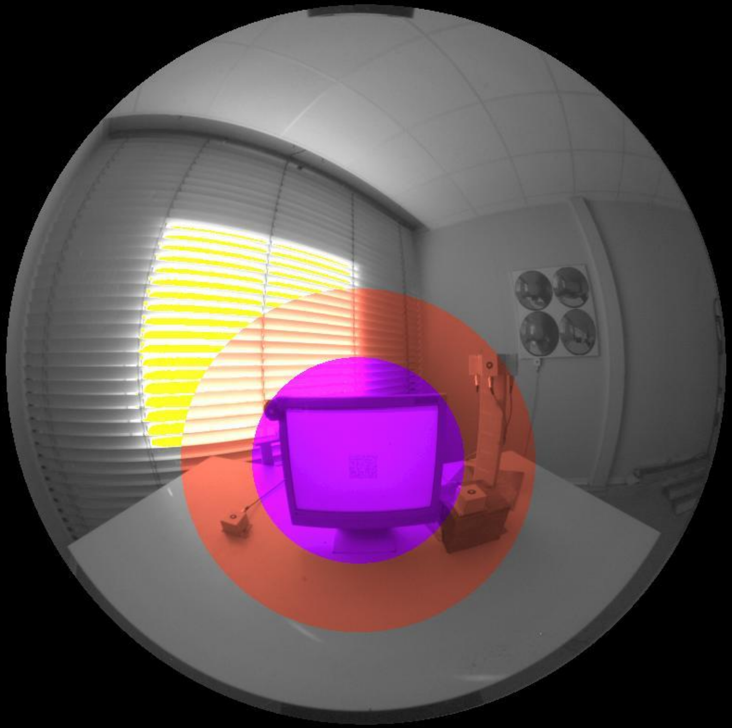
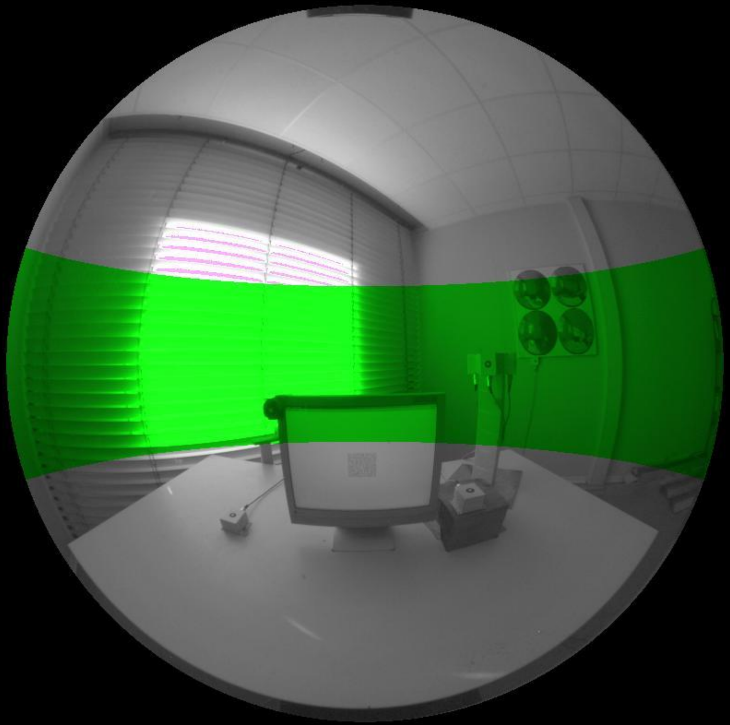
Evalglare 2.0 – new features …. faster …. and more robust HDR-image evaluation





Dr. Jan Wienold, Prof. Marilyne Andersen

École Polytechnique Fédérale de Lausanne EPFL

Interdisciplinary Laboratory for Performance Integrated Design LIPID

Content

Introduction – What is evalglare?

Bug fixes since version 1.11

New metrics and evaluation methods

New safety features for HDR-image evaluations

Repair function to sun induced pixel overflow

Speed improvement

Introduction – What is evalglare?

So far:
It is a Tool for performing a glare analysis of an Radiance-based HDR scene

It

- *detects glare sources in HDR images*
- *calculates solid angles from pixels/glare sources*
- *calculates vertical illuminance from image*
- *calculates various glare metrics (DGP, UGR, VCP, DGI, CGI…)*
- *can cut the field of view*

The phases for daylight calculation increased from 3 to 6 in the last years

But for evalglare???
The metrics were the same in the last years…

Therefore: New metrics are included!!!

Introduction – What is evalglare?

With the release of version 2.0 evalglare becomes a tool to perform luminance based evaluations of HDR images.

Following evaluations are possible now:

- *(simple) statistical analysis of the image or parts of the image (mean, median, 95 percentile, 75 percentile, standard deviation)*
- *Zonal evaluation (two circular zones possible, horizontal band)*
- *Masking within evalglare*

Various evaluations possible now.

Content

Introduction – What is evalglare?

Bug fixes since version 1.11

New metrics and evaluation methods

New safety features for HDR-image evaluations

Repair function to sun induced pixel overflow

Speed improvement

Bug fixes since version 1.11

- *Calculation of the background luminance*

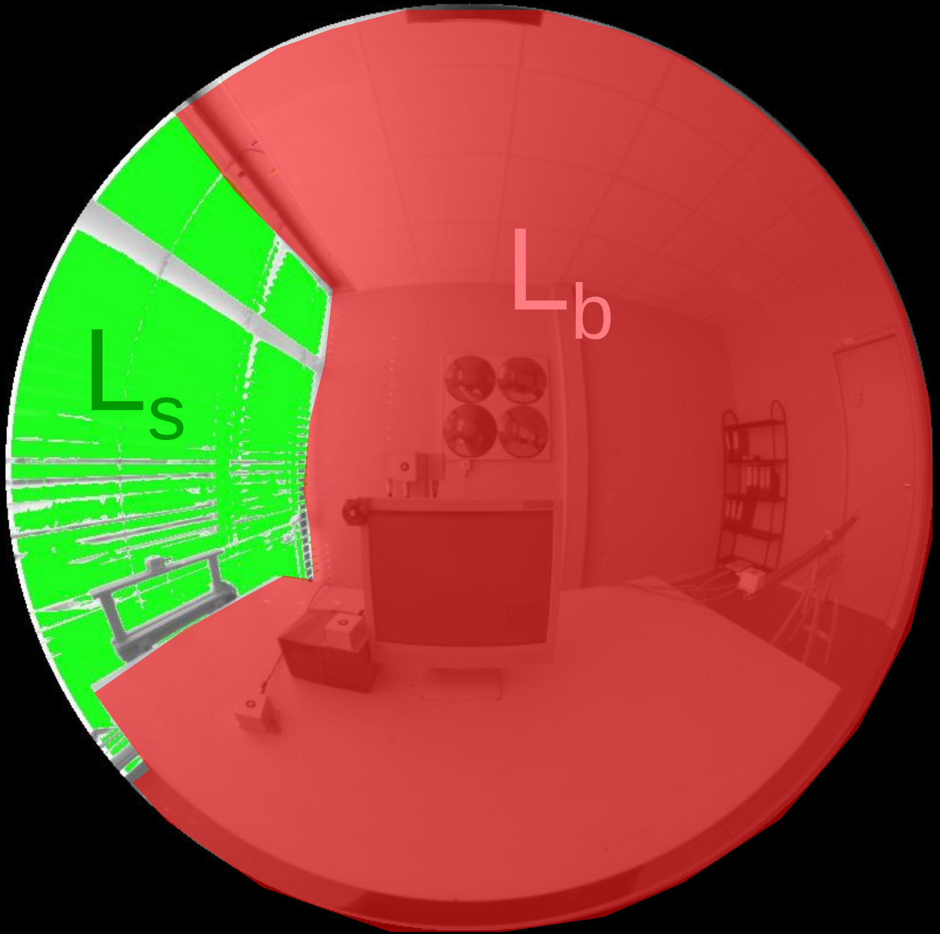
Up to version 1.16: $L_b=E_i/\Pi$

L_b: Background luminance
E_i: Indirect Illuminance

-> Pixel luminance is cosinus weighted!

From version 1.17 on:
L_b is not cosinus weighted any more, now
average luminance of the non-glare source pixels

-> Influences DGI and UGR calculation



- *Removal of the age-correction function*

Re-analysis of the data did not show significant results
Although there might be an effect, it cannot be quantified at the moment

Content

Introduction – What is evalglare?

Bug fixes since version 1.11

New metrics and evaluation methods

New safety features for HDR-image evaluations

Repair function to sun induced pixel overflow

Speed improvement

Evalglare 2.0 – new metrics and evaluation methods

Following new metrics are included:

Within standard calculation, displayed in the last line only with the –d ouput option.
Glare source section remains similar.

- *UGP : Unified glare probability according to Hirning*
- *DGI_{mod}: Modified DGI according to Fisekis*
- *UGR_{exp}: Experimental UGR according to Fisekis*
- *av_lum_pos: Average luminance of image,*
  pixels are weighted by the position index , Osterhaus
- *av_lum_pos2: Average luminance of image,*
  pixels are weighted by the squared position index, Osterhaus
- *med_lum,med_lum_pos,med_lum_pos2: Median, position weighted- and squared position weighted median luminance*

Evalglare 2.0 – new metrics and evaluation methods

Horizontal band evaluation:

activated by -B angle [rad]
e.g. for ±20° from horizontal line («40°-band») -> angle=0.349

Output in separate line (first line).
Following values within the band are calculated:

band_omega: solid angle of band [sr]

band_av_lum: average luminance of band [cd/m²]

band_median_lum: median luminance of band [cd/m²]

band_std_lum: standard deviation of luminance,

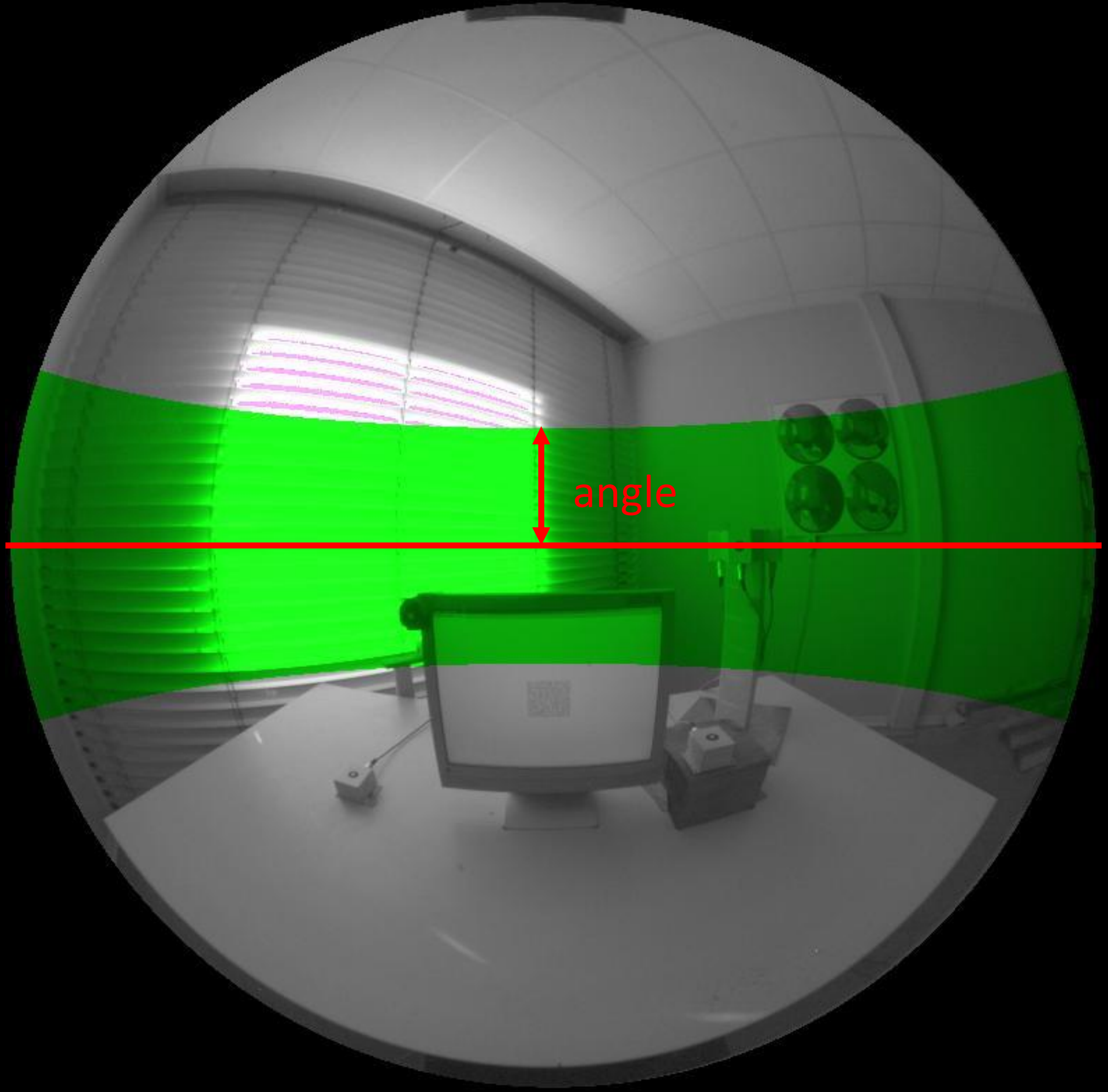
band_perc75: 75 percentile luminance of band [cd/m²]

band_perc95: 95 percentile luminance of band [cd/m²]

band_min_lum: minimum luminance of band [cd/m²]

band_max_lum: maximum luminance of band [cd/m²]

Example



Jan Wienold

LIPID Interdisciplinary Laboratory of Performance-Integrated Design

EPFL ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Zonal evaluation:

Needed for examle when performing a contrast evaluation

activated by

-l xpos ypos angle : single zone

-L xpos ypos angle1 angle2 : two zones

Angles in [rad]

Evalglare 2.0 – new metrics and evaluation methods

Zonal evaluation:

activated by

-l xpos ypos angle : single zone

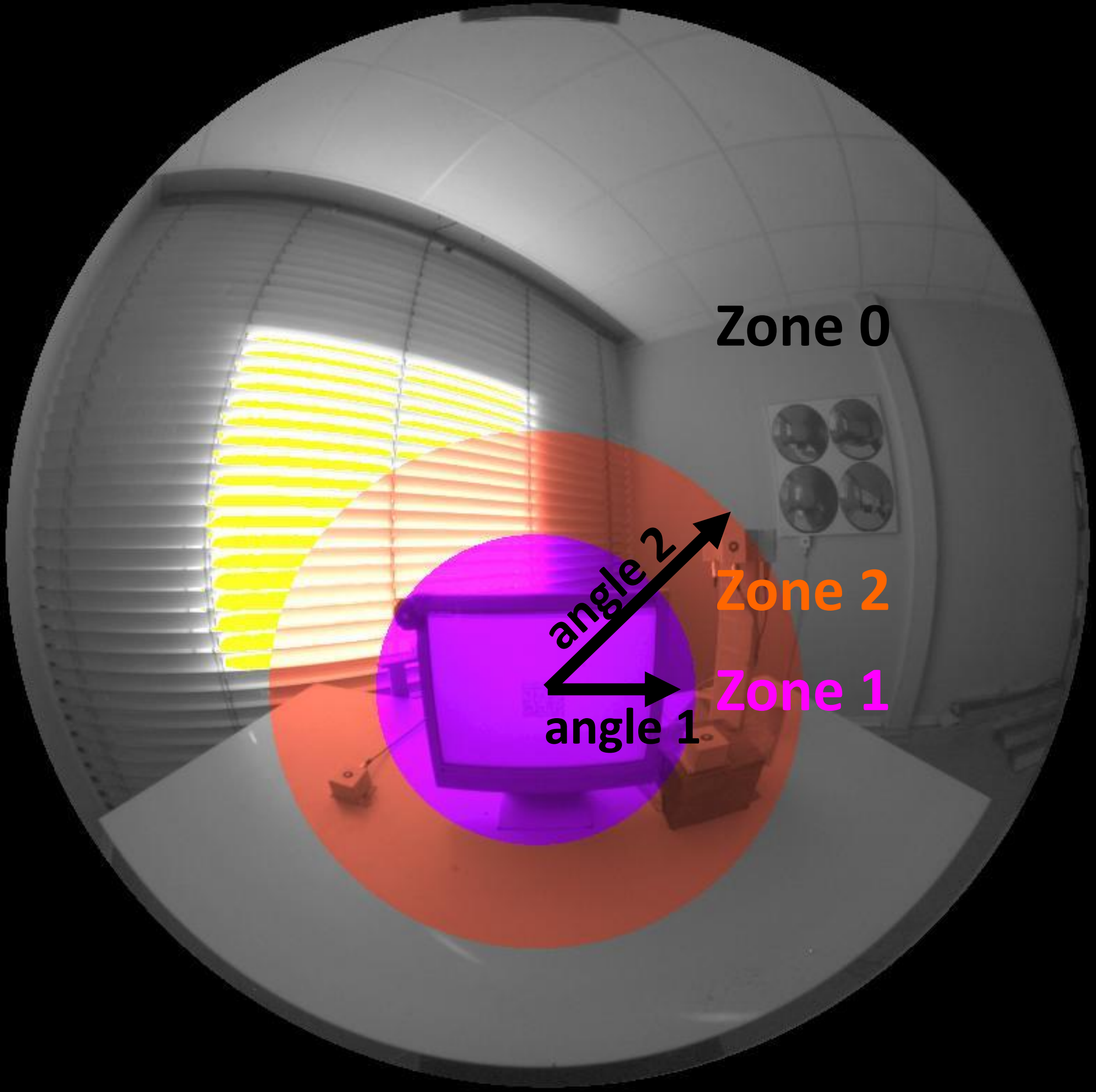
-L xpos ypos angle1 angle2 : two zones

Angles in [rad]

Output in separate lines (first lines).
Following values within the zones z1,z2 are calculated:

z1(2)_omega: solid angle of zone [sr]
z1(2)_av_lum: average luminance of zone [cd/m²]
z1(2)_median_lum: median luminance of zone [cd/m²]
z1(2)_std_lum: standard deviation of luminance of zone,
z1(2)_perc75: 75 percentile luminance of zone [cd/m²]
z1(2)_perc95: 95 percentile luminance of zone [cd/m²]
z1(2)_min_lum: minimum luminance of zone [cd/m²]
z1(2)_max_lum: maximum luminance of zone [cd/m²]

Let´s do an example evaluation ….



Jan Wienold

LIPID Interdisciplinary Laboratory of Performance-Integrated Design

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Evalglare 2.0 – new metrics and evaluation methods

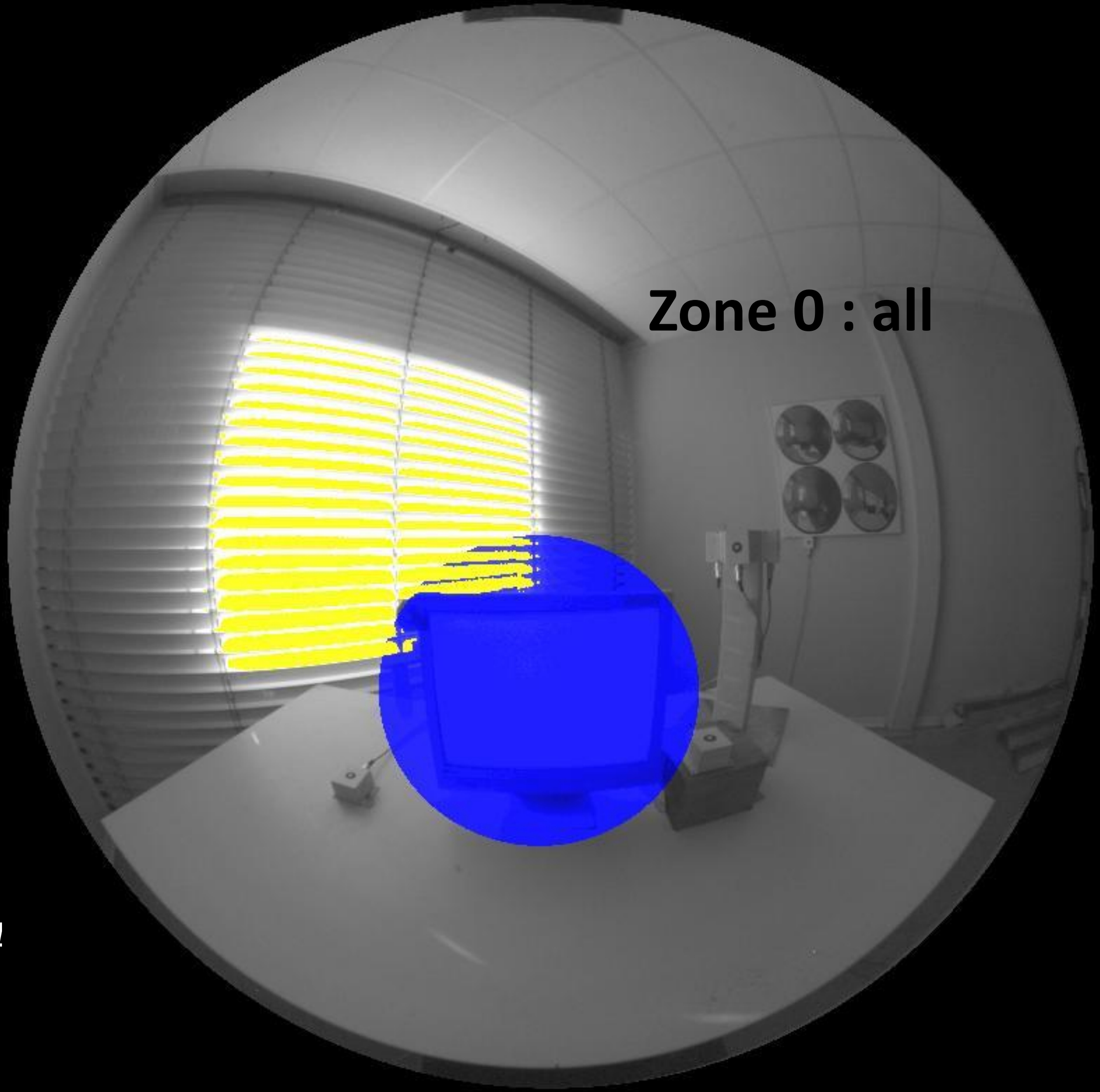
Example evaluation

No zonal evaluation

```
evalglare  –T 384 289 0.9 -d  –c output.hdr input.hdr
```

Delivers one glare source:

- *1 No pixels x-pos y-pos L_s Omega_s Posindx L_b L_t E_vert Edir Max_Lum Sigma xdir ydir zdir Eglare_cie Lveil_cie teta glare_zone*
- *1 22804.000000 253.726604 380.657331 1594.290752 0.3456751723 1.645702 155.048325 215.517090 983.203954 437.027954 10225.375000 32.430944 0.53414 0.047862 0.844038 437.027954 4.155182 32.430944 0*



Evalglare 2.0 – new metrics and evaluation methods

Example evaluation

One zone evaluation

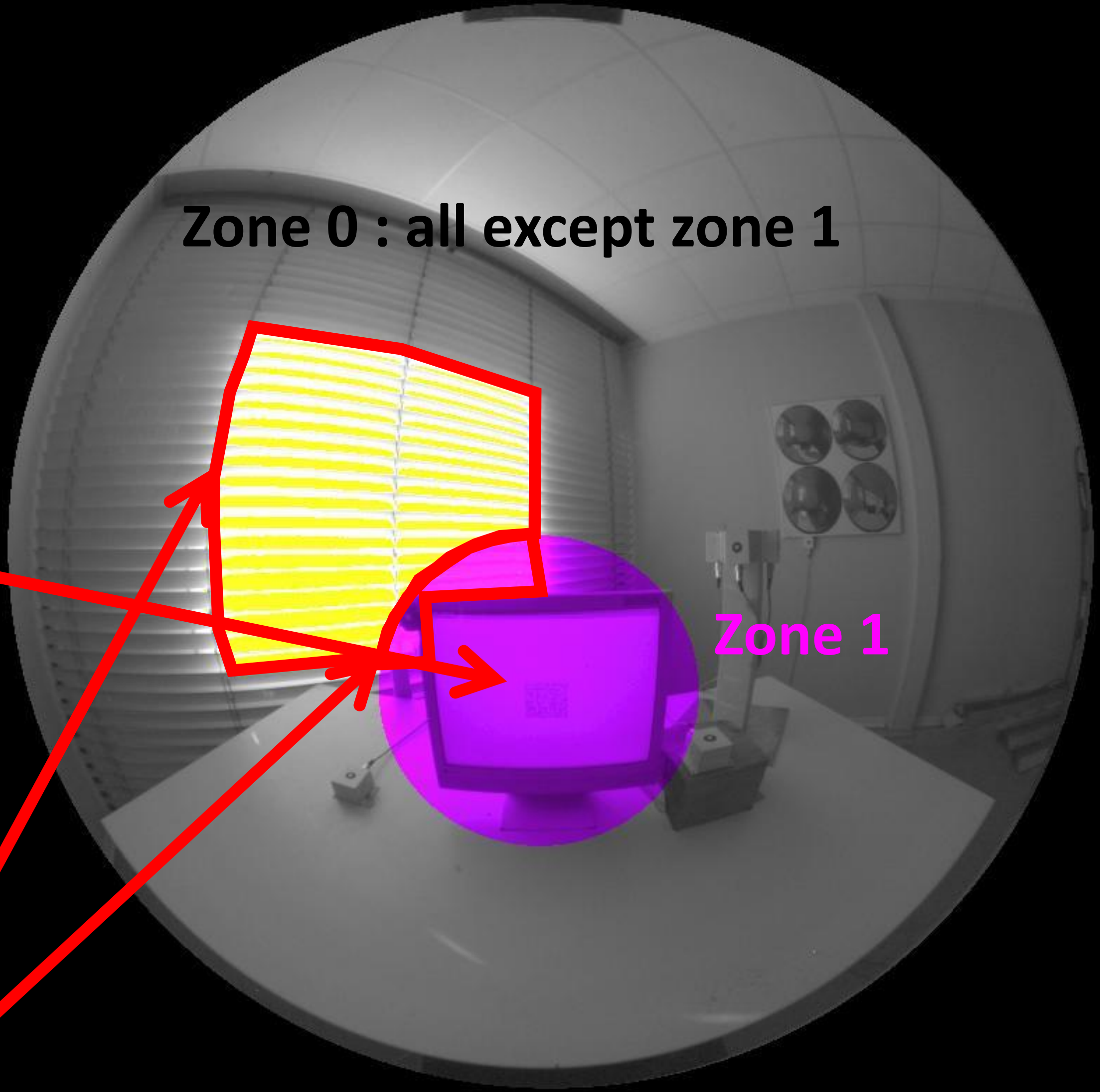
```
evalglare  –t 384 289 0.9 –l 384 289 0.9 -d
           –c output.hdr input.hdr
```

Delivers data for the zone:

*zoning:z1_omega,z1_av_lum,z1_median_lum,z1_std_lum,z1_perc_75,
z1_perc_95,z1_lum_min,z1_lum_max: 0.625647 215.517084 133.201172
351.694818 173.755859 1082.390623 13.678467 2533.968711*

And delivers two glare sources:

- *2 No pixels x-pos y-pos L_s Omega_s Posindx L_b L_t E_vert Edir Max_Lum Sigma xdir ydir zdir Eglare_cie Lveil_cie teta glare_zone*
- *1 20791.000000 244.894048 376.972998 1596.012599 0.3131968995 1.759448 155.048325 215.517090 983.203954 437.027954 10225.375000 34.597101 0.564329 0.062703 0.823165 437.027954 3.651151 34.597101 0*
- *2 2013.000000 338.901363 416.186370 1577.686503 0.0324782728 1.194143 155.048325 215.517090 983.203954 437.027954 10225.375000 13.742772 0.218148 -0.094064 0.971372 0.000000 0.000000 13.742772 1*



Evalglare 2.0 – new metrics and evaluation methods

Example evaluation

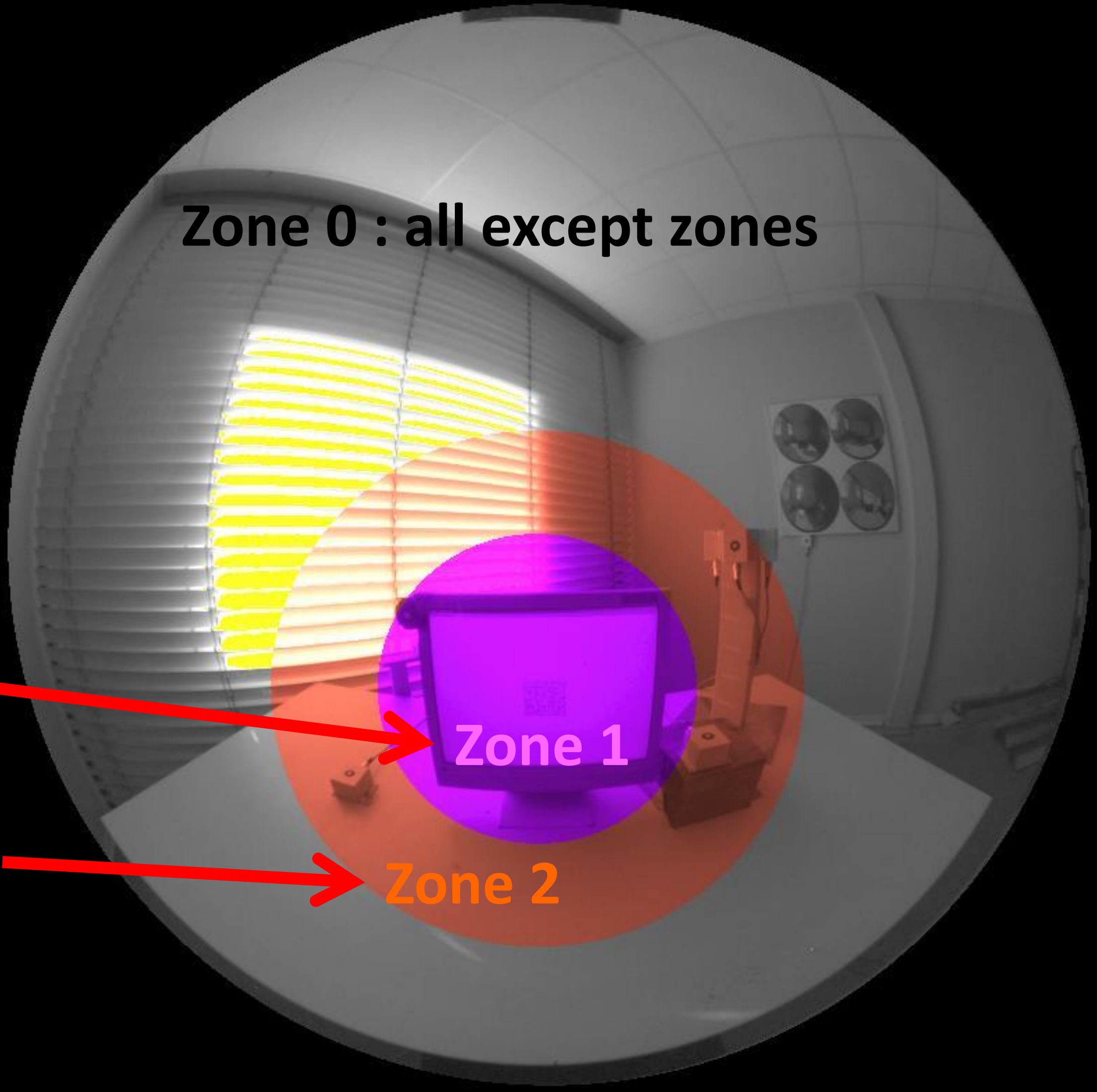
Two zones evaluation

```
evalglare  –t 384 289 0.9 –L 384 289 0.9 1.5 -d
                –c output.hdr input.hdr
```

Delivers data for the zones:

```
zoning:z1_omega,z1_av_lum,z1_median_lum,z1_std_lum,z
1_perc_75,
z1_perc_95,z1_lum_min,z1_lum_max: 0.625647
215.517084 133.201172 351.694818 173.755859
1082.390623 13.678467 2533.968711
```

```
zoning:z2_omega,z2_av_lum,z2_median_lum,z2_std_lum,z
2_perc_75,z2_perc_95,z2_lum_min,z2_lum_max:  1.060242
397.341643 109.427734 559.723433 304.160156
1717.281290 13.591064 4933.687511
```



Evalglare 2.0 – new metrics and evaluation methods

Example evaluation

Two zones evaluation

```
evalglare  –t 384 289 0.9 –L384 289 0.9 1.5 -d
              –c output.hdr input.hdr
```

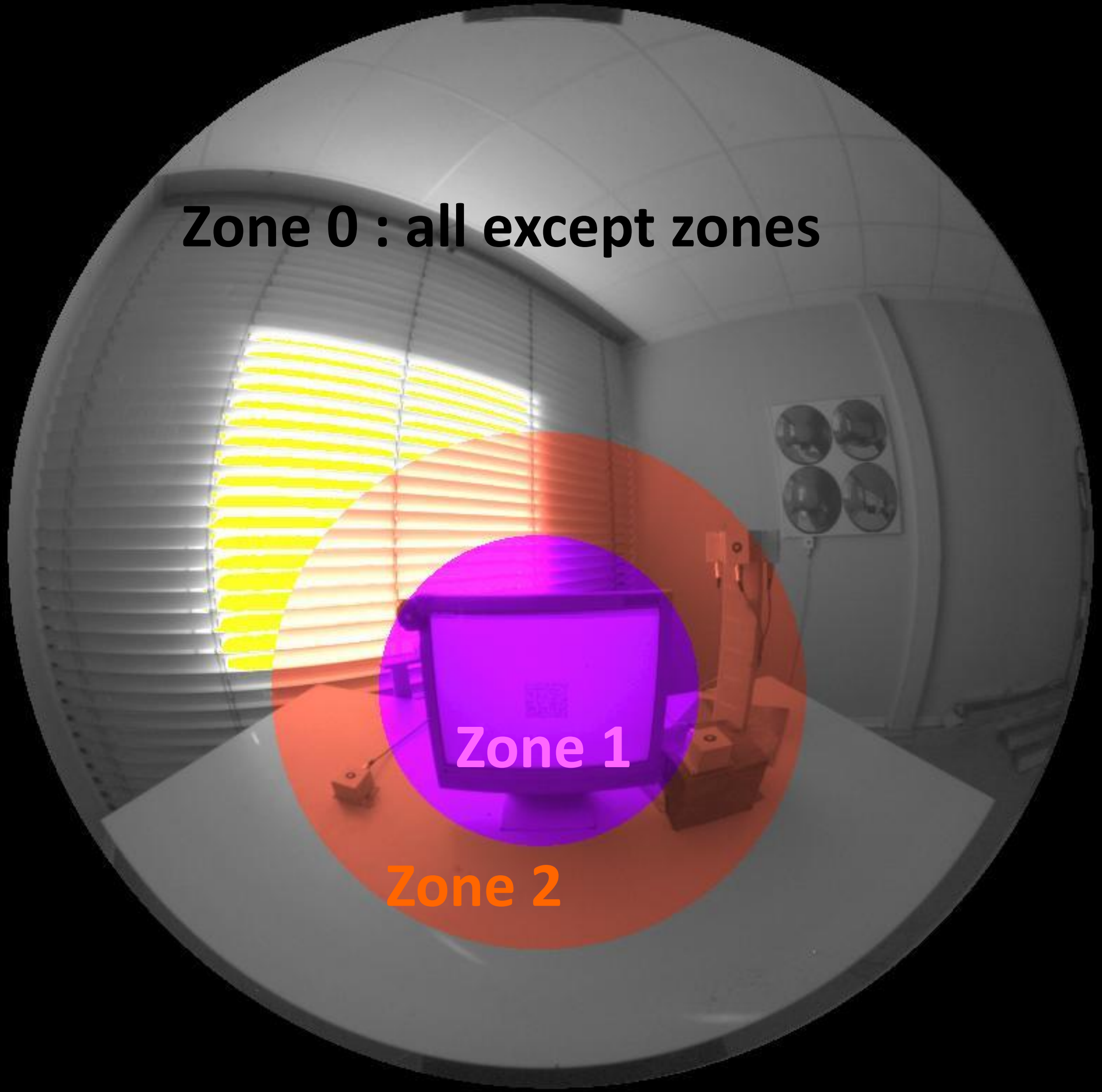
Delivers three glare souces:

```
3 No pixels x-pos y-pos L_s Omega_s Posindx L_b L_t E_vert Edir Max_Lum
Sigma xdir ydir zdir Eglare_cie Lveil_cie teta glare_zone

1 11444.000000 217.984349 352.904716 1556.418219 0.1678747451
2.364898 155.048325 215.517090 983.203954 437.027954 10225.375000
41.705430 0.648264 0.149599 0.746575 437.027954 2.512604 41.705430 0

2 9347.000000 275.979872 404.776444 1641.751643 0.1453221544 1.412642
155.048325 215.517090 983.203954 437.027954 10225.375000 27.337378
0.457045 -0.044732 0.888318 0.000000 0.000000 27.337378 2

3 2013.000000 338.901363 416.186370 1577.686503 0.0324782728 1.194143
155.048325 215.517090 983.203954 437.027954 10225.375000 13.742772
0.218148 -0.094064 0.971372 0.000000 0.000000 13.742772 1
```



Evalglare 2.0 – new metrics and evaluation methods

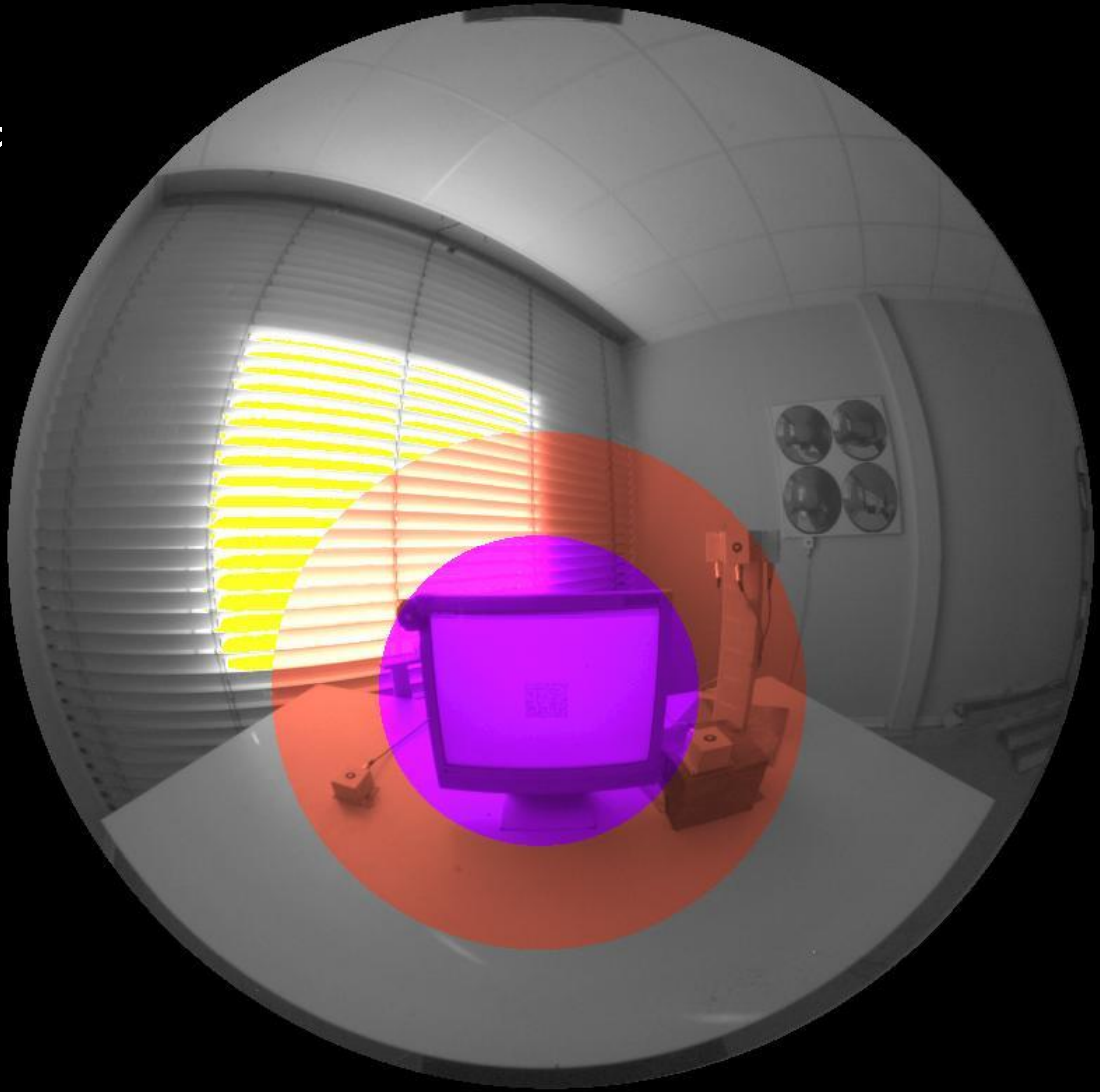
Does the zonal evaluation influence other metrics???

Yes! -> glare sources are split up!

For our example:

	DGP	*DGI*
0 zones	*0.240684*	*17.445793*
1 zone	*0.240124*	*18.075613*
2 zones	*0.240755*	*18.872232*

-> influence is usually small



Jan Wienold

Evalglare 2.0 – new metrics and evaluation methods

Masking evaluation

e.g. for evaluation of an window area

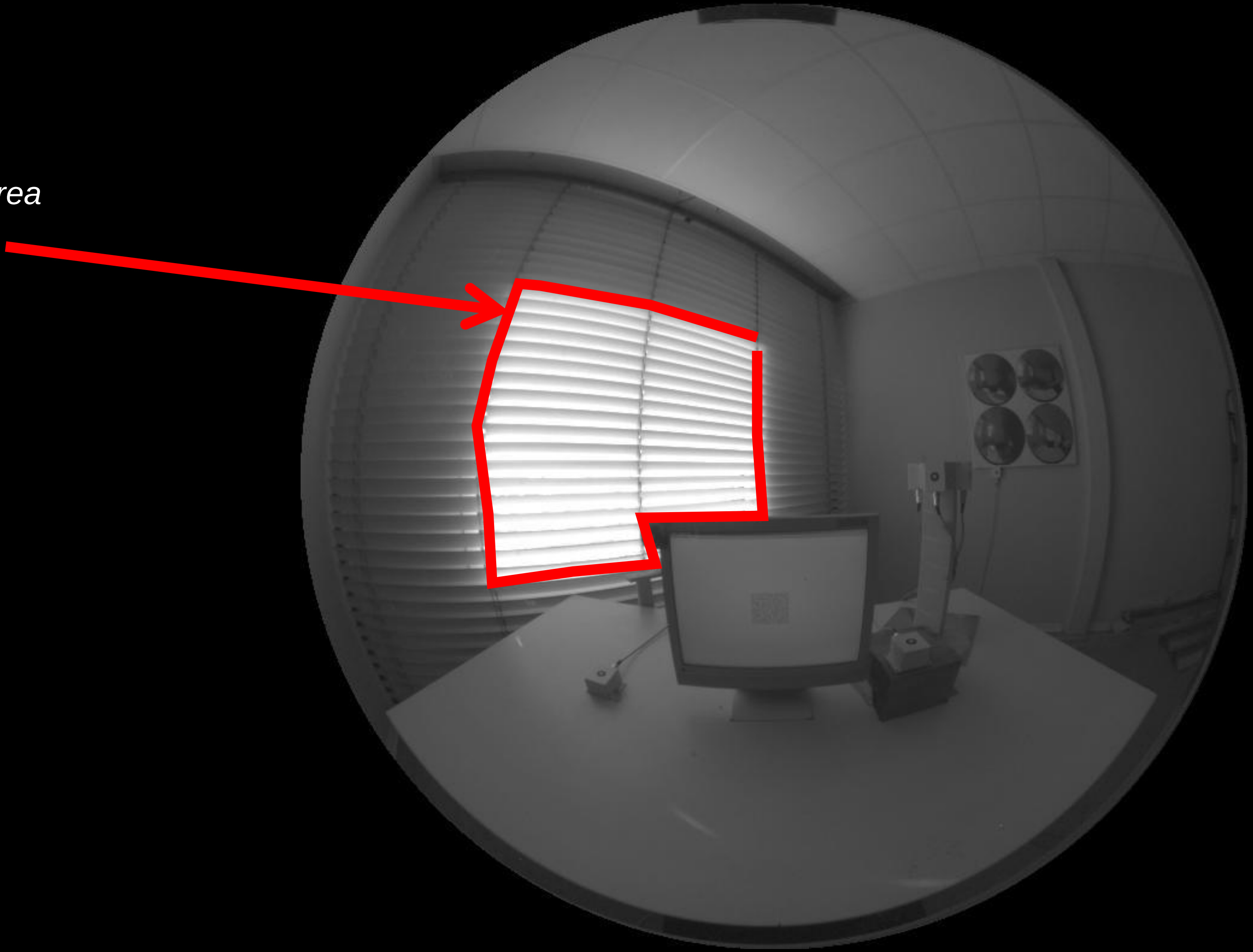
Predicted Glare Sesation Vote PGSV (Iwata)

Evalglare loads and uses a masking image to cut an area

Important: masking image must have the same size!

Not together with zoning!

activated by -A mask.hdr



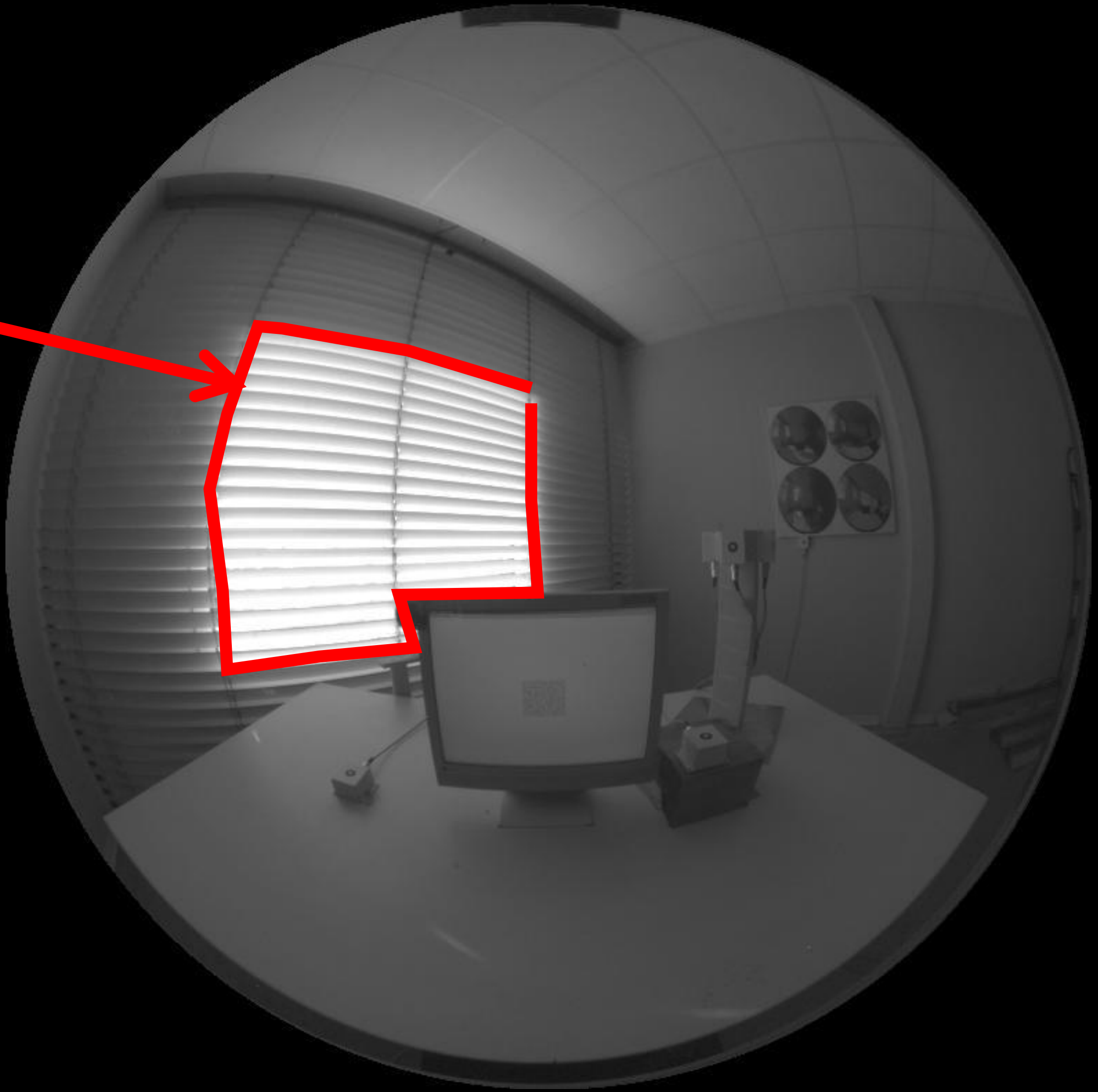
Output in separate line (first line).
Following values within the mask area are calculated:

no_pixels: no of pixels in masking area
omega: solid angle of zone [sr]
av_lum: average luminance of zone [cd/m²]
median_lum: median luminance of zone [cd/m²]
std_lum: standard deviation of luminance of zone,
perc75: 75 percentile luminance of zone [cd/m²]
perc95: 95 percentile luminance of zone [cd/m²]
min_lum: minimum luminance of zone [cd/m²]
PGSV: Predicted Glare Sesation Vote
PGSV_SAT: Saturation Predicted Glare Sesation Vote

Evalglare 2.0 – new metrics and evaluation methods

Masking evaluation – example

Steps to evaluate a window area



Masking evaluation – example

- *1. Use Photoshop or similar to create mask*

  Use ra_ppm to create a ppm file
- *Everything not of interest should be black*
- *It MUST be really black (RGB 0 0 0) !*
- *Convert it back to hdr format by*

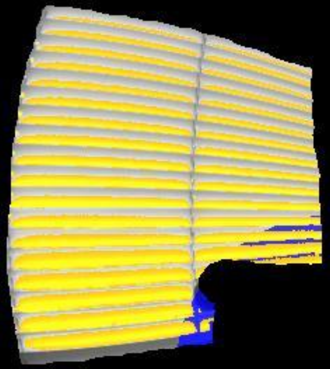
  ra_ppm –r

Evalglare 2.0 – new metrics and evaluation methods

Masking evaluation – example

- *2. Run evalglare with –A mask.hdr*

```
evalglare  –t 384 289 0.9 –A mask.hdr -d
               –c output.hdr input.hdr
```



```
masking:no_pixels,omega,av_lum,median_lum,std_lum,
perc_75,perc_95,lum_min,lum_max,pgsv,pgsv_sat:
44732 0.675010 1178.508190 1065.609375 543.535164
1459.968748 2164.781246 33.300294 7316.625082 -
0.053004 1.475234
```

Evalglare 2.0 – new metrics and evaluation methods

Summary of new metrics included:

- *UGP : Unified glare probability according to Hirning*
- *DGI_{mod}: Modified DGI according to Fisekis*
- *UGR_{exp}: Experimental UGR according to Fisekis*
- *PGSV: Predicted Glare Sesation Vote according to Iwata*
- *Mean, median std of (lower) window area (standards, Wymelenberg)*
- *Mean, median, std of 40° band (Wymelenberg)*
- *av_lum_pos: Average luminance of image,
pixels are weighted by the position index , Osterhaus*
- *av_lum_pos2: Average luminance of image,
pixels are weighted by the squared position index, Osterhaus*
- *med_lum,med_lum_pos,med_lum_pos2: Median, position weighted- and squared position weighted median luminance*

Content

Introduction – What is evalglare?

Bug fixes since version 1.11

New metrics and evaluation methods

New safety features for HDR-image evaluations

Repair function to sun induced pixel overflow

Speed improvement

Why?

Several users didn´t use «correct» headers.

Evalglare results are completely wrong.

- *Extended header check*
- *Check for fish-eye lens*

Evalglare 2.0 – 2.0 – New safety features for HDR-image evaluations

New safty features for HDR-image evaluations

Reasons for «wrong» headers:

-> using pcompos for cutting image without correcting header after

-> fish eye lens is not automatically detected by photosphere or hdrgen

What happens?

- *Exposure gets invalid*
- *View setting gets invalid (wa already checked since version 1.00)*

Evalglare 2.0 – 2.0 – New safety features for HDR-image evaluations

Example

Evalglare 2.0 – New safety features for HDR-image evaluations

- *Extended header check -> check for «valid» view and «valid» exposure*
- *Check for fish-eye lens -> check on black corners*

Content

Introduction – What is evalglare?

Bug fixes since version 1.11

New metrics and evaluation methods

New safty features for HDR-image evaluations

Repair function to sun induced pixel overflow

Speed improvement

Evalglare 2.0 – Repair function to sun induced pixel overflow

Problem: HDR contains overflow pixels

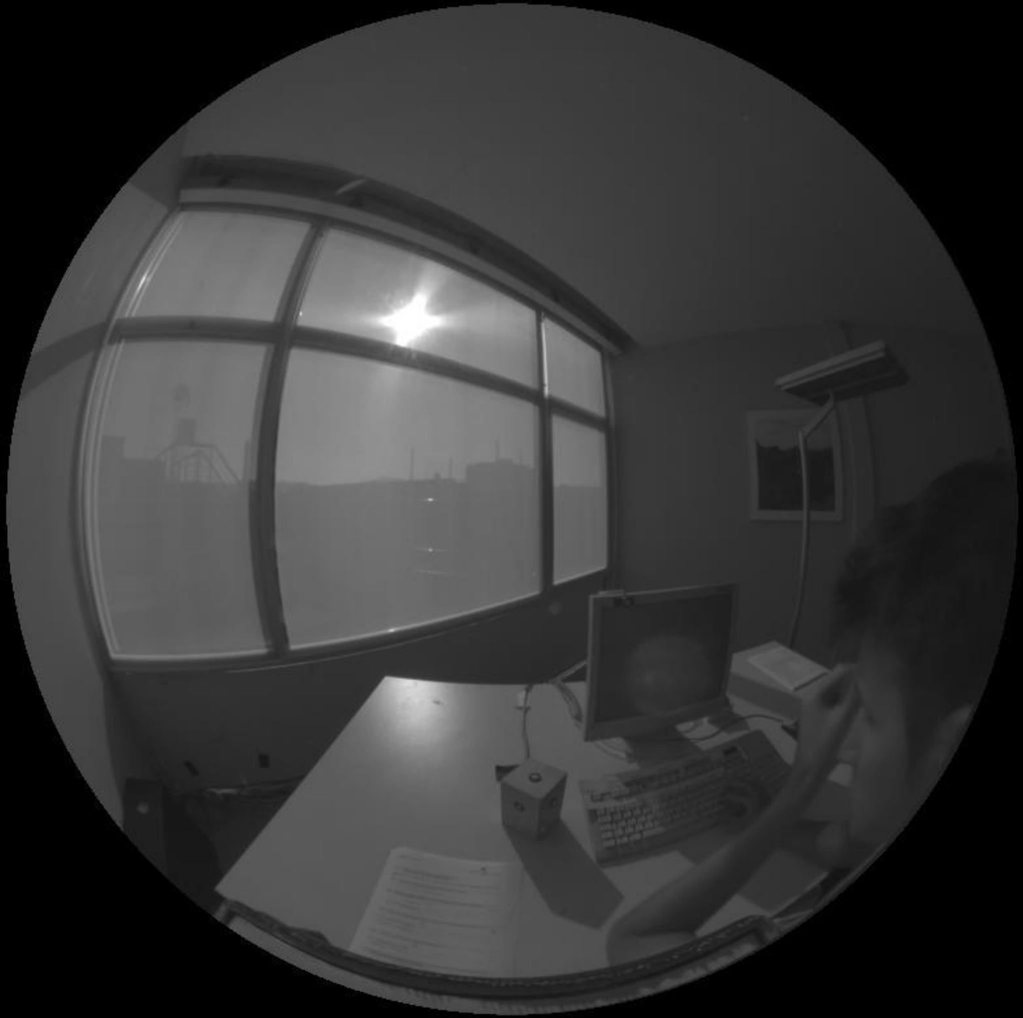
How to find out??

Measured illuminance does not match calculated

Example:

$E_{measured}$=12000 lux

$E_{calculated}$=6000 lux

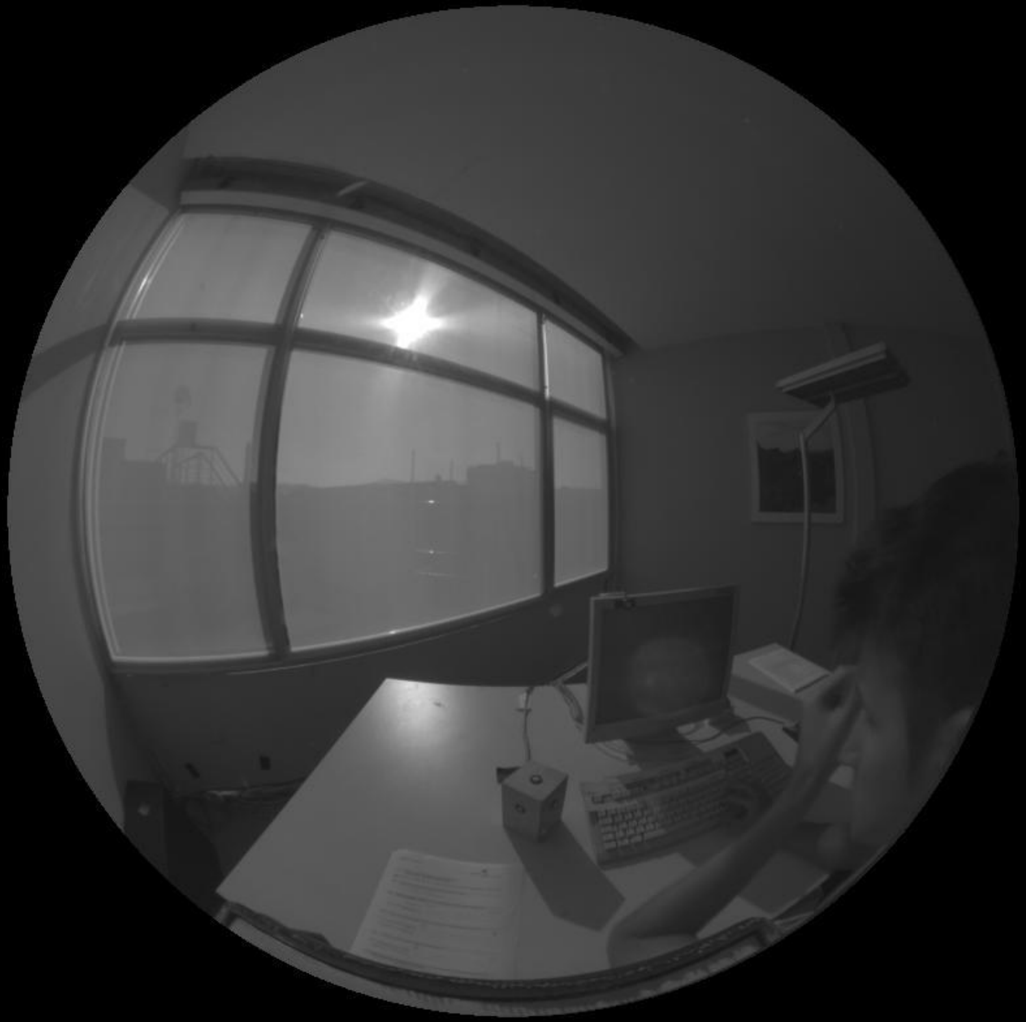


Evalglare 2.0 – Repair function to sun induced pixel overflow

How to solve?

It should be prevented from the beginning by the use of filters

But what to do when the measurements were already finished?



Evalglare 2.0 – Repair function to sun induced pixel overflow

Solution: Use the vertical illuminance to replace the «sun-pixels» to match the illuminance

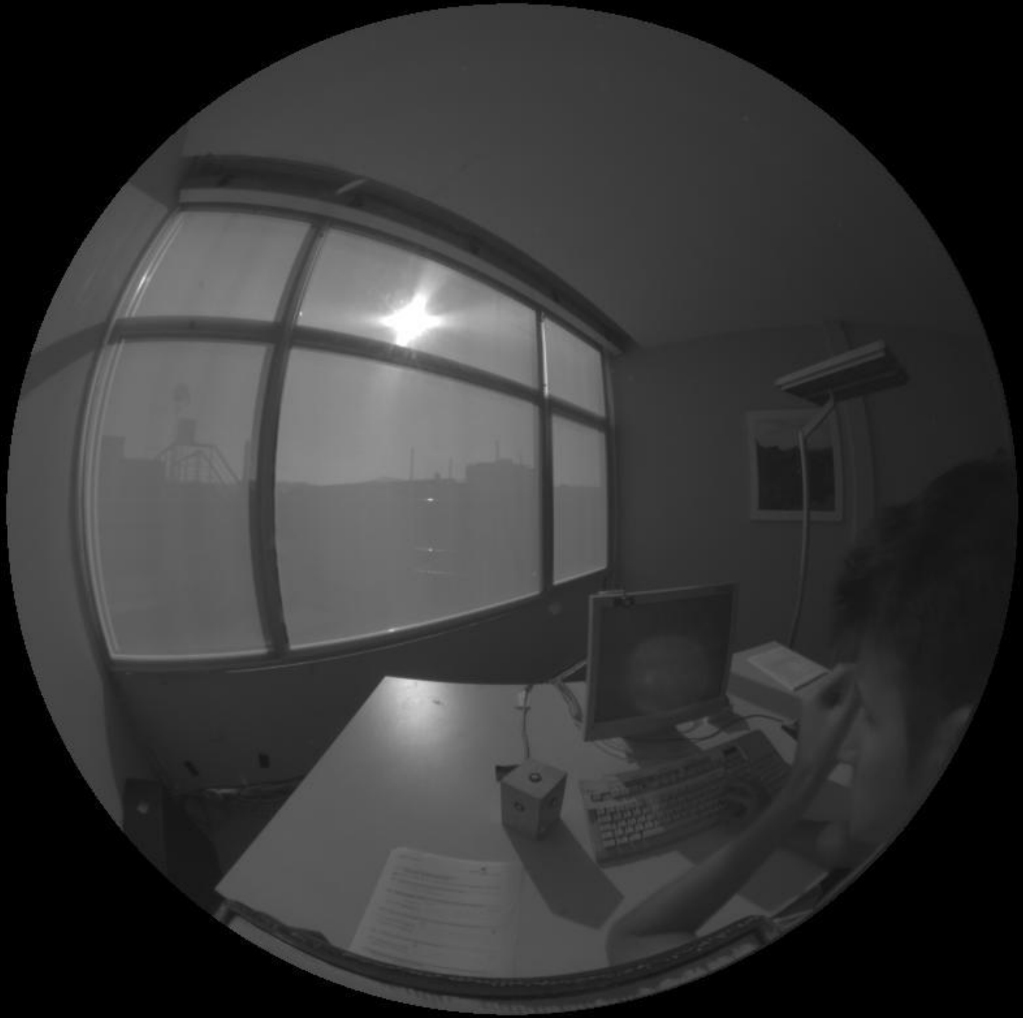
New option in evalglare –N xpos ypos angle E

xpos: x-position of sun

ypos: y-pos of sun

angle: angle of sun disk (should contain also circumsolar area

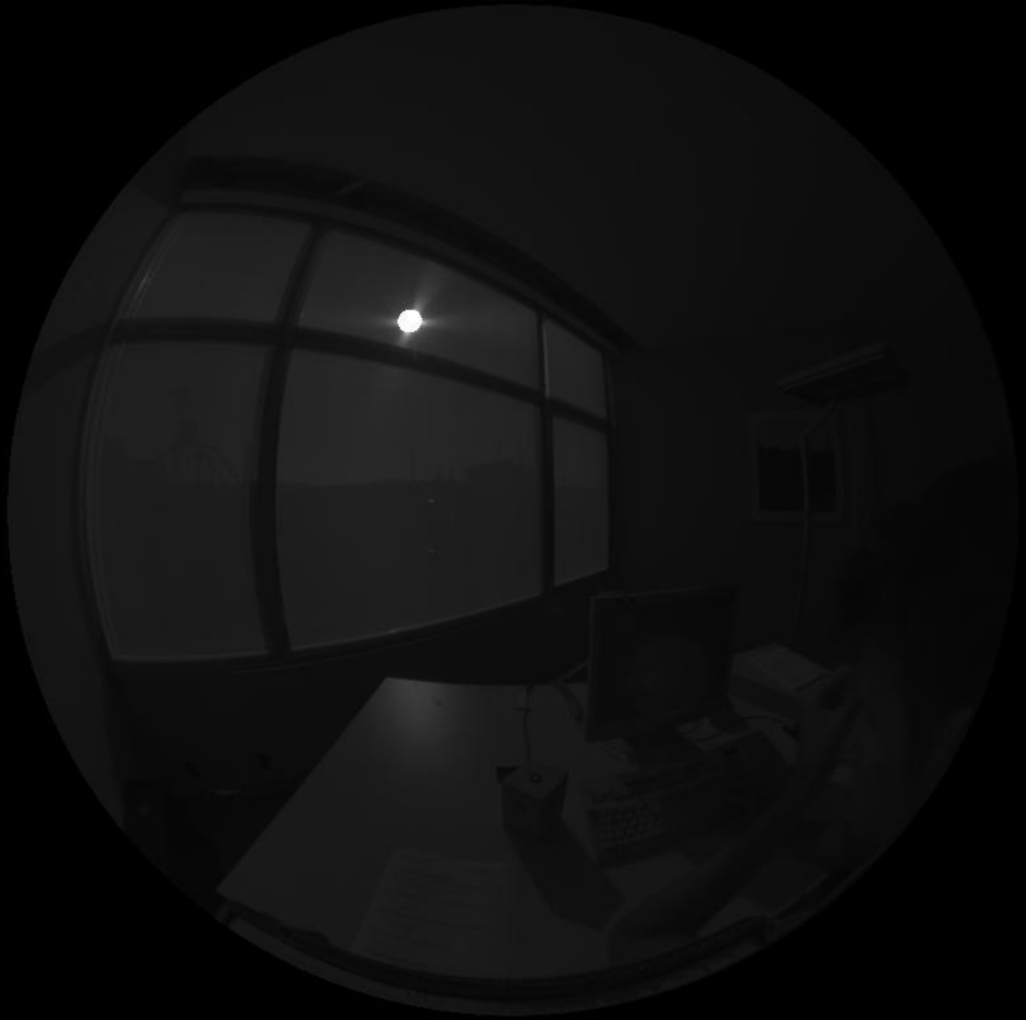
E: measured Illuminance [lux]



Evalglare 2.0 – Repair function to sun induced pixel overflow

Result

Example



Content

Introduction – What is evalglare?

Bug fixes since version 1.11

New metrics and evaluation methods

New safty features for HDR-image evaluations

Repair function to sun induced pixel overflow

Speed improvement

Evalglare 2.0 – speed improvement

Example

Algorithm for combining glare sources has been improved

In average factor 10 speed improvement!

The longer it took before, the more is the improvement.

The bad news: Not affecting the annual calculation, this was already optimized before..

Evalglare 2.0

The presented version is already as beta-version 1.31 in the head release since several weeks.

The official 2.0 release will be in the next days.

I´m thankful for any suggestion !

Open to discuss any other hdr-evaluation to include

Thank you for your attention!