21NRM01 HiDyn

{HiDyn}

D5: Report describing the implementation of an open source HDR algorithm for the generation of traceable luminance images, scaled to one or a few traceable spot measurements of the scene.

Organisation name of the lead participant for the deliverable: CNAM

Due date of the deliverable: 31 July 2025

Actual submission date of the deliverable: 28 November 2025

Confidentiality Status: PU - Public, fully open (remember to deposit public deliverables in a trusted repository)

Funded by the European Union. 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.

The project has received funding from the European Partnership on Metrology, co-financed from the European Union's Horizon Europe Research and Innovation Programme and by the Participating States.

Glossary

DSLR	Digital single-lens reflex
HDR	High dynamic range
ILMD	Imaging luminance measurement device
LDR	Low dynamic range
PRNU	Photon response non-uniformity
SNR	Signal-to-noise ratio
TLM	Temporal light modulation

TABLE OF CONTENTS

1 Summary ... 4
2 Introduction ... 4
3 HDR merging method ... 4
3.1 Principle of the HDR merging algorithm ... 4
3.2 Algorithms implemented in the code ... 6
3.3 Algorithm parameters ... 7
3.4 Determining the luminous responsivity from traceable spot measurements ... 7
4 MATLAB code ... 8
4.1 Code architecture ... 8
4.2 Camera parameters ... 8
4.3 Image selection ... 9
4.4 Exposure time file ... 9
4.5 RAW file conversion ... 9
4.6 Dark correction ... 10
4.7 Choice of HDR merging algorithm ... 10
4.8 HDR image saving ... 10
4.9 HDR image display ... 11
5 Additional features not included in the MATLAB code ... 11
5.1 LDR series acquired by varying parameters other than integration time ... 11
5.2 Non-linearity correction ... 11
5.3 LDR images acquired using a non-linear camera ... 11
5.4 Additional processing steps not included in the code ... 12
6 Example dataset ... 12
7 Licence ... 13
8 Code availability ... 13
9 Conclusion ... 13
10 References ... 13

1 Summary

This report describes the implementation of an open-source high dynamic range (HDR) algorithm code for the generation of traceable HDR luminance images. The toolbox implements four merging algorithms compatible with uncertainty propagation and provides a complete processing chain including dark correction, raw file handling, and camera calibration.

The code is made available under MIT licence and can be downloaded from GitHub:
https://github.com/HiDyn-EURAMET-EPM-21NRM01/HDRmerge_HYDIN

This document aims that anyone in lighting or measurement fields can produce HDR luminance maps using ordinary cameras, helping to standardise measurements for glare, road lighting, and light pollution studies.

2 Introduction

Imaging devices, such as cameras and Imaging Luminance Measurement Devices (ILMDs), are increasingly used for optical measurements in applications like glare analysis and obtrusive light evaluation. However, the dynamic range of digital imaging sensors is often insufficient for typical scenes in these applications, requiring high dynamic range (HDR) imaging methods.

Digital HDR imaging techniques, including merging algorithms, were established in the early 1990s [1]. Today, HDR is commonplace in devices like digital single-lens reflex (DSLR) cameras and smartphones, sometimes employing AI-based algorithms optimized for visually pleasing results, i.e. colour balancing and local tone mapping. In quantitative imaging measurement, rather than aesthetic images, the focus is on producing accurate, SI-traceable luminance maps with well-defined measurement uncertainties, which is the objective of our project.

HDR imaging involves capturing multiple low dynamic range (LDR) images at different exposures (e.g., integration times) and combining them using an HDR merging algorithm to create an image with an increased dynamic range. Regardless of the method used to vary exposure, the merging principle remains the same. When a pixel is well-exposed in multiple LDR images, each provides a good estimate of the luminance, whose uncertainty is not limited by noise. The main challenge for HDR algorithms is to combine these images to minimize error, not only due to noise but also from potential systematic errors.

In the framework of the joint research project 21NRM01 HiDyn, a review of HDR merging algorithms compatible with uncertainty propagation was conducted. From this review resulted a selection of 4 algorithms for HDR merging that have been implemented in a MATLAB code.

3 HDR merging method

This section gives an overview of the HDR merging algorithm principles and detail the four algorithms implemented in the MATLAB code. The case of LDR images captured at varying integration times (or exposure time[1]) by a linear camera is presented. The content of this section is also developed in [2].

3.1 Principle of the HDR merging algorithm

For a LDR image captured by a perfect linear camera with varying exposure times, the luminance L_{LDR} of the area of the scene corresponding to a non-saturated pixel is:

---

[1] *Vocabulary notes: In this document, the terms integration time and exposure time are used interchangeably to denote the duration during which a pixel collects light. Strictly speaking, integration time refers to the period when a pixel accumulates charge from incoming light, whereas exposure time refers to the period during which the pixel is illuminated.*

$$L_{\mathrm{LDR}} = \frac{1}{s_V} \cdot \frac{(D - D_0)}{t} + \varepsilon, \quad (1)$$

where

t	is the integration time in *s*;
D	is the value of the LDR pixel signal in counts;
D_0	is the value of the LDR pixel associated dark signal in counts;
s_V	is the luminous responsivity (or count-to-luminance adjustment factor) in count·s^{-1}·cd^{-1}·m^{2};
ε	is the measurement error.

The linear relationship between the luminance, pixel value, integration time and luminous responsivity expressed by Eq. (1) can also be visualized on the plot representing the dark corrected pixel value as a function of integration time, as shown on Figure 1. The slope of the line between the origin and the point of coordinates (t_i, D-D_0) corresponds to the luminance value multiplied by the luminous responsivity s_V.

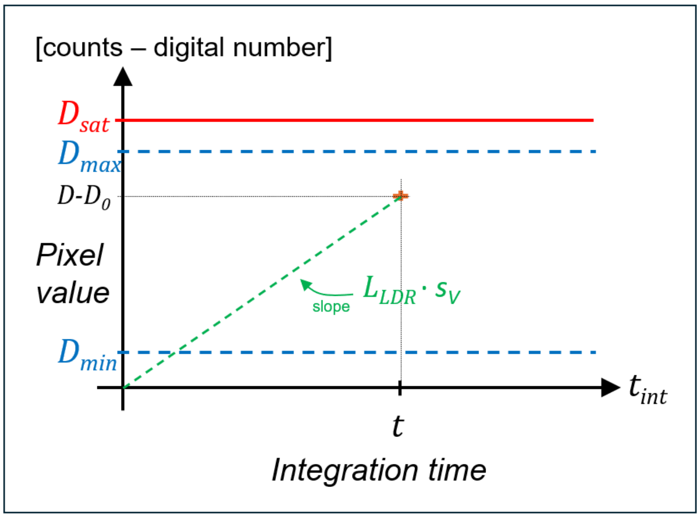


Figure 1. For a linear camera, the relationship between integration time, corrected pixel value, luminous responsivity and luminance is linear.

When HDR imaging methods are applied, several LDR images of the same scene are captured at various integration time. For a given pixel, we consequently obtain several estimations of the luminance of the scene area imaged on a pixel. For a non-saturated pixel in the i^{th} image, we have:

$$L_{\mathrm{LDR,i}} = \frac{1}{s_V} \cdot \frac{(D_i - D_{0,i})}{t_i} + \varepsilon_i, \quad (2)$$

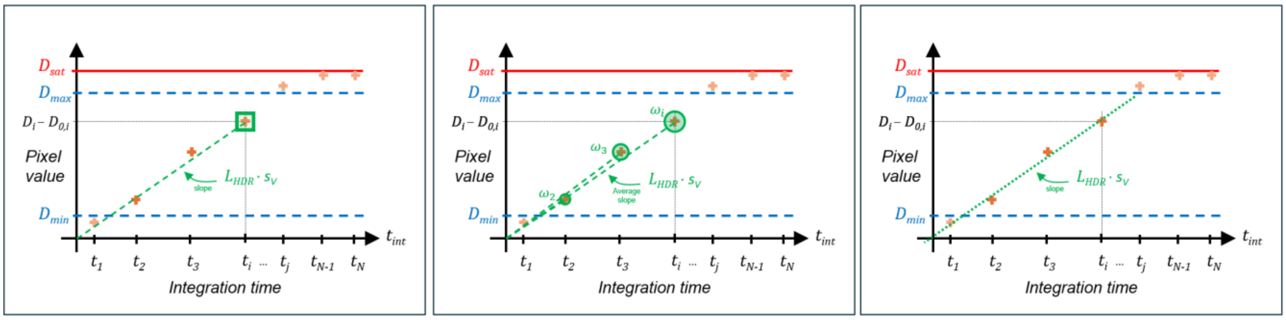
where each parameter is identical to the parameters introduced in Eq. (1) but specific to the i^{th} image of the LDR series.

In the case of a perfect camera unaffected by errors, all dark corrected pixel values should be placed on the same dotted green line represented in Figure 1, and any well-exposed value can be used to calculate luminance. However, for a real camera, different error sources affect each luminance estimate to a greater or lesser extent. The question of the HDR reconstruction algorithm is therefore how to use the acquired data to calculate the luminance of the HDR scene with the highest possible accuracy.

Well-exposed values are determined using the threshold values D_{min} and D_{max}, which define the range within which non-linearity and noise of the measurement are acceptable.

3.2 Algorithms implemented in the code

Algorithms were selected for their simplicity and ability to produce accurate luminance estimations from HDR image captures. They fall into three main categories illustrated in Figure 2. The mathematical expressions for the four implemented algorithms are given below. As they are deterministic, continuous and differentiable, they do not pose any limitations to uncertainty propagation.



"Best exposure" algorithm *"Weighted average" algorithm* *"Linear regression" algorithm*

Figure 2. Visual representation of the algorithms implemented in code, valid for a linear camera.

3.2.1 Best exposure algorithm

The best exposure algorithm constructs the HDR image by selecting the best-exposed pixel from the LDR images. The best-exposed pixel is the one with the highest value D_i below the maximum usable value (due, for instance to clipping or nonlinear behaviour). For LDR images at different exposure times, the HDR pixel value can be obtained as:

$$L_{\mathrm{HDR}} = \frac{1}{s_V} \cdot \frac{(D_i - D_{k,i})}{t_i}\text{, with } t_i = \max\{t_i \text{ such that } D_{\min} \leq D_i \leq D_{\max}\} \quad (3)$$

The threshold values D_{min} and D_{max} define the range within which pixels can be considered as well-exposed. If the user needs to avoid non-reconstructed pixels, D_{min} can be set to zero. These thresholds also apply to the other HDR merging algorithms.

3.2.2 Weighted-average algorithm

Weighted-average algorithms aim at reducing the measurement error by using information from all well-exposed pixels across the LDR images. With w_i, the weight applied to the pixel of the i^{th} image, the HDR luminance is the weighted average of the N well-exposed pixel values at different exposure times:

$$L_{HDR} = \frac{1}{s_V} \frac{\sum_{i=1}^{N} q_i\, w_i\, \frac{(D_i - D_{k,i})}{t_i}}{\sum_{i=1}^{N} q_i\, w_i}\text{, with } q_i = \begin{cases} 1 & (D_{\min} \leq D_i \leq D_{\max}) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Weighted average algorithms can similarly be applied in the logarithmic domain [3]:

$$L_{HDR} = 10^{\left(\frac{\sum_{i=1}^{N} q_i\, w_i (log10\,((D_i - D_{k,i})/s_V - log10(t_i))}{\sum_{i=1}^{N} q_i\, w_i}\right)}\text{, } with\ q_i = \begin{cases} 1 & (D_{min} \leq D_i \leq D_{max}) \\ 0 & otherwise \end{cases} \quad (5)$$

To leverage the redundant information between the LDR images, adequate weights need to be selected. Various weighting methods have been proposed, typically based on pixel value, exposure time, or signal-to-noise ratio (SNR). Pixel-value-based weights favour values near the camera's dynamic range centre, as seen with the hat function or bowler hat function [4]. Exposure-time-based weights reduce measurement noise, proportional to exposure time under photon noise, improving SNR. In the photon-limited regime, exposure time as the weight corresponds to a theoretical optimal of SNR.

In the proposed open-source code, the implemented algorithms are the exposure time weighted average algorithm in the linear and logarithmic domains, described by Eqs. (4) and (5) with $w_i = t_i$, which is a quantity directly related with the square root of the SNR.

3.2.3 Linear regression algorithm

As illustrated in Figure 2, HDR reconstruction can also be visualized by plotting the pixel values from different LDR images against exposure time. One approach is to calculate the slope of the line through these points, representing luminance multiplied by the luminous responsivity s_V. This slope can be determined using least-squares regression:

$$L_{\mathrm{HDR}} = \frac{1}{s_V} \frac{\sum_{i=1}^{N} q_i \cdot \sum_{i=1}^{N} q_i\, t_i\, (D_i - D_{k,i}) - \sum_{i=1}^{N} q_i\, t_i \cdot \sum_{i=1}^{N} q_i\, (D_i - D_{k,i})}{\sum_{i=1}^{N} q_i \cdot \sum_{i=1}^{N} q_i\, t_i^2 - \left(\sum_{i=1}^{N} q_i\, t_i\right)^2}, \text{ with } q_i = \begin{cases} 1 & (D_{\min} \leq D_i \leq D_{\max}) \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

3.3 Algorithm parameters

HDR algorithms are applied to well-exposed pixels only, considering upper and lower threshold values D_{min} and D_{max}. These thresholds are set to limit the impact of noise and temporal light modulation (TLM) at small values and non-linearity and saturation near the sensor's upper limits.

Together with minimum and maximum exposures, these thresholds define a range of well-exposed pixels (Figure 3), which constrains the dynamic range that can be reached in an ideal scenario without camera internal stray light.

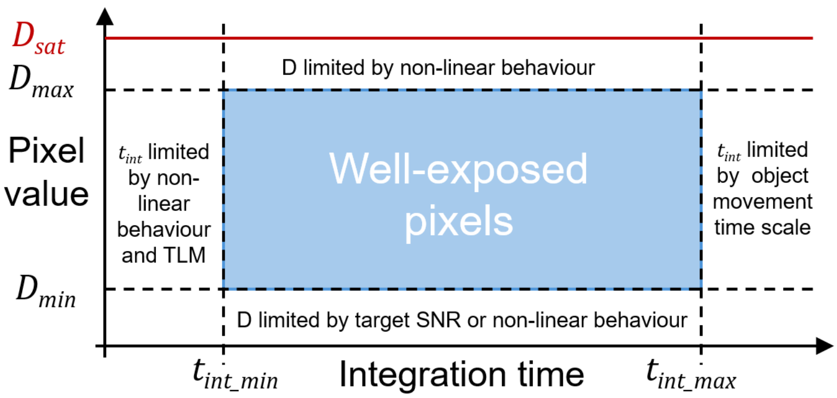


Figure 3. Proper acquisitions and processing conditions.

3.4 Determining the luminous responsivity from traceable spot measurements

When the luminous responsivity of the camera s_V is not known, but the user has access to a calibrated device for luminance measurements (such as a luminance meter performing spot measurements), the parameter s_V can be calculated as follows.

The s_V parameter of a camera can be determined by measuring the same uniform area (the *target*) with both the camera and a calibrated luminance measuring instrument, and then comparing the reference luminance value with the measurement obtained from the camera.

First, a scene of limited luminance range containing a uniform area (or target) must be identified. The target should exhibit luminance that is as independent as possible from the direction of observation, in order to minimise measurement errors caused by the fact that the camera and the reference instrument might not observe the target from exactly the same viewpoint. This condition can be achieved by using an integrating sphere port or a diffusely reflecting surface as target. The scene must not contain very bright sources, as these may introduce stray light into both the camera under calibration and the reference instrument.

Next, an LDR image of the scene must be captured with the camera for which the s_V coefficient is to be determined, using the same focus, aperture, and gain settings that are applied during HDR acquisition. As the s_V parameter is particularly sensitive to aperture, this setting must be adjusted correctly. For the LDR acquisition, the exposure time should be set so that the pixel values corresponding to the target remain within the sensor's linear range (typically in the middle of the dynamic range). The target should be positioned as close as possible to the centre of the camera's field of view, where lens shading has the least influence on luminance measurement.

Finally, the same target must be measured with the reference instrument, preferably at the same time and positioned as close as possible to the camera being characterised. The s_V parameter is then calculated as follows:

$$s_V = \frac{1}{L_{\mathrm{ref}}} \cdot \frac{(D - D_0)}{t}, \quad (7)$$

where

- t is the integration time used to capture the LDR image,
- D is the average pixel value of the LDR image on the target,
- D_0 is the average pixel value of the associated dark image,
- L_{ref} is the luminance measured on the target using the calibrated instrument.

To lower the uncertainty on the determination of the luminous responsivity, several reference targets can be placed in the scene, and/or the procedure can be repeated several times, as long as the calculation of the luminous responsivity of the camera is done in a range of exposure where the camera is linear (accurate knowledge of the integration time and linearity range of the sensor).

If a flatfield correction, used to correct shading from the lens and photon response non uniformity (PRNU) from the sensor, is known, the correction should be applied to the LDR image prior to computing the value of the luminous responsivity using Eq. (7). In that case, the same flatfield correction must be applied to the HDR luminance image produced using the HDR merging code. It should be noted that neither non-uniformity nor responsivity value affect the performance of the merging algorithm itself.

4 MATLAB code

This Section presents the MATLAB code for HDR image generation from a series of LDR images.

4.1 Code architecture

The HDR merging code has been implemented in a MATLAB script. It relies on a main script ("HDRmerge.m") and side functions corresponding to the different types of merging algorithms, placed in a dedicated folder ("HDR_algo").

The code proposes a user interface based on dialog boxes for loading the different images and parameters needed to reconstruct HDR luminance images. Some parameters, specific to the device used to acquire the LDR images, are hard coded in the main script. However, once these parameters are specified, the user doesn't need to modify the code anymore to generate HDR images from various LDR series captures.

The main script starts with a header providing the code name, authors, funding agency, open-source licence and version number, as well as a paragraph on how to use the code ("HOW TO RUN THIS SCRIPT"), summarizing each step of the processing. These steps are further described below.

4.2 Camera parameters

The HDR merging relies on camera parameters defined by the user according to the properties of the device used for image acquisition. The code has been designed to propose a list of cameras that the user can use. Three cameras with their parameters can be entered in the code in the current version. However, the user can modify the code to propose more cameras if needed. The camera name displayed in the list can be edited for a personalized user experience.

For each camera, the user must edit the main script to specify the LDR image dynamic range, file format, the minimum and maximum thresholds for defining well-exposed pixels and luminous responsivity (see Section 3):

- "nbits" is the number of bits onto which the LDR images are coded, generally 8, 12 or 16 bits.
- "sV" is the luminous responsivity of the camera (see Section 3), expressed in [count·**s**$^{-1}$·cd^{-1}·m^{2}] or in [count·**ms**$^{-1}$·cd^{-1}·m^{2}], used to convert count values in luminance.
- "Dmin" is the minimum threshold, in count, applied to determine the well-exposed pixels. It is expressed as a percentage of the total dynamic range of the instrument. For example, defining Dmin as 0.05×2^{nbits} corresponds to keeping pixel values higher than 5 % of the camera dynamic range.
- "Dmax" is the maximum threshold, in count, applied to determine the well-exposed pixels. Similarly to Dmin, it is expressed as percentage of the total dynamic range of the instrument. For example, defining Dmax as 0.92×2^{nbits} corresponds to keeping pixel values that are below 92 % of the camera dynamic range.
- "ImageFormat" is a string containing the file format extension of the LDR images. It may be format adapted to grayscale images such as .tif, .tiff, .png, or .bmp. In case the images used for HDR image generation are color rather than greyscale, a conversion into luminance must be added. An example of processing with .CR2 files is given in the code (Camera 3).

When running the code, a dialog box asks the user to select the camera (associated with the parameters defined above) that must be used for the HDR reconstruction.

4.3 Image selection

Once the user has selected a type of camera, a second dialog box asks to select the LDR image files used for the HDR reconstruction. Only files corresponding to the file format specified in the camera parameters are proposed.

The order of the selection must be the same order as the exposure times listed in the .txt file (see 4.4).

4.4 Exposure time file

The integration (/exposure) times used for the LDR images acquisition must be provided by the user in a text file (.txt). A dialog box pops up to ask the user to select the file.

The integration times should be listed in the same order as the LDR image selected. Each value is separated by a line break.

The unit in which the integration times are expressed should be coherent with the unit of the luminous responsivity parameter. For example, if the luminous responsivity sV is expressed in [count·**s**$^{-1}$·cd^{-1}·m^{2}], the integration times should be provided in **seconds**. If sV is expressed in [count·**ms**$^{-1}$·cd^{-1}·m^{2}], the integration times should be expressed in **milliseconds**.

An example file, "Texp_Orca.txt", is given in the folder "Examples".

4.5 RAW file conversion

As mentioned in 4.2, the format of the LDR images might require a conversion into a file format representing luminance values. This is the case with raw RGB images acquired with colour DSLR cameras (providing .CR2 files). A simple solution is to use the green channel of the images to represent luminance values. However, if the colour camera has been calibrated, it is possible to use the RGB to XYZ conversion matrix. In that case, luminance is proportional to the Y channel obtained after conversion.

In the proposed code, .CR2 files are converted disabling colour transformation, default white-balance correction and tone curve mapping. The green channel is used to represent luminance values.

4.6 Dark correction

The fourth dialog box displayed to the user concerns the image dark correction. The dialog box asks the user if they wish to apply a dark subtraction or not.

If dark subtraction is chosen, the user must load images corresponding to the dark. One dark image per integration time should be acquired (typically, the lens cap is placed on the instrument for this LDR series acquisition). When running the MATLAB code, dark images must be loaded in the same order as the LDR image files, so that each dark image is properly associated to its integration time.

Then, the following options may apply:

- Option 1: A pixel-to-pixel dark subtraction is performed, meaning that each LDR image is subtracted by its associated dark image, in a per pixel operation. This option is recommended only if the dark images at hand are obtained from averaging multiple dark acquisitions and present a low level of noise. Otherwise, subtracting noisy dark images to the LDR images will degrade the signal to noise ratio (SNR) of the reconstructed HDR image. However, this option is the only option that accounts for the non-uniformity of the dark offset (due to sensor non-uniform temperature for example).
- Option 2: An average value is calculated from each dark image loaded by the user. This average value is subtracted from each pixel of the LDR image acquired at the same integration time. This option discards any information on the dark non-uniformity.
- Option 3: The median value is calculated from each dark image loaded by the user. The median is subtracted from each pixel of the LDR image acquired at the same integration time. This option is similar to option 2, but using the median rather than the average provides more robust results in case of defective pixels on the sensor (such as hot pixels, that presents very high values whatever their exposure).

These options are hard coded in the main script. The 3 options are pre-written, and the user must select the most appropriate option by commenting (using '%') and uncommenting (removing '%') lines in the code.

Note that using LDR images pre-corrected from the dark offset at the image selection step might yield to HDR reconstruction errors. Indeed, the HDR algorithm relies on defining well-exposed pixels based on the pixel saturation value. If saturated pixels end up with a value lower than the upper threshold Dmax after dark correction, they will be wrongly considered as well-exposed pixels.

4.7 Choice of HDR merging algorithm

A dialog box proposes several types of HDR merging algorithms from which the user can choose from (see Section 3 for the algorithms description). The use can choose to apply all algorithms ('ALL ALGOS') and obtain as a result the 4 HDR images generated by the 4 proposed algorithms, or can choose between

- The "best exposure" algorithm ('BEST EXPOSURE'), which relies on the MATLAB function "best_exposure.m".
- The "integration time weighted average" in the linear domain ('WEIGHTED AVERAGE TINT'), which relies on the MATLAB function "weighted_average_texp.m" with log option off.
- The "integration time weighted average" in the logarithmic domain ('WEIGHTED AVERAGE TINT LOG'), , which relies on the MATLAB function "weighted_average_texp.m" with log option on.
- Or the "linear regression" algorithm ('LINEAR REGRESSION'), which relies on the MATLAB function "linear_regression.m".

4.8 HDR image saving

Finaly, a dialog box proposes to save the resulting HDR images, and the use can choose between YES and NO.

If YES is selected, the user must input a file name with no format extension. The HDR images are then saved as a .mat MATLAB file, and as .hdr files, the file format from Radiance software widely used in daylight glare studies.

Note: For the calculation of any metric that requires the solid angle (e.g., for glare analysis), the header of the *.hdr* image must contain lens information of a projection method supported by Radiance or related tools (such as *evalglare*). If the real projection of the lens differs from the theoretical projection, a correction may be necessary. Additionally, the black borders of the image may need to be cropped to ensure compliance with Radiance image standards. More details can be found in [5].

4.9 HDR image display

The reconstructed HDR luminance images are displayed with a logarithmic colormap. By default, the minimum and maximum values for the colormap are set to the minimum and maximum luminance values of the image (MATLAB function ‘imagesc’).

The user can also predefine in the main script the desired minimal and maximal luminance values of the colormap by setting the parameters “lmin” and “lmax”. The user then needs to uncomment the lines of code applying these colormap limits (% caxis([lmin, lmax]);).

5 Additional features not included in the MATLAB code

5.1 LDR series acquired by varying parameters other than integration time

The proposed code can also be used to merge LDR images into HDR when the images are acquired by varying another parameter than integration time. In that case, the proper exposure parameters associated with each image should be determined and entered as if it was a different integration time in the required text file. Hybrid methods (for example combining different integration times and neutral density filters) are also possible.

For example, if we consider 3 images captured at the same integration time *t* using 3 different gain (or ISO) values *g1*, *g2* and *g3*, the text file listing integration time will contain the 3 following values: *g1 * t*, *g2 * t* and *g3 * t*.

5.2 Non-linearity correction

Non-linearity is an important aspect to consider for accurate HDR image reconstruction. Indeed, the algorithms presented in Section 3 all rely on the assumption that the camera is linear. If the camera is not linear by a significant amount, the user can improve the HDR reconstruction results by applying a non-linearity correction as a preprocessing to the data.

Depending on the type of camera and the way the non-linearity correction was determined, the correction is applied either before or after dark subtractions. No non-linearity correction is applied in the proposed code, however, a commented line indicating where the correction can be performed is written.

In case of non-linearity on the integration time (that is, the integration times given by the camera do not correspond to the actual time during which the sensor is exposed to light), a correction on the integration times given in the text file is necessary.

5.3 LDR images acquired using a non-linear camera

The mathematical expressions for the HDR merging algorithms implemented in the code (Eqs (3) – (6)) are valid for images acquired with a linear camera. If the LDR images were captured using a camera with a non-linear response function, they must first be linearized before applying the HDR merging function. This can be achieved by applying the inverse camera response function, in the same way as a non-linearity correction would be applied. Note that dark images may also require correction using the inverse camera response function.

5.4 Additional processing steps not included in the code

For accurate HDR luminance images, two additional corrections, the flatfield correction and the geometrical distortion correction, must be applied. These corrections are not specific to the exposure value used to acquire the LDR images, they can therefore be applied identically to LDR images or HDR images. For LDR images intended to be merged into an HDR image, flatfield correction should not be applied before merging though, to guarantee that the determination of well-exposed pixels based on the saturation value remains valid (similarly to dark correction, a flatfield correction could lower a saturated pixel to a value for which the pixel might be mistaken for a well-exposed pixel).

Flatfield correction is crucial for accurate luminance measurement. Geometric distortion correction only impacts the pixels' location in the scene and should be applied for applications where knowledge of the scene geometry is required.

6 Example dataset

Two example datasets are provided with the code in the folder "Examples": a dataset titled "Canon EPFL images (.CR2)" and a dataset titled "Orca CNAM images (.png)". Each folder contains input files corresponding to a series of LDR images, with their associated dark images only for the Orca dataset (captured at the same integration times with the lens cap on). The text file with the used integration times is also provided with the input files. The HDR images obtained using the MATLAB code are placed in a "HDR results" folder.

The Canon dataset was created at EPFL by capturing an outdoor scene with a 16 bits Canon camera which provides colour images saved as .CR2 ("CAMERA 3" in the main code). Note that the resulting HDR images has not been adjusted in luminance in our example. This image is therefore proportional to the luminance. The HDR images were obtained by applying minimum and maximum thresholds of 12.5% and 70% in the HDR merging algorithms. The weighted average algorithm was applied to generate this example.

The Orca dataset was created at CNAM by capturing a lab-built high contrast scene with a Hamamatsu scientific camera (ORCA-Flash4.0 V3) with a 16 bits sensor and images saved in .png ("CAMERA 1" in the main code). For this image series, no luminance adjustment was done either, the HDR image results is therefore also only proportional to the luminance of the scene. The HDR images were obtained by applying minimum and maximum thresholds of 3% and 93% in the HDR merging algorithms. The LDR images' names start with "Orca_". A dark correction is applied (based on the median), using the LDR images named "OrcaK_". The figures in each image name corresponds to the integration time expressed in µs. The best exposure algorithm was applied to generate this example.

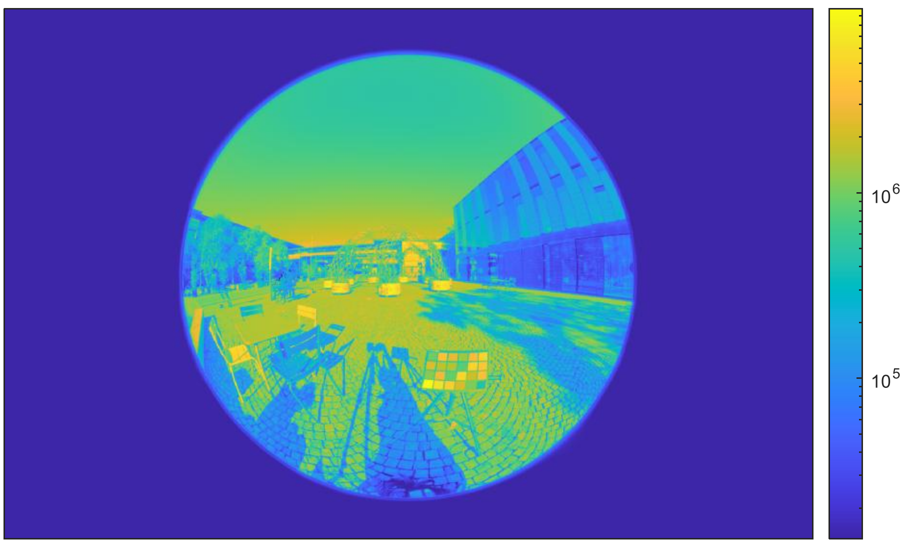


Figure 4. HDR image obtained using the Canon EPFL dataset with the best exposure algorithm.

7 Licence

The code produced in this project is released under the MIT License, a permissive open-source license that is widely regarded as the most suitable for software. The MIT License allows anyone to freely use, modify, distribute, and even incorporate the code into commercial applications, provided that the original copyright notice and license are included. Its simplicity and flexibility make it ideal for sharing code on platforms like GitHub, promoting reuse, collaboration, and transparency, while still giving proper credit to the original author. By choosing the MIT License, this project ensures that the code can be broadly adopted and integrated into other research, educational, or commercial projects without legal ambiguity.

A licence file is placed in the code folder.

8 Code availability

This project's code is available on GitHub for download, use, and collaboration. The project URL address is : https://github.com/HiDyn-EURAMET-EPM-21NRM01/HDRmerge_HYDIN .

By sharing the repository on GitHub, users can freely download, explore, and contribute to the code, as well as report issues or suggest improvements, while ensuring version control and transparency.

9 Conclusion

High dynamic range (HDR) imaging is essential for accurate, SI-traceable luminance measurements in high-contrast scenes typically encountered in glare analysis and obtrusive light evaluation. In the 21NRM01 HiDyn project, four reliable HDR merging methods capable of full uncertainty propagation were chosen and implemented in an open-source MATLAB toolbox: best-exposure, weighted averaging (linear and logarithmic domains), and linear regression.

The user-friendly code handles LDR image series, exposure times, dark correction, raw conversion files, and camera calibration. The outputs are HDR images saved in MATLAB files and in .hdr files, which is the format widely used in daylight glare studies. Practical guidelines are provided for determining luminous responsivity via traceable spot measurements and for correctly applying additional corrections (flatfield, distortion, non-linearity).

This toolbox helps the metrology and lighting communities create reliable HDR luminance maps using commonly available cameras. It combines everyday HDR imaging techniques with strict measurement requirements, making standardised work in road lighting, glare testing, and light-pollution studies easier.

10 References

[1] P. Debevec, E. Reinhard, G. Ward, and S. Pattanaik, “High Dynamic Range Imaging,” *High Dynamic Range Imaging*, p. 276, 2004.

[2] L. Gevaux, A. Ferrero, A. Dupiau, C. Bouroussis, and J. Ledig, “High dynamic range imaging methods for traceable optical measurements,” presented at the CIE Midterm session, Vienna, 2025.

[3] F. Banterle, A. Artusi, K. Debattista, and A. Chalmers, “Advanced high dynamic range imaging, 2nd Ed.,” *AK Peters/CRC Press*, 2017.

[4] M. A. Robertson, S. Borman, and R. L. Stevenson, “Dynamic range improvement through multiple exposures,” in *Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348)*, Kobe, Japan: IEEE, 1999, pp. 159–163. doi: 10.1109/ICIP.1999.817091.

[5] Pierson, C., Cauwerts, C., Bodart, M., & Wienold, J. (2020). Tutorial: Luminance Maps for Daylighting Studies from High Dynamic Range Photography. *LEUKOS*, 17(2), 140–169. https://doi.org/10.1080/15502724.2019.1684319,