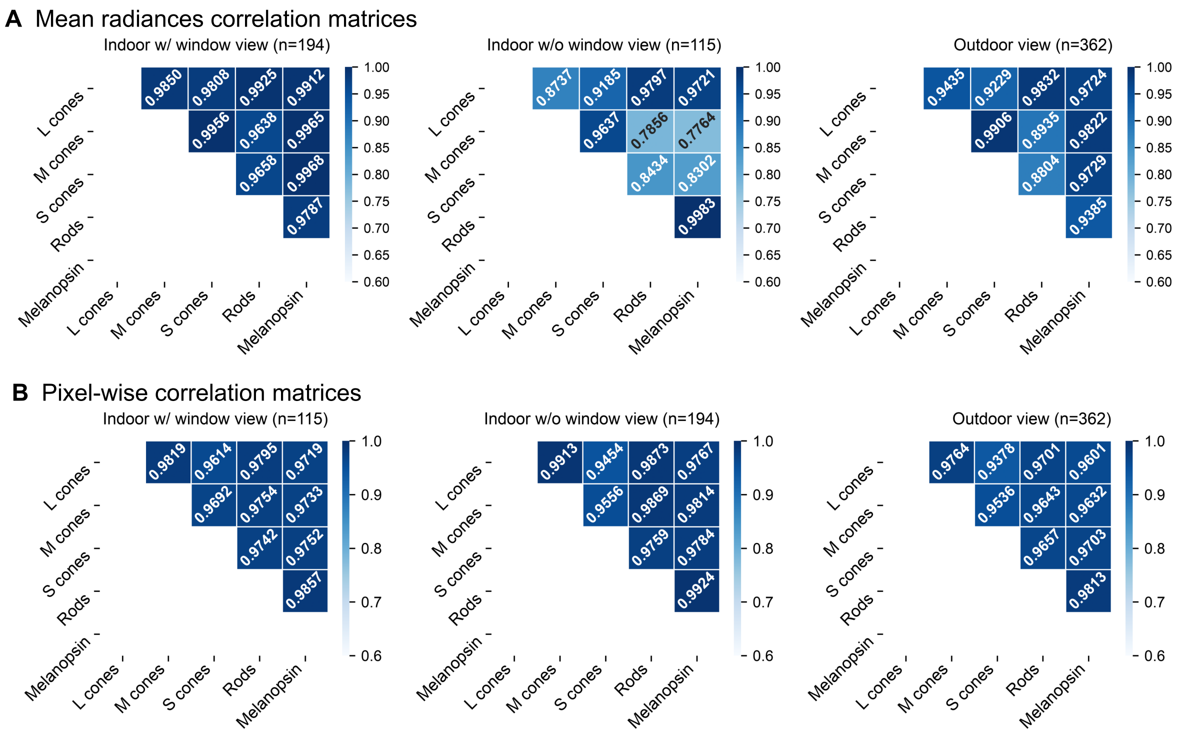
results/figures/Figure2.pdf

Author comment: *Figure 2*



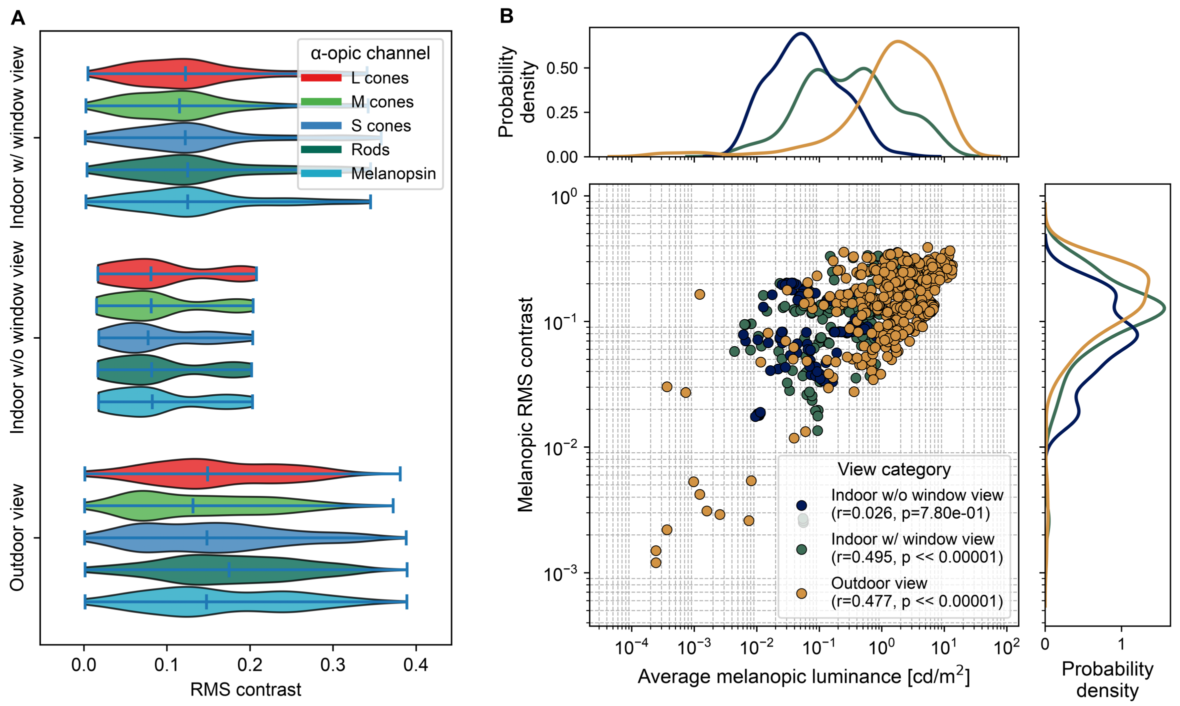
`results/figures/Figure3.pdf`

Author comment: *Figure 3*



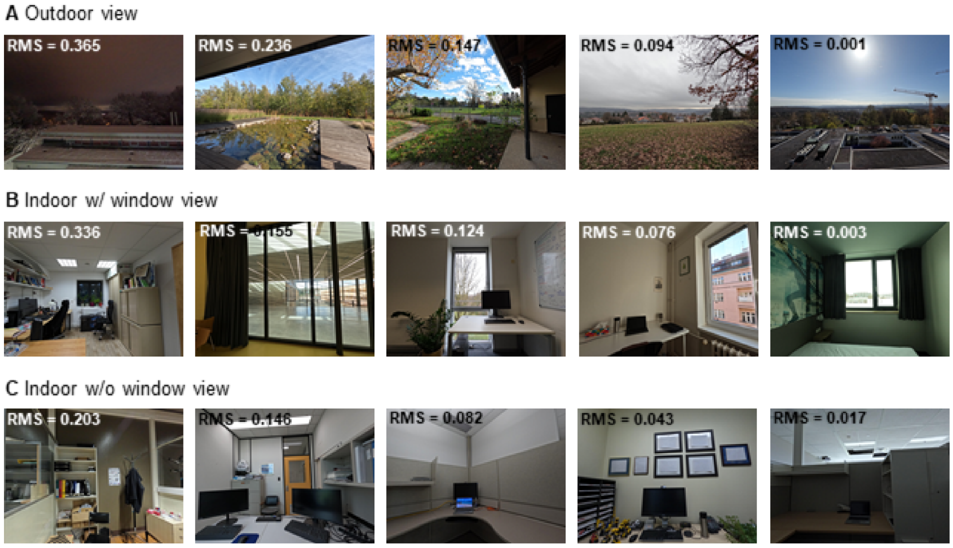
results/figures/Figure4.pdf

Author comment: *Figure 4*



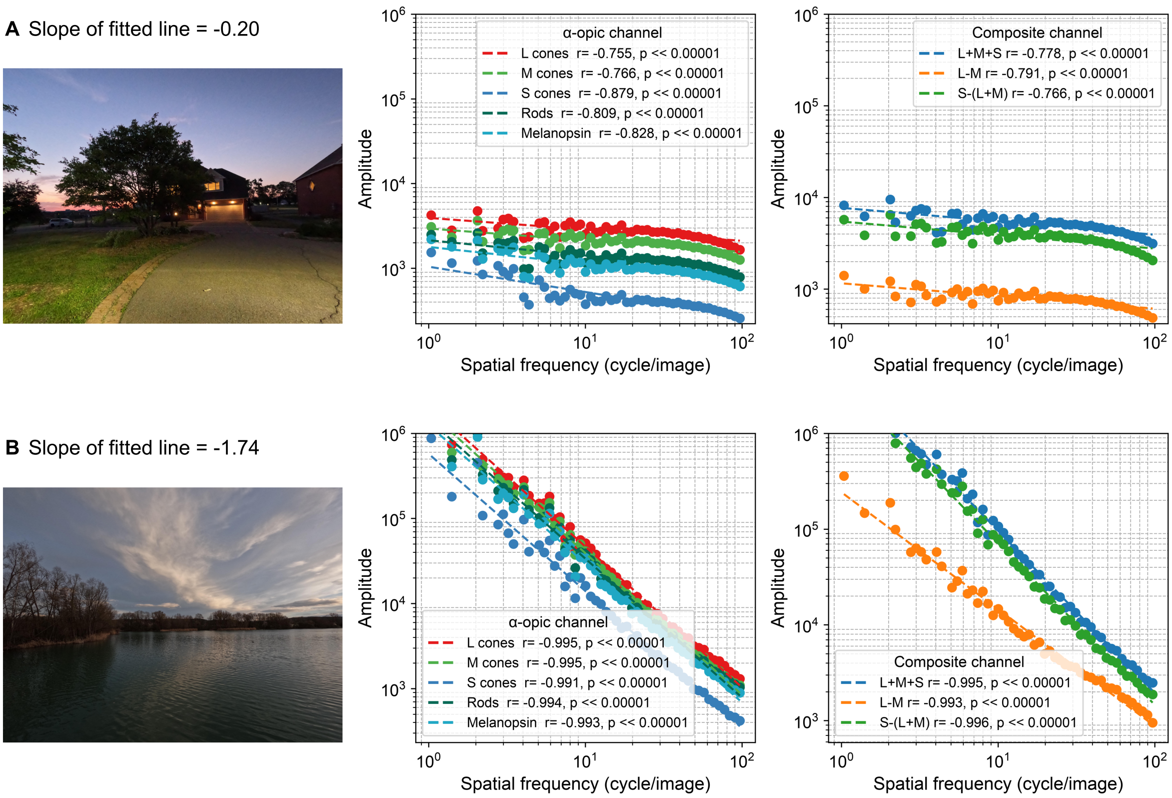
results/figures/Figure5.pdf

Author comment: *Figure 5*



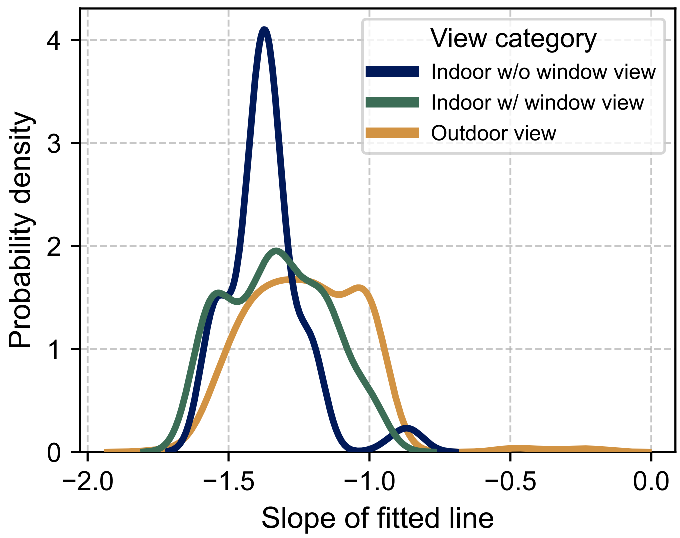
results/figures/Figure6.pdf

Author comment: *Figure 6*



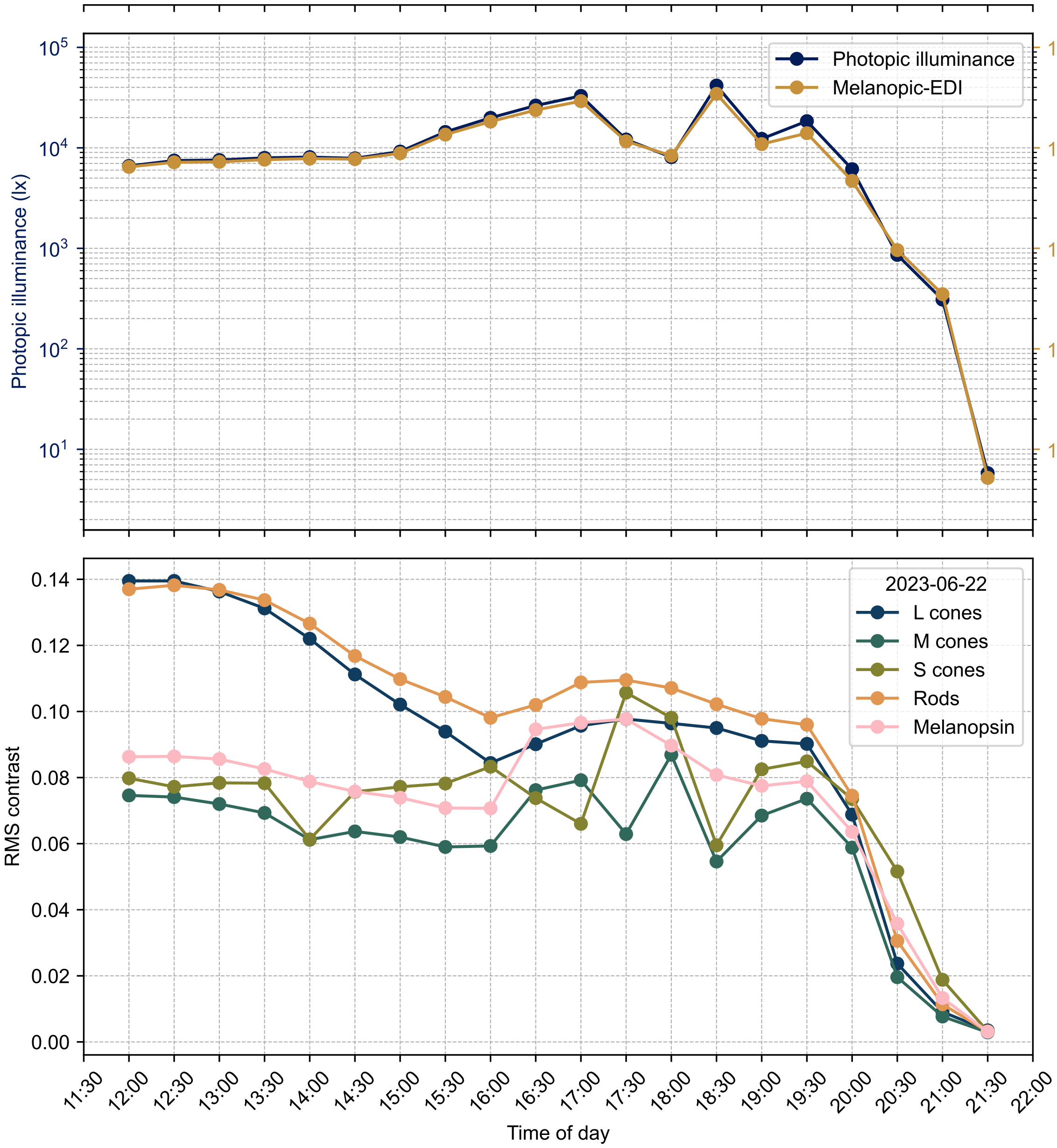
results/figures/Figure7.pdf

Author comment: *Figure 7*



`results/figures/Figure_sup.pdf`

Author comment: *Supplementary figure*



Photopic illuminance
Melanopic-EDI
Photopic illuminance (lx)
Melanopic-EDI (lx)
2023-06-22
L cones
M cones
S cones
Rods
Melanopsin
RMS contrast
Time of day

Acknowledgements

CODECHECK is financially supported by the Mozilla foundation.

Citing this document

Jan Haacker (2026). CODECHECK Certificate 2026-019. Zenodo. https://doi.org/10.5281/zenodo.21238767

About CODECHECK

This certificate confirms that the codechecker could independently reproduce the results of a computational analysis given the data and code from a third party. A CODECHECK does not check whether the original computation analysis is correct. However, as all materials required for the reproduction are freely availableby following the links in this document, the reader can then study for themselves the code and data.

About this document

This document was created using codecheck-py (a Python-base template for creating CODECHECK certificates). The CODECHECK details are filled into a jupyter notebook which is then converted into Markdown via nbconvert. Afterwards it gets converted into Typst using cmarker and then into PDF using Typst. **`sh notebook_to_pdf.sh`** will regenerate the report file.

```python
import session_info2 as si
si.session_info(os=True, cpu=True, gpu=True, dependencies=True)
```

```
tabulate  0.10.0
requests  2.34.2
wcwidth 0.8.1
executing 2.2.1
pure_eval 0.2.3
charset-normalizer  3.4.7
python-dateutil 2.9.0.post0
traitlets 5.15.1
tornado 6.5.7
session-info2 0.4.1
asttokens 3.0.1
debugpy 1.8.21
certifi 2026.6.17 (2026.06.17)
psutil  7.2.2
PySocks 1.7.1
urllib3 2.7.0
ipython 9.14.1
platformdirs  4.10.0
six 1.17.0
comm  0.2.3
Brotli  1.2.0
PyYAML  6.0.3
pandas  3.0.3
jupyter_core  5.9.1
jedi  0.19.2
prompt_toolkit  3.0.52
packaging 26.2
ipykernel 7.3.0
typing_extensions 4.15.0
decorator 5.3.1
idna  3.18
stack_data  0.6.3
parso 0.8.7
pyzmq 27.1.0
jupyter_client  8.9.1
Pygments  2.20.0
```

```
numpy 2.4.6
----  ----
Python  3.14.6 | packaged by conda-forge | (main, Jun 12 2026, 08:51:42) [GCC 14.3.0]
OS  Linux-6.8.0-124-generic-x86_64-with-glibc2.39
CPU 16/16 logical CPU cores, x86_64
GPU No GPU found
Updated 2026-07-07 13:28
```

License