

## Efficient Simulation for Visual Comfort Evaluations

Stephen Wasilewski<sup>a,b,\*</sup>, Lars O. Grobe<sup>a</sup>, Jan Wienold<sup>b</sup>, Marilyne Andersen<sup>b</sup>

<sup>a</sup> Lucerne University of Applied Sciences and Arts, Horw, Switzerland

<sup>b</sup> École Polytechnique Fédérale de Lausanne, Switzerland



### ARTICLE INFO

#### Article history:

Received 30 November 2021

Revised 18 April 2022

Accepted 25 April 2022

Available online 4 May 2022

#### Keywords:

Daylight

Visual comfort

Glare

Climate based daylight modeling

Rendering

Lightfield

Validation

### ABSTRACT

This paper provides a validation of a novel sampling, storage, and evaluation method named *raytraverse* that can quickly and accurately compute glare and visual comfort metrics including vertical illuminance ( $E_v$ ), Daylight Glare Probability (DGP), and Unified Glare Probability (UGP). The motivation is to provide a path towards understanding the spatial and temporal distribution of daylight conditions in an architectural space. Current spatial temporal simulation workflows are constrained by the trade-offs between simulation time, accuracy, generality, and storage requirements. *Raytraverse* provides a bridge between illuminance sensor calculations, which are fast to calculate but provide limited information, and high-resolution image generation, which provide more information but have long simulation times. To make this bridging possible, it relies on a pair of strategies that yields both high accuracy and high information data. First, an iteratively guided sampling approach based on the discrete wavelet transformation greatly reduces the number of view rays submitted to the renderer. Second, rather than collecting returned values in a raster image or dense matrix, a spatial data structure is used to both sparsely store and re-weight the samples according to their effective solid angle, allowing for the direct integration of any view based lighting metric and the synthesis of interpretable high dynamic range images (HDRi). These strategies can be coupled with existing rendering and climate based daylight modeling (CBDM) methods. Through a comparison with high-quality reference simulations and a best practice CBDM method based on *Radiance*, the *raytraverse* methods are shown to significantly reduce the simulation time needed to accurately simulate saturation, contrast, and combined visual comfort metrics for a complete set of annual hourly sky conditions from a range of locations within an office floor plan. The stored simulation data can be quickly re-analyzed for different view directions, metrics or images, and sky conditions.

© 2022 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

### 1. Introduction

A primary task when using simulation based analysis for daylight planning is determining where, when and in how much detail to look across the space under evaluation. Because daylight, building occupation, and building control are not fixed states, the goal of any prospective simulation is not to predict particular conditions experienced at individual time-steps, but to use a set of possibilities to understand something about the future buildings performance. How this domain of states is chosen impacts the resulting distribution of predicted lighting values and can have as large of an effect on the results as any particular scenario being studied. The more constrained a task is by computing resources, the harder it is to select a domain that will be representative.

The evolution of simulation protocols for daylight performance standards illustrates how an increase in computing power enables a larger analysis scope. The motivation for moving from a daylight factor calculation at a typical point, to a room of points, to multiple sky conditions, to annual hourly conditions, is to get closer to a representative distribution. Without this broader domain, it is impossible to quantify the impacts (which are both spatial and temporal) of building orientation, fixed shading, dynamic facade operation, and interior layout. Without efficient computation, evaluating this broader domain would be impractical. A focus on speed or accuracy alone cannot answer the question of how useful a simulation method is. Instead, it is the relationship between simulation time and accuracy, which we call efficiency, that determines utility. The most accurate method that takes weeks to simulate with computers consuming energy 24 hours a day is not useful for quickly analyzing design iterations. Likewise, a fast method that introduces significant error or bias is also not useful. The efficiency of any given method is not a fixed relationship, and may vary depending on the scene, parameters, target accuracy, target time,

\* Corresponding author at: Lucerne University of Applied Sciences and Arts, Horw, Switzerland.

E-mail address: [stephen.wasilewski@hslu.ch](mailto:stephen.wasilewski@hslu.ch) (S. Wasilewski).

## Nomenclature

### Symbols and Abbreviations

$E_v$	vertical illuminance ( <i>lux</i> )	DGP	daylight glare probability
$\bar{L}$	average luminance ( $cd/m^2$ )	MAE	mean absolute error $\sum_i^n  x_{i,test} - x_{i,ref} /n$ , where $n$ is the number of cases, $i$ is the case, <i>test</i> is the value of $x$ according to the <i>test</i> method, and <i>ref</i> is the value according to the reference method
GCR	a unitless measure of contrast in the field of view, defined in Eq. 1	MAPE	mean absolute percentage error $100 * \sum_i^n  x_{i,test} - x_{i,ref} /x_{i,ref}/n$
MAE <sub>dn</sub>	distribution normalized mean absolute error, see Eq. 2	MSD	mean signed deviation $\sum_i^n (x_{i,test} - x_{i,ref})/n$
MAPE <sub>dn</sub>	distribution normalized mean absolute percentage error, see Eq. 2 with: $\Delta x_i =  x_{i,test} - x_{i,ref} /x_{i,ref}$	MSPD	mean signed percentage deviation $100 * \sum_i^n (x_{i,test} - x_{i,ref})/x_{i,ref}/n$
MSD <sub>dn</sub>	distribution normalized mean signed deviation, see Eq. 2	RMSE	root mean square error $\sqrt{(\sum_i^n (x_{i,test} - x_{i,ref})^2/n)}$
ASE	annual sunlight exposure	TMY	typical meteorological year
BSDF	bi-directional scattering distribution function, here specifically referring to the material primitive implemented in <i>Radiance</i>	UDI	useful daylight illuminance
CBDM	climate based daylight modeling	UGP	unified glare probability
DDS	dynamic daylight simulation		

available hardware, and the spatial and temporal scope of the analysis.

The major limiting factor for capturing the spatial and temporal distribution of daylight conditions within a space is the exponential growth of the analysis domain known as the curse of dimensionality. Fully characterizing daylight is up to an eight dimensional problem, grouped into four by their type: position (3), direction (2), source direction (2), and spectrum (1). Since all of the target metrics currently rely on photometric integrals, the spectral dimension is not considered here. Sampling this domain uniformly means that adding a single step in one dimension propagates to all of the other dimensions. This paper shows how the *raytraverse* sampling method, which has previously been shown to accurately sample directional view rays [1], can be generalized and applied to sampling source directions and positions within a space. By adaptively sampling these three dimensional sets (position, direction, source direction) separately, the relationship between the magnitude of the analysis domain and the effective resolution of each axis is no longer multiplicative: adding additional samples in one part of the sample space does not require calculating those samples along the entire axis. Each point has its own pattern of sampled sun positions, and each source has its own pattern of directional sampling. While this paper presents the method for adaptively sampling position, source and direction, the validation uses predefined sampling locations to match the reference simulations. Future work will be needed to validate adaptively sampling across positions.

To validate these – or any – new methods for an annual temporal domain requires careful consideration of the cases included in the validation. Each site, building, orientation, and point within a space will experience its own unique distribution. Daylight simulation methods tend not to have the same errors, magnitude or type, for different daylight pathways (direct/indirect, sun/sky, transmission/reflection) and conditions (high/low, contrast/brightness). The over or under representation of a particular condition in the validation set, relative to any future use cases, can skew the expected accuracy. To control for this, the scope of the validation needs both to be broad enough to cover expected conditions and the reported error statistics need to control for the distribution of the validation set. This paper presents a simulated reference validation procedure that bins the error according to the reference conditions using the target metric. This yields normalized error statistics for assessing overall accuracy and itemized error reporting to determine the reliability of the methods across conditions.

### 1.1. Climate based daylight modeling in visual comfort studies

Climate based daylight modeling (CBDM) uses daylight coefficients as an efficient means to compute hourly values that span a time period representative for the local climate. CBDM and its corresponding annual metrics were originally motivated by the need for quantitative predictions of daylight supply, measured as horizontal (work-plane) illuminance, as a supplement for electric lighting [2,3]. Sharing Typical Meteorological Year (TMY)<sup>1</sup> data as a common basis with general Building Performance Simulation (BPS), CBDM helps account for the beneficial effects of daylight on the energy demand of buildings [5].

The increased efficiency of electric lighting and the observation that the expected beneficial effects of daylighting were often not realized, motivated an increased interest in the interplay of visual comfort in naturally illuminated spaces and energy demand [6]. A tempting approach was to simply reinterpret the simulation results produced for BPS as proxies for visual comfort, e. g. by defining upper boundaries of beneficial ranges of horizontal illuminance (as in Useful Daylight Illuminance, UDI [2]) or testing for direct exposure to sunlight (Annual Sunlight Exposure, ASE[7]) [8]. Research however showed that horizontal illuminance is a poor proxy for visual comfort and in particular for glare [9].

As the field has moved to recognizing that the quality of daylight is as important as the quantity of daylight, performance standards now include view based glare metrics (such as the CEN daylight standard EN 17037:2018). Compliance is demonstrated either through precomputed tables or on-site evaluations of extreme conditions. For example, the tables in the CEN standard are made from annual simulations from a single point looking at the 5th percentile of glare occurrence at a worst case point. This is because the calculation of the most widely adopted and validated glare metrics, DGP in controlled window adjacent studies [10] and UGP in extensive field studies of open offices [11], typically require HDR images, whether captured or simulated, for calculation. The computational methods that drive CBDM methods can be used to accurately produce these images over the course of a TMY[12], but calculation time typically constrains the extent of such an analysis. Techniques exist that take a hybrid approach

<sup>1</sup> TMY data are hourly annual weather conditions for a given location based on historical conditions at that weather station. Different standards exist for the range of years included and the criteria used for selecting typical conditions. See [4] and ISO 15927-4:2005.

supplementing illuminance calculations with direct sun (and in some cases specular reflections) to capture the most common contrast scenarios. One of these techniques is eDGPs [13], the general principles of which have since been implemented in a number of *Radiance* [14] front-ends. While this method can be accurate, it is dependent on the accuracy of the externally calculated illuminance and is still not practical to compute across a large spatial domain. More recently two other approaches offer significant time savings. *ClimateStudio* is a widely used software package that includes an annual glare calculation. According the documentation (<https://climastudiiodocs.com/docs/annualGlare.html>), "For annual DGP simulations, ClimateStudio relies on the vertical illuminance portion of the DGP formula, plus a contrast measurement from the solar disc." This approach, with proper settings, could be accurate under certain circumstances, but will not account for other glare metrics, rough specular transmissions, bright sky, or diffuse and specular reflections that may act as glare sources. Imageless DGP [15] is a matrix based method that is included with *Radiance* as the *dcglare* executable. This method may not account for glare sources from reflections or through low transmission glazing or shading.

## 1.2. Daylight simulation methods validation

The models, renderers, and workflows commonly used in daylight simulation have a long history of validation, both referenced against physical measurements and in comparison to a reference simulation. Measurement based validations are critical to make sure that simulations accurately reflect the physical world, are necessary for processes where a simulated reference does not exist for the model being tested, and are a prerequisite for simulated reference validation, like the one presented in this paper.

Different validations capture the error over different parts of the simulation process, Understanding which parts of a simulation process and what additional confounding errors may exist is important for properly contextualizing the reported error. The captured error of the validation presented in this paper is shown relative to that of some notable validations in Fig. 1. Six studies [25,17,18,26,19,20] validate daylight coefficient based simulations using *Radiance* against physical measurements. The captured error of all include: geometry, material modeling, light propagation modeling, simulation algorithm, and simulation parameters, and is confounded by measurement error. While the methods differ between the studies, all found similar magnitude mean signed deviations, MSD, (10%) and root mean square errors, RMSE, (20%) for horizontal illuminance.

Other validations have taken a different approach to reporting error. Lee et al. [21] did a validation study of image based 3-phase and 5-phase methods, against both measured data and a reference simulation method (the reference simulation only includes illuminance sensor calculations). Here, error is quantified by its distribution, the percentage of cases within 5%, 10%, and 20% relative error. Brembilla and Mardaljevic [3] focus on checking the accuracy of annual summary metrics, like UDI and ASE. While the daylight availability metrics are in general not very sensitive, they found that daylight coefficient methods that do not account for the true source size are not well suited to tasks requiring precise angular specificity.

In addition to point in time illuminance validations, there are also image based validations. Grobe [24] validates the use of the *Radiance* photon-mapping implementation [27] coupled with high resolution bi-directional scattering distribution functions (BSDF). Like Lee et al.[21], this study validates images both in terms of direct photometric quantities and discomfort glare metrics. Jones and Reinhart [22] perform detailed assessments of the luminance distribution for each of the assessed methods. In validating cap-

tured HDRi simulations vs. GPU based *Accelerad* and *Radiance*, they found that measurement and modeling errors play a larger role than errors introduced by simulation, although it is important to note that both simulation methods used the most physically accurate representations of solar source and indirect light propagation. This confirms an earlier finding by Reinhart and Andersen [19] where after partially correcting for the sky measurement and modeling error, RMSE percentage errors for their proposed translucent material model went from 14–18% to 8–10%.

While the summary error metrics in all these studies give general confidence in using CBDM with *Radiance* to understand something meaningful about building performance, they do not on their own suggest the source of the errors. This is important in the context of developing new simulation methods. In the case of a non-uniformly distributed errors, if the simulation method is applied to a case with a different distribution of conditions, the error for that case could be much different. To understand this distribution of errors, these validations use a number of strategies, including: excluding high consequence geometric errors where the photocell may be in sun while the virtual sensor is not and vice versa [17], and scaling the results based on the light incident at the facade to control for the sky model [18,19]. Several of these studies report errors separately by sky-type, including Reinhart and Walkenhorst [18] and Mcneil and Lee[20]. In Mardaljevic[25], the MSD between the six test points differs, where there is a tendency towards an under estimation of illuminance at points that receive more direct sun and a tendency towards over estimation of illuminance at points that receive less direct sun. This reveals the dependence of the error on the distribution of conditions evaluated.

This study validates *Raytraverse* and a comparison method against the best available simulated reference, using what is typically referred to as "*Radiance* classic," meaning the *rpict* executable, without employing any of the approximations that help make CBDM so much more efficient than this brute force approach. Returning to the idea of captured scope shown in Fig. 1, what this means is that we are attempting to minimize the error introduced between a hypothetical measurement and the reference simulation. Through this, the captured scope of the validation presented here overlaps with these physical measurement validations, so it is neither that any error here directly adds on top of these errors, nor that it avoids these errors. In the case of small deviations between test and reference, the strongest conclusion we could make is that the test method is as close to as accurate as the reference.

## 2. Simulation methods

One of the goals for developing *raytraverse* is to find more efficient ways to simulate daylight that compliment existing, ongoing, and future developments in CBDM. To realize this goal, it is helpful to view the daylight simulation process as a series of components. Fig. 2 illustrates these components and shows that *raytraverse* is focused on *which* samples are rendered and evaluated, not *how* they are modeled or rendered. This means that the methods detailed here can be integrated with other innovations, such as GPU based rendering [22] or photon-mapping [27], to potentially offer compounding gains in simulation efficiency. The core of the method is an adaptive sampling approach that reduces the amount of work that the renderer needs to do. Associated with this are the methods needed to evaluate the sparse representation of the traversed domain that this sampling provides.

Before broaching the details of *raytraverse*, readers familiar with *Radiance* may find the following description of Fig. 2 in terms of the corresponding *Radiance* executables helpful. Beginning with the input, parameters are selected that control various steps in the

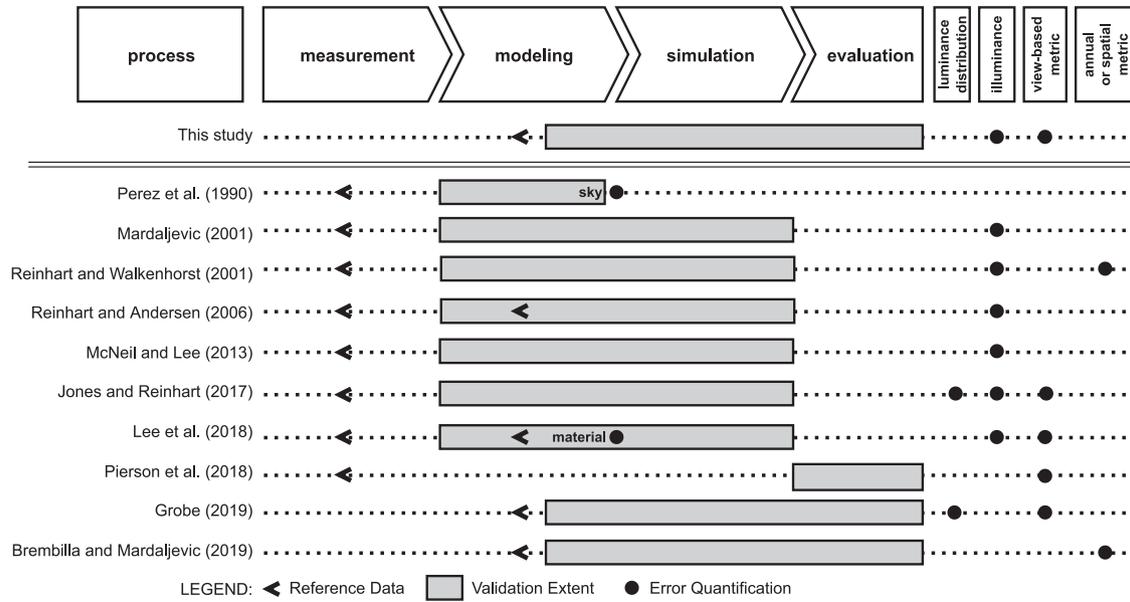


Fig. 1. Validations are shown with the approximate extent of the process covered. In a comparative assessment of errors, process outside of this extent is the same across the comparisons.[16–24,3].

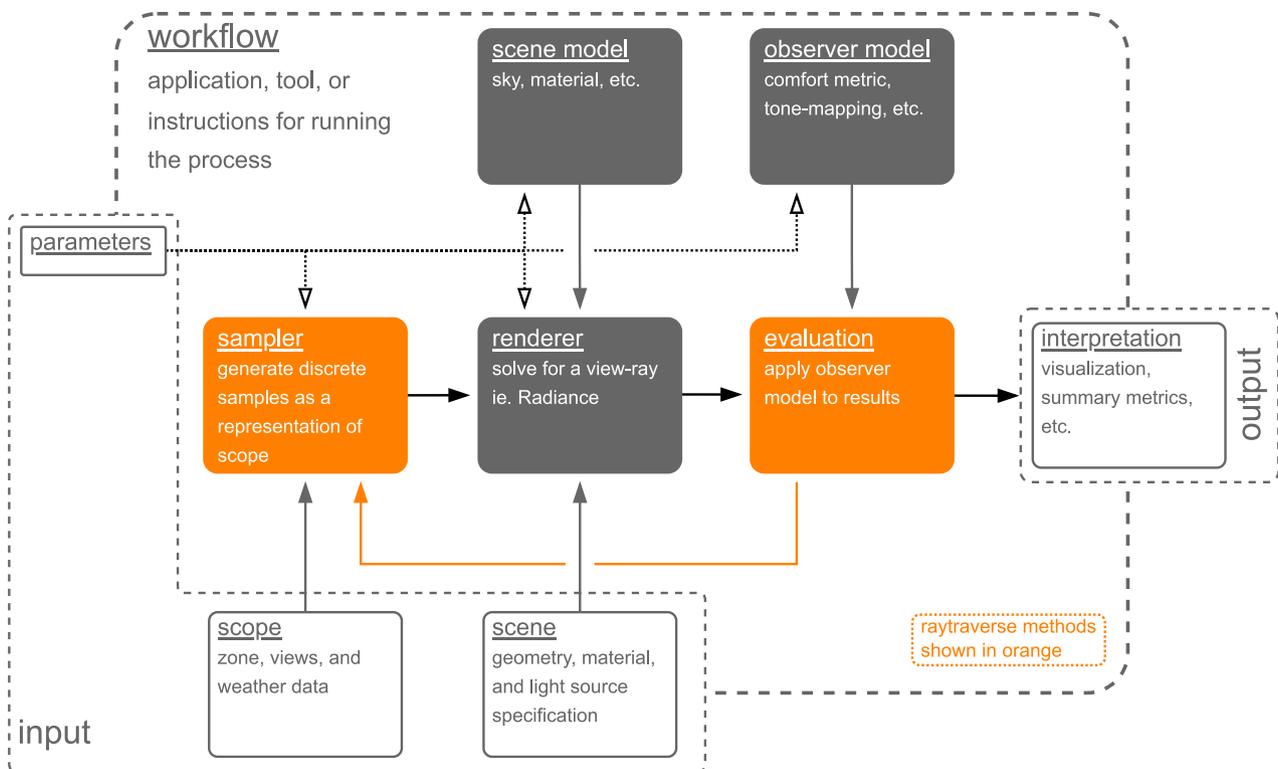


Fig. 2. Diagram of a daylight simulation process. In many cases, view-ray sample generation is embedded in the renderer (such as Radiance’s rpic1) or is statically determined as an input (rtrace). Raytraverse generates samples, including view-rays and source directions, based on the partial output from the renderer. For sampling across positions and source direction, interim evaluation is also used.

workflow (dashed lines with open arrows). For example, the image resolution would be fed to the sampler, ambient sampling parameters to the renderer, sky discretization to the sky model, and glare source thresholds to the observer model. The scope (where to look) and scene (what to look at) are similarly statically described by

sensor points or view files and Radiance scene files (this input information is shown as grey filled arrows). Moving to the simulation process, using Radiance in this diagram, the sampler would be the vwrays program, which takes a view specification, defined by the scope, and passes view rays to the renderer (shown as black

filled arrows denoting it is part of the output data). The renderer would be the *rtrace* or *rcontrib* program<sup>2</sup>, which would solve for the luminance in the direction of the view rays, based on the provided scene, and the sky and material models (coded by *Radiance* primitives, data, etc.). These luminance values are then output as a HDRi suitable for further evaluation. The *evalglare* program takes this luminance data, applies an observer model, and outputs a view based metric ready for visualization, statistical analysis or other interpretation:

---

```
# sampler
vwrays -ff -vf view.vf -x 1000 -y 1000 > viewsamples
# renderer
rtrace -x 100 -y 1000 -ld- -ffc scene.oct
  < viewsamples > view.hdr
# evaluation
evalglare view.hdr
```

---

*Raytraverse* breaks this uni-directional flow of data from input to output, indicated by the orange filled arrows in Fig. 2. The *raytraverse* samplers, which operate across direction, position, and solar source, rely on intermediate evaluation of the rendered results to guide subsequent view and position rays, and in the case of solar source sampling, additional sun positions to render. The following sections describe the sampler and evaluation components in more detail.

### 2.1. Wavelet guided adaptive sampling

A wavelet transform can characterize a sampling space by the rate and magnitude of changes in the distribution of the incident light. *Raytraverse* uses this structure to construct probability distribution matrices from lower resolution estimations in order to target high-variance locations for additional sampling [1]. Wavelet guided sampling has been previously utilized for generating photo-realistic renderings, including by Overbeck et al. [28]. They also use a wavelet representation to construct a sampling probability, but the methods have some key differences. Overbeck et al. use the sampling to determine the number samples needed for each result and sample the full wavelet domain simultaneously; in order to sample a low detail region, multiple samples are generated to cover the region and then averaged together. Instead, *Raytraverse* treats individual samples, regardless of the resolution being sampled, as discrete independent samples. For the purposes of constructing sample probabilities, they serve as the mean for the current sampling resolution. This is a poor way to estimate a region, but the sample is more valuable for final reconstruction if it is not associated with a particular sampling level, and in practice the estimate works well. *Raytraverse* uses the sampling to determine the number of output results, while Overbeck et al. use a fixed sampling budget. Additionally, *raytraverse's* approach to identifying regions of high variance is employed not only in image generation but to all dimensions, including view regions, sensor grids, and time-steps.

The general wavelet guided sampling process is described in Fig. 3. The initializing vectors must be stratified across the sampling domain at a sufficient resolution to capture the major scene details. Unlike uniformly sampling, where samples can be drawn from anywhere in the space, stratified samples are drawn into layers (strata) along each dimension, forming a jittered grid in two dimensions as shown in the top left of Fig. 4. This is a common

variance reducing technique in Monte Carlo sampling, but is also used here so that each sample corresponds to a matrix element in the detail matrix. In the case of angular space sampling, including both directional view sampling and solar source sampling, samples are from a square grid representing a hemisphere. Using the Shirley-Chiu disk to square transform [29] provides a convenient way to both draw evenly spaced samples, as the transform preserves fractional area, and to apply the high pass filters across each axis orthogonally. We can use the details found in the square sampling space as a suitable representation of detail in spherical space because the disk to square transform is bi-continuous (points remain adjacent in both directions) and low-distortion (a chair will still look like a chair).

Returning to the initial vector samples in Fig. 4, they are solved for by the renderer. In the case of directional sampler, the result of each vector is a luminance value (or sky coefficients), but in the case of sampling across position or source direction, the result will be a complete description of the incident light (a vector comprising samples of the angular luminance distribution). In other words, to sample across sources and positions, each sample initiates its own directional sampling. These results, in addition to being saved as part of the final result, are used to help determine subsequent samples. Continuing through Fig. 4, the results from the initial vectors (top left) are used as an estimate of the lowest resolution scale coefficients of the wavelet transform (top middle). These are stored in a weight tensor that holds, for each cell at the current resolution, the features used to estimate the detail coefficients. In the case of a single source sampler, the single value returned by the renderer can be used directly to update the weights. For a sky-patch sampler, where multiple values are returned for each view-ray, the maximum patch value (coefficient) is used. For a solar source sampler or area sampler, where the engine returns the complete distribution of light incident at a point, several metrics must be computed to populate the weights and ensure that different forms of variance are captured.

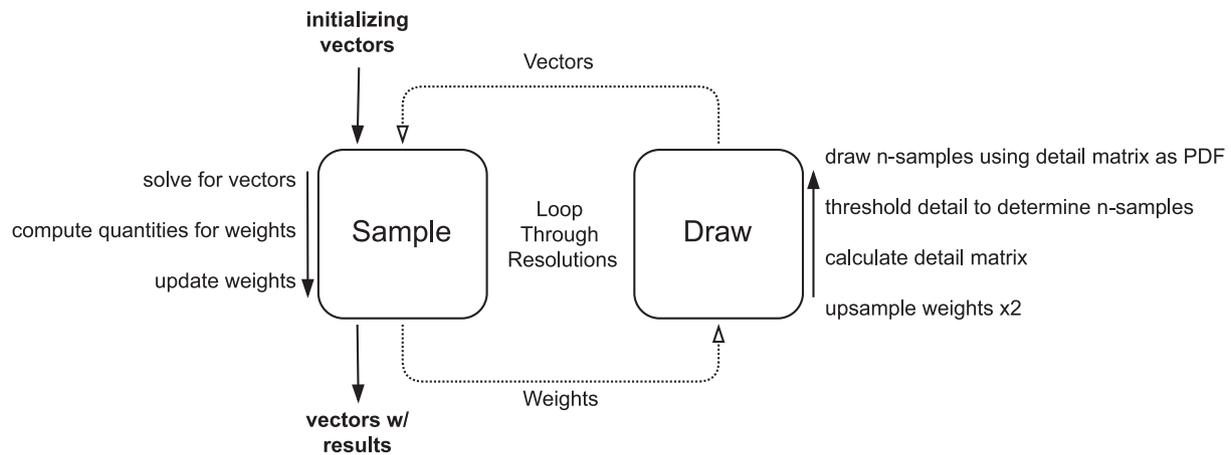
Once the weights are updated, it is up-sampled by a factor of two to the next resolution of the wavelet transform. A detail matrix (top right) is calculated by convolving the weights with the detail filter-bank, summing the magnitudes, and in the case of multiple features, taking the maximum detail coefficient. This detail matrix is thresholded to determine the number of samples. The threshold is lower initially (choosing more samples) and then grows at higher sampling levels where the relative importance of each additional sample is lower. The same detail matrix is then used to draw the next round of samples (second row left). The rendering results from these samples update the weights, and the process repeats to a fixed final resolution. The final sampling density is governed by the threshold parameter and this process, not by the stopping resolution. In this way *raytraverse* automatically adjusts the sampling across all dimensions independently.

Applying this method to sampling sun positions requires deciding which scalar properties of light arriving at a point are important for deciding whether to resolve additional sources. Testing showed that both a measure of brightness and a measure of contrast were needed to cover expected scenarios. The average luminance (chosen since there is no dependence on view direction) is used alongside a global contrast ratio:

$$\text{GCR} = \log \sum_{i=1}^n \frac{L_i^2 \omega_i}{\bar{L}^2 \omega}, \quad (1)$$

where  $L_i$  and  $\omega_i$  are the luminance and effective solid angle of each incident ray,  $\bar{L}$  is the average luminance of all rays (weighted by  $\omega_i$ ), and  $\omega$  is the total solid angle ( $4\pi$  in the case of full-spherical sampling). GCR expresses the contrast from a point in a similar form to many glare equations, but it is completely independent of intensity

<sup>2</sup> *rpict*, which is not employed by *raytraverse*, combines both the sampler and renderer



**Fig. 3.** The General wavelet sampling process. Vectors describe the dimensions being sampled, such as view-rays, view-points, or solar source directions. Weights are the quantities used to calculate the detail coefficients, in the case of a single source sampler, this is just the luminance result of each view-ray.

and does not require any glare source detection. This is critical because GCR is being applied to a coefficient result that does not have meaningful units. Fig. 5 shows an example of how the sampling evolves. Note that in the bottom left image showing the complete sampling, there is a cluster of additional sampling to the west (right side) induced by the specular reflection of light off of the east glazing after being transmitted through the south glazing. This potential pathway is highlighted on the image in magenta and was derived from the sky patch sampling results.

Applying wavelet sampling to positions requires first deciding which dimensions to sample. For most conventional building analyses, this typically means sampling a horizontal plane located at some average eye height (seated or standing). This horizontal plane estimates everywhere an occupant's eyes might be across a level zone. In the case of source direction sampling, measuring the variance in source direction would undermine the sampling, but for area sampling it can be important to determine the direction towards the light sources from a point. For example, consider sampling a sky in a room with windows on opposite walls. From the middle of the room, a step towards either direction will change the relationship to each of the windows but the overall contrast and brightness would be the same when measured by the sum of the sky-patch coefficients. However, looking ahead to evaluation, these points could be different once a sky vector is applied and a view direction is selected. Working within our implementation requiring positive scalar features, in addition to average luminance ( $\bar{L}$ ) and GCR, the horizontal peak direction, as separate cartesian direction coordinates  $x$  and  $y$  translated and scaled from  $-1 \leq x \leq 1$  to  $0 \leq x \leq 1$ , is included. The peak direction is the normalized vector sum of all rays within 10% of the maximum value. Fig. 6 shows how this positional sampling resolves sampling for different source directions.

## 2.2. When the wavelet misses

The wavelet based sampling process can capture most of the photometrically important information in typical architectural scenes as long as the initial sampling resolution is high enough that it is smaller than the apparent size of the windows. With this initial sampling, it will find smooth gradients, sharp transitions, and rough specular transmission and reflection. What this approach cannot do is reliably find small point sources, like the direct sun and specular reflections of the direct sun. In a similar spirit to the iterative sampling – which posits that if you already know

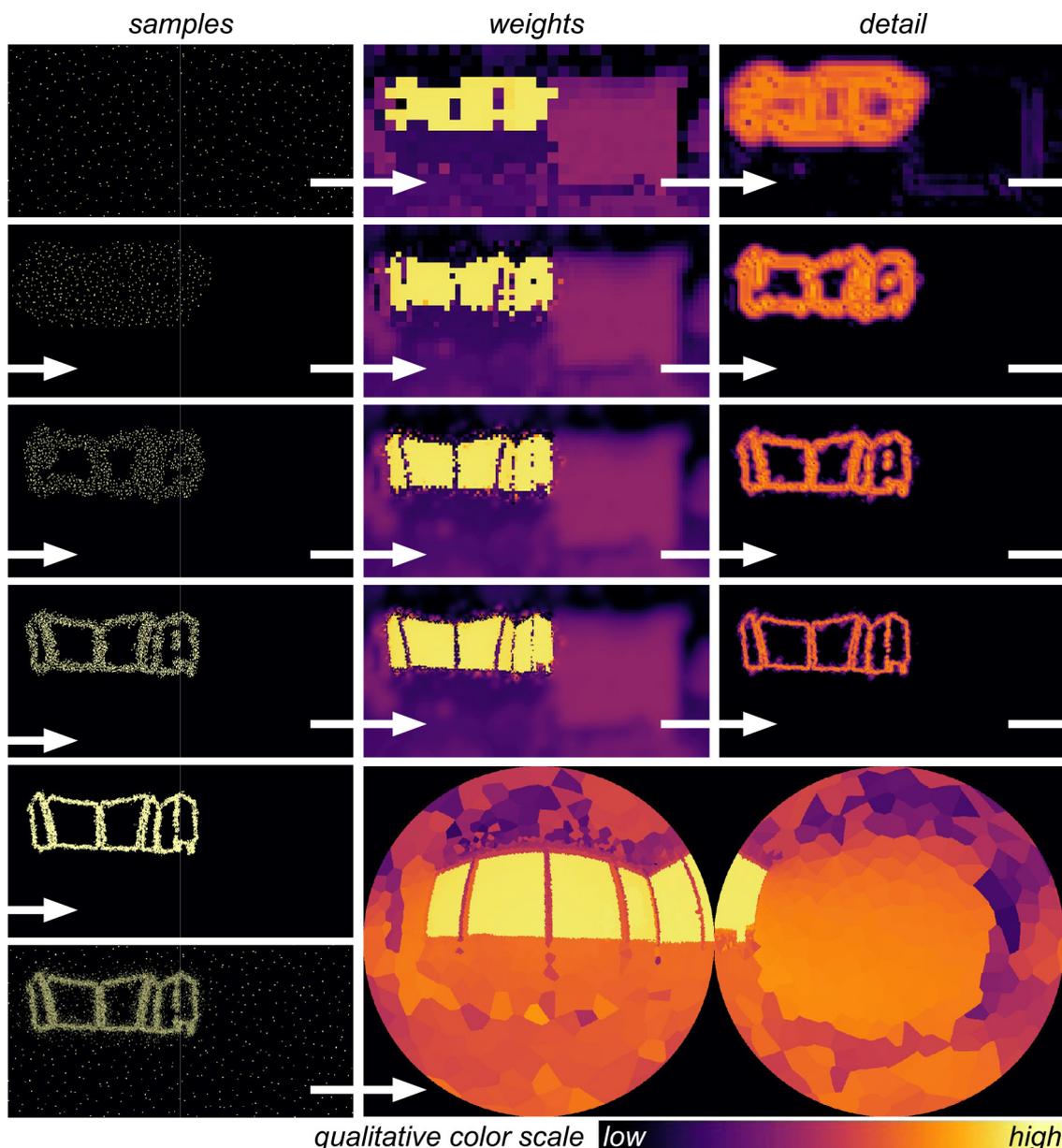
something about lighting conditions, the sampler can use that information – *raytraverse* can use two pieces of pre-existing information prior to sampling for a source direction at a point. First, the source direction is known, so it is easy enough to sample the direct view to the sun separately. Because this can be such a high impact contribution to glare metrics, which can be sensitive to small errors in source solid angle, and because this sampling can be done without any inter-reflection calculation, the direct view of the sun is sampled at a high resolution, with between 256 and 4096 samples sent within the solar subtense to capture even small details like a dot frit or partially occluded views. Second, as described in Wasilewski et al.[1], if a sky patch sampling has already been completed for the point being sampled (or the nearest available when adaptively sampling an area), then the source patch coefficient can be used to guide a search for specular reflections. When the sampling of a given source for view directions reaches an effective resolution of 512x512 samples per hemisphere, the sun patch coefficients from the sky sampling can be used as an additional probability distribution from which to draw additional samples. Any coefficient with a value over the predefined specular sampling threshold will be populated with rays at this resolution, which is fine enough to ensure that at least one ray will hit any fully visible sun reflection. Once seeded with this initial hit, the final level(s) of view direction sampling will refine this detail with the normal wavelet process.

## 2.3. Data storage and evaluation

Assuming the results of the sampling process successfully capture the necessary detail in the scene, the problem of how to store and evaluate this data remains. A typical CBDM workflow requires vector operations on each view-ray or pixel for each evaluated time-step. Then to calculate a metric, each result (across the full temporal domain) is evaluated again. If the results include all of the pixels in a full-sized image, this represents a substantial time cost that can, for simple scenes and large numbers of time-steps, exceed the rendering time. While image reconstruction from non-uniform samples can be used as an efficient rendering technique, for CBDM it is more efficient to evaluate the samples directly.

### 2.3.1. Storing Light at a Point

Analogous to an image or set of images covering all directions from the point, *raytraverse* stores the non-uniform sampling in



**Fig. 4.** Adaptively sampling directions from a single point for sky patch contributions. Samples generate weights, which are used to find details, which are used to generate more samples. When the sampling is complete, all samples (bottom left) are used to describe the complete distribution of incident light at a point. Sample, weight, and detail images are shown in Shirley-Chiu square coordinates. The visualization of the final sampling result is shown with two angular fisheye views.

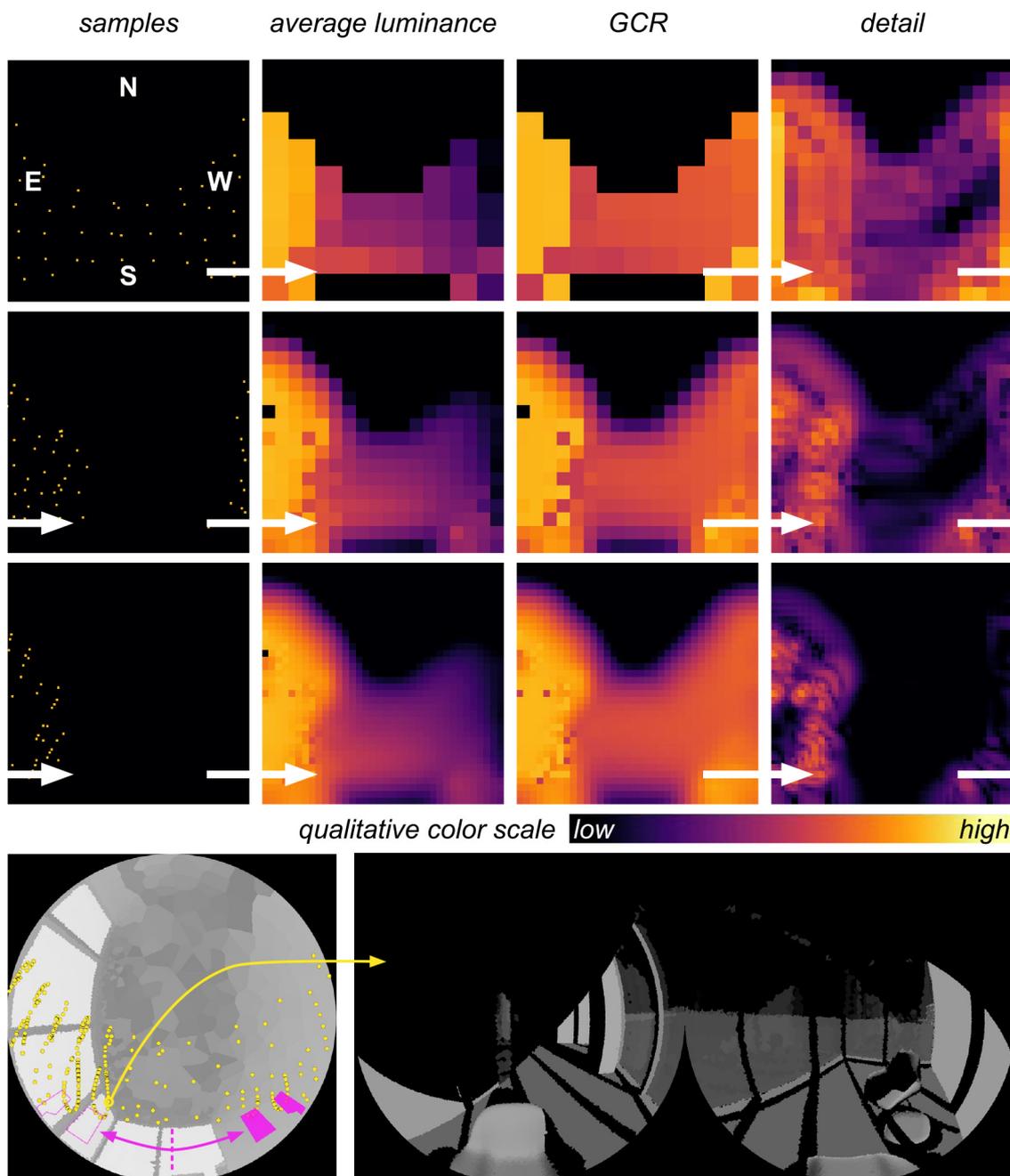
an object called a LightPoint. The LightPoint represents all light incident on a point in the assessed space and manages two challenges with storing non-uniform data. To integrate view based metrics or synthesize images, the results need to be located (by direction relative to view direction) and weighted (by effective solid angle). With an image and a defined view specification, view direction and solid angle can be recovered by knowing the pixel location, as in the *Radiance* executables *pcomb*, *ximage*, and others. In *raytraverse*, there is no predefined pixel coordinates, so rays are stored with a direction, a solid angle, and a value. The value could be a luminance, a source coefficient, or a vector of source coefficients. To efficiently find relevant rays, a kd-tree<sup>3</sup> is built from

<sup>3</sup> The kd-tree is implemented by *scipy* [30]. This implementation provides methods to quickly return the closest result to a vector or all of the vectors within a search radius (view-angle).

the direction vectors. The solid angles are calculated using the Voronoi regions<sup>4</sup> of each direction vector on the unit sphere.

With component (e.g. daylight coefficient) based sampling it is necessary to add the components together, in *Radiance* this is done with the *dstimestep* and *pcomb* programs. In the case of energy conserving operations like vertical illuminance ( $E_v$ ) or  $\bar{I}$  results can be calculated separately and then added together. Likewise, when synthesizing an image, it is also possible to interpolate each component individually and then add the result. When calculating con-

<sup>4</sup> Mitchell [31] investigated this problem for the purposes of super-sampling to combat aliasing in rendered images. They mention that a satisfying, but impractical approach would be to use the Voronoi areas of the samples. In their case, the impracticality stems from attempting to do this on a per pixel basis for an entire image, with very little down-stream payback. For this application the relative cost of building the Voronoi regions is small compared to the savings for simulation and evaluation time. The spherical voronoi routine used in implemented by *scipy* [30], which calculates spherical polygons, correctly summing to  $4\pi$ .

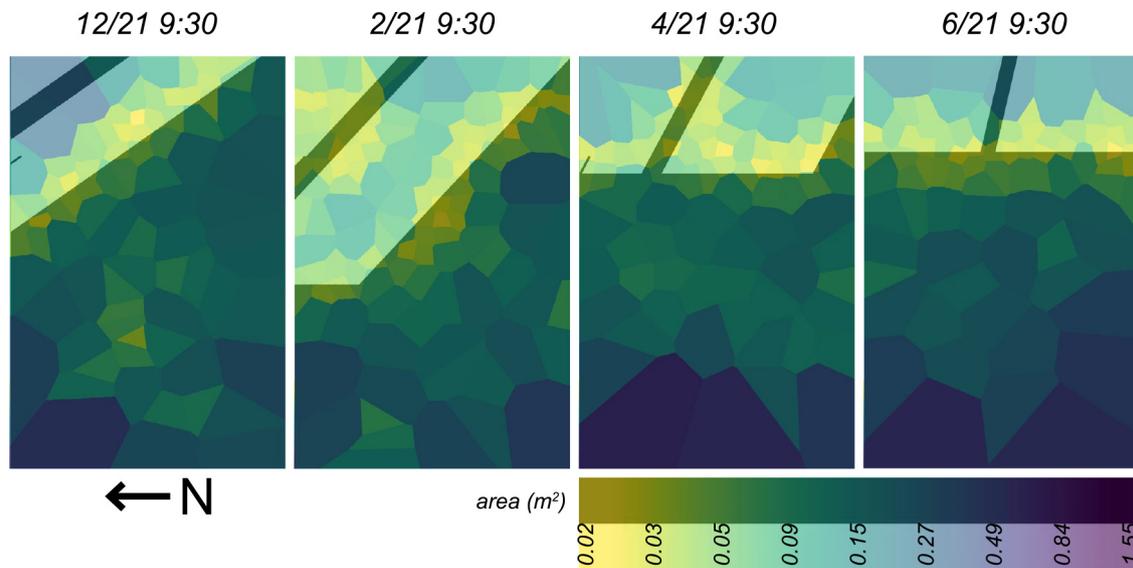


**Fig. 5.** Adaptively sampling solar source directions from a single point. Images are Shirley-Chiu squares showing a worm's eye view, where North is up and East is to the left. The detail is calculated from a combination of  $\bar{I}$  and GCR. Each sample represents the directional sampling for that source at a point, an example of which is shown bottom right. The bottom left image shows a complete solar sampling for this point overlaid over a sky sampling image to help locate the geometry.

trast based metrics, it is necessary to add the LightPoints together prior to metric calculation. This can be achieved by querying each LightPoint with the sampled ray direction vectors of the other, culling duplicates, and constructing a new LightPoint with the union of both vector sets. The Voronoi regions also need to be recalculated. Each sun LightPoint only holds a single source component, but a sky LightPoint may have hundreds of components. Assuming similar sampling rates between the suns and the sky, the combined LightPoints require 20–50 times the disk space. For disk storage and computation efficiency, it is practical to first group sky conditions by matching queries. All time-steps using the same pre-calculated solar positions are evaluated at once. This way the combined LightPoint only needs to be constructed once, used, and discarded without writing to disk.

### 2.3.2. Storing Light on a Plane

Using the same logic and implementation as the LightPoint, a similar method is needed to store a set of point sample results for a particular source on a plane representing possible viewing locations. The object that *raytraverse* uses to manage this is called a LightPlane. The representative area of each point is computed by intersecting the Voronoi regions with the boundary of the sampling area. Using these weights, zonal statistics can be calculated without interpolating to a regular grid. In the case of sky-patch coefficients, where one sampling scheme represents all of the sky-patch sources, this is all that is needed to organize the light-field data. However, the results for individual sun positions will not all match, because the sampling density will vary, as seen in Fig. 6.



**Fig. 6.** Positional sampling patterns for several different sun positions in single occupancy office facing East, the color scale shows the effective area of each sample. Direct sun penetration at eye-level is overlaid (bright vs. dark regions) to show the relationship between finer resolution sampling and abrupt changes in lighting condition.

Evaluating a point on the LightPlane at a particular time requires finding the closest sun position and point in space. Because the angular units of sun position do not match the units of spatial position it is not straightforward to build a kd-tree from the combined vector in a way that will minimize the combined error. Preliminary results have shown promise using two methods. A normalization coefficient can be determined from the resolutions of the sampling schemes for source and position, then the closest query can be used. Alternatively, in the case of a zonal evaluation, where there are not fixed points to query, the lookup can be done in two steps, beginning by filtering all points that have a source within a tolerance, and then culling points with greater source errors that have suitable alternatives within a positional tolerance. Future work will be needed to continue to develop and validate these methods.

Once a proper source and point combination has been selected and the various components of light sources added together, applying source coefficients and calculating view-based lighting quantities and metrics is straightforward and directly analogous to the *Radiance* executables *gendaymtx*, *dctimestep*, and *evalglare*. After collecting all the rays originating from a point within a view cone, the rays can be summed according to the metric. Given the low computational expense, at this stage the view to the direct sun is recalculated if the estimated position is above a threshold, to remedy high consequence errors, such as the sun being just behind (or just next to) a mullion. For the results presented in this paper, all suns were resampled when the source position error is greater than zero.

### 3. Validation Methods and Setup

Given the repeated validation of carefully implemented *Radiance* based workflows against measurement, there is support for using simulated reference data to assess the accuracy of alternative models and simulation methods that may offer ease of use or calculation speed improvements. As suggested by Reinhart and Andersen [19], if we are more interested in understanding the "intrinsic" than the "compound" error, a reference that controls for measurement and modeling errors outside the process being validated is potentially more informative. Where appropriate, a

simulated reference also can be used to validate a much wider range of conditions.

Building on a previous validation of *raytraverse* that assessed failure modes including specular reflections, complex fenestration systems, and rough specular reflections [1], this validation is focused on validating the efficiency of the process for a more typical case where both the simulation time and accuracy correspond to expected applied cases. Efficiency is defined in terms of the balance between accuracy and relative computation time. The goal of this validation is to isolate the additional reduction in simulation time and error introduced by *raytraverse*. To do this, the validation consists of a comparison between three methods: a reference simulation (for evaluating accuracy), a comparison simulation (for evaluating efficiency), and a complete workflow using *raytraverse*.

#### 3.1. Simulation Workflows

The three workflows use *Radiance*, as well as the same material and sky models. The reference simulation provides the best estimate of this simulated reality, avoiding any biased techniques typically used to speed up the calculation in favor of higher accuracy. To offer a more meaningful comparison for accuracy at more practical simulation times, the comparison and *raytraverse* workflows are both based on the 2-phase dynamic daylight simulation (DDS) method first proposed by Reinhart and Walkenhorst [18] and extended to a higher level of sun position discretization by Bourgeois [26]. This method was chosen over other CBDM methods based on a decision tree for selecting the appropriate method developed by Subramaniam [32]. The 2-phase DDS method is currently the only N-phase method that handles high scene complexity, provides accurately sized solar sources to calculate glare metrics, and produces an image output.

##### 3.1.1. Comparison simulation

The comparison simulation is an angular fisheye image based 2-phase DDS implementation, based on the example scripts included with Subramaniam's tutorial [32]. The 2-phase DDS method requires three simulation steps. First, the full ray depth (including direct and indirect contributions) daylight coefficients (DC) are simulated. Then, the direct component of the daylight coefficients

(DDC) is simulated. Finally, the direct component of a higher resolution grid of direct suns (SC) is simulated. To generate a time-step result, a sky vector (including direct sun) is applied to DC, the direct sun sky vector is applied to DDC, and the solar radiance is applied to SC. By subtracting DDC from DC and adding SC, the sky-patch direct sun contribution is replaced by the direct sun contribution. This yields a high accuracy result for direct view rays with an approximation of the indirect contribution of the direct sun. The referenced tutorial did not include an image-based example for the two-phase DDS, so the following was adapted from the 2-phase sensor example and 5-phase image example with *Bash* (linux/macOS) syntax for each view:

Because the solar sources are only used for a direct calculation, the cost of a finer resolution is generally small and can be partially offset by filtering the sources that are outside of the annual sun path. *Rcontrib* will still create placeholder images for these missing sources, but the simulation time is reduced. By adding a run-line encoding step, these placeholder images also take up very little disk space. For the highest accuracy run of the 2-phase, a sun discretization of 5185 bins is used. Filtering these values by the sky conditions used in this validation scope (see Section 3.2) means that only 686 source positions are actually simulated. This precise value varies depending on the number of time-steps and how the annual sun path for a particular site intersects with the Reinhart patch subdivision used.

---

```
vwrays -vf $vf -x $res -y $res -ff | rfluxmtx -w -ffc -v -n l2 -x $res -y $res \
$settings -o dout/images/"$bn"_MF1_%03d.hdr - $scene
vwrays -vf $vf -x $res -y $res -ff | rfluxmtx -w -ffc -v -n l2 -x $res -y $res \
$settings -ab l -o dout/images/"$bn"_MF1direct_%03d.hdr - $scene
# create filter for used suns bins (speeds up simulation time, but still
# creates images for all bins):
gendaymtx -5 0.533 -d -r 90 -m 6 dout/recs/lax.wea > dout/MF6_lax_suns.smx
echo "void light solar 0 0 3 le6 le6 le6" > dout/filtered_suns.rad
rmtxop -c l l l l -t dout/MF6_lax_suns.smx | getinfo - | total | rcollate -ir l -fal \
-ic 5186 -oc l -h | rcalc -e MF:6 -f reinsrc.cal -e 'Rbin = recno-l;cond=$l' \
-o 'solar source sun 0 0 4 $Dx $Dy $Dz 0.533' >> dout/filtered_suns.rad
vwrays -vf $vf -x $res -y $res -ff | rcontrib -w -ffc -n l2 -x $res -y $res -ld- \
$settings -ab 0 -o dout/images/sun_"$bn"/"$bn"_MF6sun_%04d.hdr \
-e MF:6 -f reinhart.cal -b rbin -bn Nrbins -m solar dout/filtered_suns.oct
# the following is run for each timestep/skyfile "sky.rad"\
genskyvec -m l sky.rad | dctimestep \
-oc dout/images/"$vf"_MF1_%03d.hdr > dc.hdr
genskyvec -m l -d sky.rad | dctimestep \
-oc dout/images/"$vf"_MF1direct_%03d.hdr > ddc.hdr
genskyvec -d -m 6 -5 sky.rad | dctimestep \
-oc dout/images/sun_"$vf"/"$vf"_MF6sun_%04d.hdr > sc.hdr
pcomb -o -s l dc.hdr -o -s -l ddc.hdr -o -s l sc.hdr > output.hdr
```

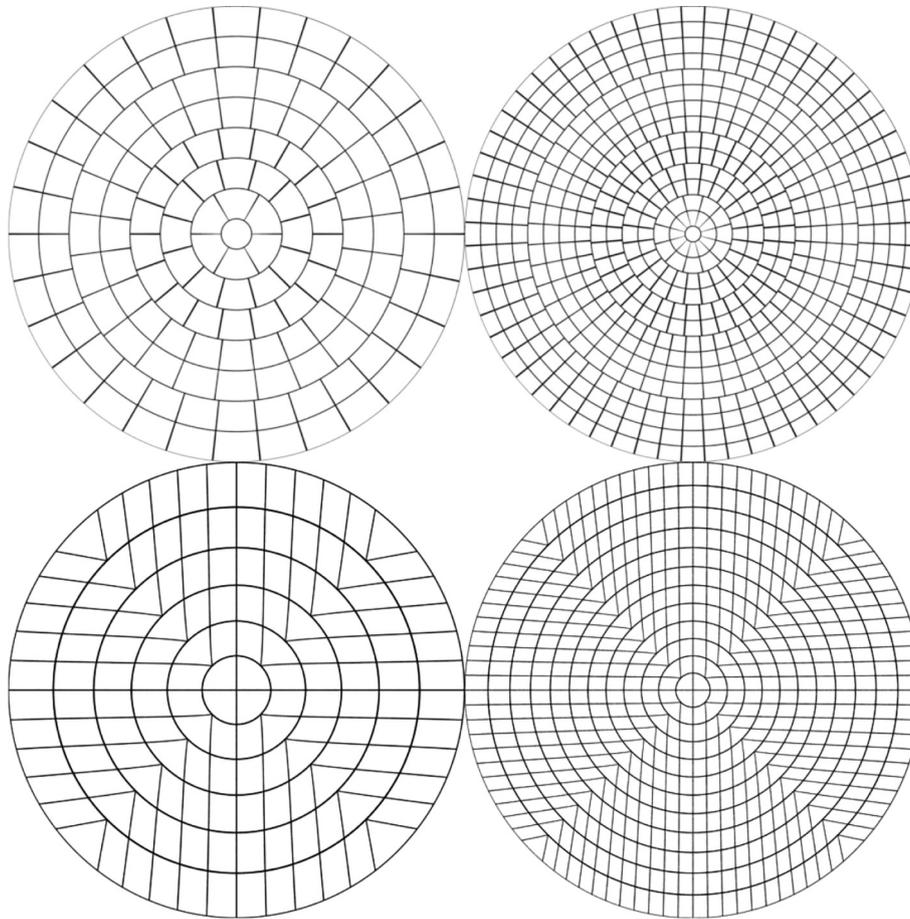
---

In addition to all of the *Radiance* rendering parameters, several additional parameters are required for a 2-phase workflow. These are the sky-patch discretization and solar source discretization. Sky-patch discretization presents an interesting tradeoff. A higher number of patches will better resolve the sky distribution, but the average number of rays terminating at each patch is inversely proportional to the patch area. In cases where adjacent patches differ, and without a complimentary increase in *-ad*, the relative gain in resolving the direct view of the sky can be undermined by the loss of precision in sampling. In general, this is not an issue for the smooth gradients of the Perez sky without direct sun. If a ray hits a nearby patch, it will likely be assigned a close value. If the patch resolution is increased, the difference between patches is reduced, but the likelihood of under-sampling a patch goes up, which mostly balances out. The problem arises when there is a large difference between sky vector values applied to patches, as is the case here when calculating the indirect component of the direct sun. Even for the highest accuracy run of the 2-phase, a Tregenza sky discretization of 145 patches is used because rendering parameters that did not introduce large systematic biases could not be found for certain conditions with higher Reinhart patch subdivisions [26]. The highest accuracy run produces 900x900 pixel resolution to match the reference simulation.

### 3.1.2. Raytraverse with 2-phase DDS

To accommodate the 2-phase DDS workflow, a process for calculating and subtracting the direct daylight coefficients is needed. To ensure the best possible match between the daylight coefficient rays and the direct daylight coefficient rays, the rays used to simulate the daylight coefficients are repeated, but with the parameters updated to stop at the direct contribution. Along with the results of sampling the direct sun contributions outlined in Fig. 5, we now have, for any given sky condition, three LightPoints, matching the DC, DDC, and SC images from the conventional 2-phase DDS method. These can be combined using the method described in Section 2.3.1.

There are two other distinctions from previously described implementations of the 2-phase DDS and that of the comparison simulation. First, the sky is discretized according to a digitized square projection of the sky hemisphere using the Shirley-Chiu disk to square transform instead of a Tregenza or Reinhart patch subdivision, see Fig. 7. This matches the discretization used by the directional sampling and the sun position sampling, and has several other advantages: patch areas are all equal size and discretization can happen at any square number. The second distinction is that only the enclosing sky-patch, instead of splitting energy to the nearest three, is used for approximating the direct sun.



**Fig. 7.** Comparing the Tregenza (top left, 145 patches) and Reinhart sky subdivision (top right, MF=2, 577 patches) to a square grid projected to the hemisphere with the Shirley-Chiu disk to square transform (bottom left, side=12, 144 patches; bottom right, side=24, 576 patches).

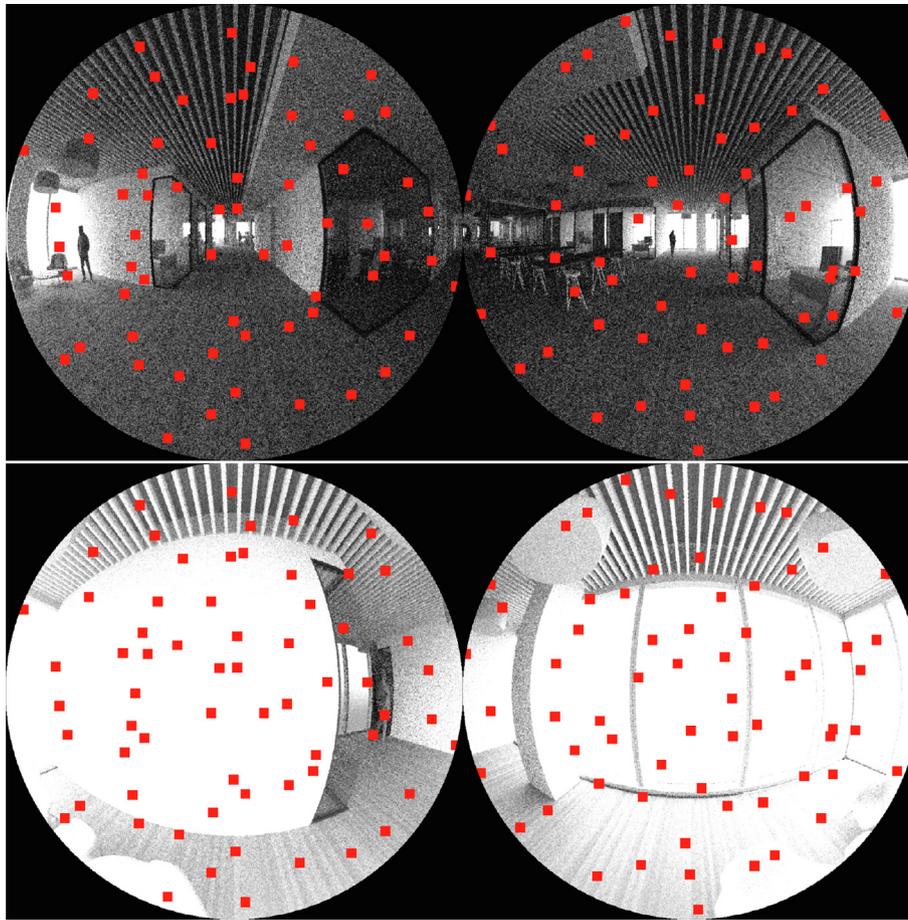
Testing showed that a coarser sky-patch discretization is more efficient, resulting in already large solid angles. Any additional grouping could introduce more directional and indirect source visibility errors than those resulting from off-centered distributions.

### 3.1.3. Reference simulation

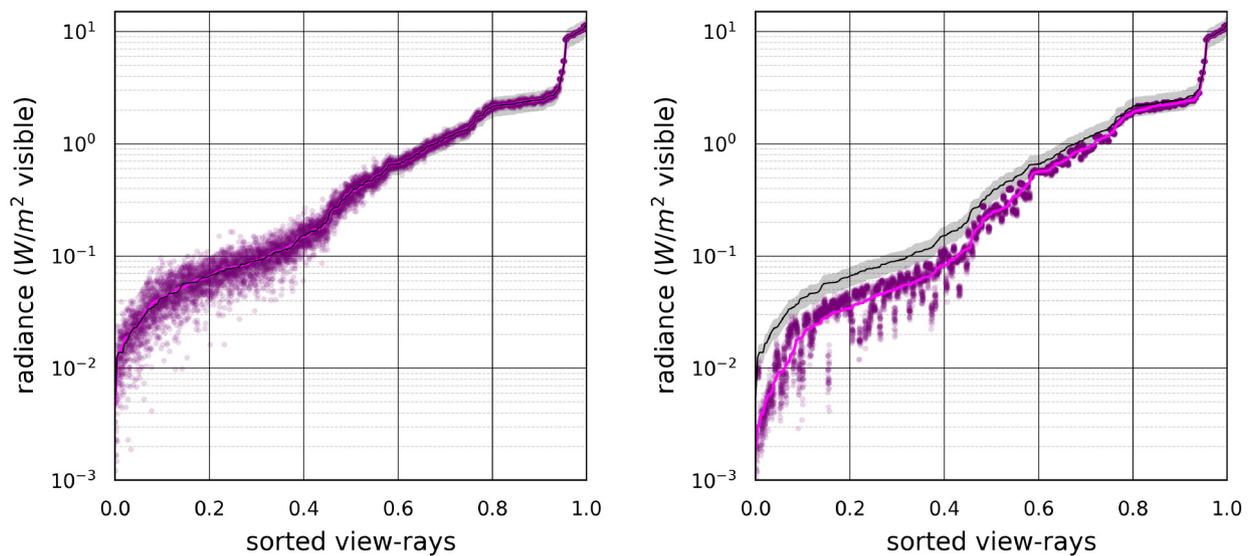
The reference simulation is not intended as a viable workflow, but a best estimate of ground-truth given the various models employed. By using a simulated reference, all of the models (materials, sky, weather) can be shared between workflows. This means that any deviation from reference is introduced by the tested workflow. The reference simulations are calculated using the *rpict* program of *Radiance*. Because of the size and complexity of the scene geometry, it was not practical to use ambient caching. Also, the ambient cache is not available in *rcontrib*. As a biased technique, the ambient cache could introduce an artificial difference in results between the reference and test workflows. Time and computer resource constraints required a relaxation of rendering parameters. Given that generating the reference data requires rendering 34,352 images, to select stable and unbiased rendering parameters that are also calculable with the available computing power, a manual parameter relaxation was performed:

1. A random set of 256 view rays are drawn from a dark and bright region of the scene (see Fig. 8) with an overcast sky (to isolate the indirect calculation).
2. To establish a benchmark, each view ray is simulated 32 times with `rtrace -ad 4000 -lr -14 -lw 1e-5` and averaged (roughly equivalent to `rtrace -ad 128000 -lr -14 -lw 3.125e-7`).

3. The `-lr` and `-ad -lw` parameters are independently relaxed, repeating the the same simulation process, but this time using the 32 repetitions to estimate the variance. While it is more common to vary the `-ab` parameter, `-lr` functions similarly for ambient rays.
4. These values are plotted against the reference values, sorted by intensity for readability, and the simulation time was recorded. Fig. 9 shows examples for `-lr -2` and `-ad 250`.
5. There was no significant time savings through reducing `-lr` until `-4`, but these settings have a negative bias of 10% (see Fig. 9, right). On the other hand, even at `-ad 250` there is no observed bias, just a large variance. `-lr -14 -ad 250` provides tremendous savings in simulation time (12 times faster than `-lr -2 -ad 4000` and 14 times faster than `-lr -14 -ad 4000`). Despite the large variance at lower intensities, this was determined to be acceptable, since the endpoint metrics are calculated for full images, where local noise, especially at low intensities will cancel out. This was also a practical decision because it was estimated that running on the 60-core cluster available, it would still take 2 months to calculate to full dataset, even with these relaxed settings.
6. The final rendering parameters used are: `rpict -u+ -ab 14 -lr -14 -aa 0 -av 0 0 0 -ss 1 -st 0.001 -dc 1 -dt 0 -x 900 -y 900 -ad 250 -as 0 -lw 1.58e-3 -ps 3 -pt.04 -dj 1`. The `-ps 3` parameter was used to reduce simulation time while still maintaining high resolution at sharp edges. It was discovered after simulating the entire set that this sampling failed to catch all of the specular reflections of the direct sun for some of the views. To remedy this, but also avoid an additional 3 months of simulation time and



**Fig. 8.** A wide range of representative samples are used to determine appropriate parameters for the reference simulation. Background images are low quality estimations with the same exposure to highlight both the range of values and varying rates of convergence (more noise in darker regions).



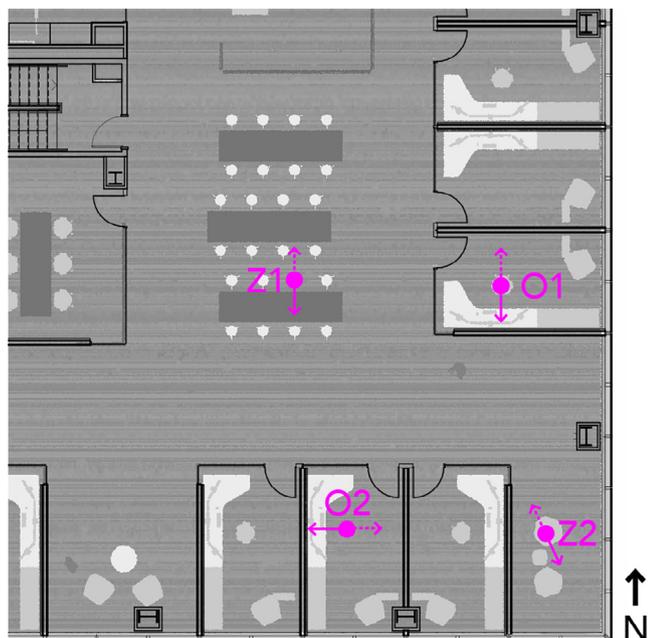
**Fig. 9.** Results from parameter check simulations. Left: `-ad 250 -lr -14 -lw 1.6e-4` Right: `-ad 4000 -lr -2 -lw 1e-5`. Black line shows reference values with 20% error band. Magenta line shows mean result for each view-ray (resorted) to show distributional bias. Dots show individual samples. Note that the left plot has increasing normally distributed noise at low intensity but no bias. The right plot shows increasing negative bias at low intensity. The noticeable vertical clumps of points reveal that the bias varies by sample direction. Those with larger negative bias have proportionally larger contributions from higher order bounces.

the energy use this implies, all views with sun positions within 3 degrees of a possible specular reflection were rerun with  $-ps\ 1$ . For an explanation of how the views and skies were filtered, see [Appendix B in the supplemental materials](#).

7. The original  $-ps\ 3$  renderings took 55 days. As a final check, 200 randomly selected sky conditions were re-rendered with a slightly different resolution (903x903) to avoid potential aliasing effects. The mean absolute error (MAE) of the Unified Glare Probability (UGP) and Daylight Glare Probability (DGP) is less than 0.001 and the mean absolute percentage error (MAPE) of  $E_v$  is less than 0.2%. This confirms that the increased noise in single pixel results does not propagate to whole image evaluations.
8. For the set of 398 views and times with possible reflections, 203 have either no sun or a full view of the sun and 195 have some combination of a reflection and/or a partial view to the direct sun. For the first set, between the  $-ps\ 1$  and  $-ps\ 3$ , the MAE of UGP and DGP is less than 0.001, and the MAPE of  $E_v$  is less than 0.3%. This confirms that for conditions without partial views to the direct sun or reflections, the  $-ps\ 3$  parameter is sufficient. However, the re-run of the reference data only covers reflections. Among the eight cases of partially visible sun without reflections, the two runs had a MAPE for  $E_v$  of 14.0%. While this is a limited and possibly non-representative set, it does imply that the rest of the reference data could have a significant error for these conditions, of which there are only 128 (0.4%) such cases in the entire reference set of 34,352.

### 3.2. Validation Scope

The scene is an office building, currently under construction, located in Irvine, CA. Materials and geometry are taken as is from the provided model, which was used in consulting on daylight and electric lighting design for the real project (see acknowledgements). Materials do not include any complex fenestration systems or rough specular surfaces. Specular surfaces are limited, but include interior glazing, exterior glazing, and computer monitors.



**Fig. 10.** Partial floor plan. Solid arrows indicate primary view direction, dashed arrow is the reverse view (denoted with an R when referenced). Metrics are calculated for all eight view-point/directions, but all simulations support calculating metrics for any view direction from one of the four points.

Four view points in the southeastern corner of the second floor cover a wide range of conditions: an east facing private office, a south facing private office, an interior open space, and a corner lounge area, see [Fig. 10](#). For each point, two 180 degree angular fisheye views are simulated, a primary direction and the reverse view. While the validation uses just the two view directions at each point, it is possible to calculate view-based metrics for any view direction from a pair of fisheyes. While not an important factor for measuring accuracy, this is useful for measuring efficiency between image-based methods and *raytraverse*.

The facade is clear glazing (64% visible light transmittance) and the complete analysis is performed without dynamic shading. While less indicative of a building in real operation, this is a commonly simulated scenario to understand how the base building is performing. Additionally, having small transient patches of direct sun visible from deep within the space creates a particularly challenging scenario for adaptive sampling because there is greater variance between sun positions and the indirect distribution of light in the space. Any sun control, such as roller shades, that reduces this variance and contribution will be relatively easier for the adaptive source sampler. The sky conditions include all daylight hours in the TMY3 dataset from Los Angeles International Airport (downloaded from: [climate.onebuilding.org](http://climate.onebuilding.org)). Daylight hours were defined as times with the sun at least two degrees above the horizon and a diffuse horizontal irradiance of at least  $5\ W/m^2$ , resulting in 4,294 sky conditions. Skies were modeled using the Perez all-weather sky model [33] with solar angle, direct normal irradiance, diffuse horizontal irradiance, and dew-point temperature.

For each individual view and time, accuracy is evaluated for three values,  $E_v$ , UGP, and DGP, which offer a range of saturation, contrast, and hybrid glare metrics. While other glare and visual comfort metrics may evaluate differently, a workflow that can accurately measure these three metrics will likely be accurate for any metric composed of similar values. Metric values are compared directly, on a per time-step basis and not used as thresholded glare predictions. This provides a broad range of applicability because any derived, summary, or binary metrics and statistics will be at least as accurate. This also insulates this analysis against a lack of generality that could otherwise be introduced by changing glare threshold values or intensity changes in the lighting condition, such as a different glazing transmission or sky irradiance, that do not otherwise impact the accuracy of the workflow.

In the course of the data analysis, large discrepancies due to glare source grouping were observed. This confirmed an observation made by Pierson et al. [23] that for open office environments the radius parameter used for grouping sources may need to be carefully tuned. For a large scale simulation with widely varying conditions, this is simply not practical. To minimize discrepancies introduced by post simulation analysis, this study evaluated both the reference and comparison workflows without glare source grouping.

### 3.3. Data normalization/ error analysis

Among the validations reviewed in Section 1.2, the most common error metrics reported were MSD, quantifying systematic over or under prediction, and RMSE, quantifying the unsigned error. The advantage of these statistics is they are now well established in the literature. However, MAE is used because it can be interpreted more intuitively than RMSE. MAE is the linear average error, whereas RMSE is quadratically weighted and represents the standard deviation, or square-root of the variance, which is not a meaningful value for non-normally distributed residuals. This means MAE also has the same scale as MSD, so the ratio of bias to noise

can be inferred and they can both be normalized as percentage errors in the same way.

A disadvantage of a mean quantity is that it provides no information about the population distribution. The types and magnitudes of simulation errors vary greatly by condition, so including a disproportionate number of error-prone or error-free scenarios skews the result. Imagine a method that predicts a quantity in direct sun well, but in indirect lighting poorly. The same analysis done for a North facing facade in the northern hemisphere will show high error, but moving the scene to the southern hemisphere will eliminate the error. This type of distribution bias is present in a number of studies that erroneously claim illuminance can be used as a general replacement for image-based glare metrics, see Wasilewski et al. [34] for examples. Another issue, for metrics like  $E_v$  with magnitude spanning multiple orders, is that the magnitude of the error may not be independent from the magnitude of the metric. An absolute or square error will mostly reflect high value errors, but if a percentage error is calculated, then low value errors will dominate. Both of these issues can be mitigated by splitting the error quantification, because at least then the error-prone regions can be identified. A number of the validations reviewed in Section 1.2 do this by categorizing the conditions in some way, including by sky condition, sun visibility, measurement point, and transmitting material.

The approach taken here is to bin the data by each metric's reference value and then calculate the error for each of these bins. Assuming enough samples exist in each bin to calculate a reliable estimate, then the result is an error profile that does not depend on the distribution of conditions in the reference data set. The binned errors are used to calculate a normalized mean. Taking a mean of the bin means yields an average that is equivalent to a weighted average, where each sample is weighted by the reciprocal of the number of samples in its bin. UGP and DGP errors are quantified by MSD and MAE. Both are grouped into 8 bins split at (0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8), such that the global mean is uniformly weighted from 0.2–0.8 and the low and high regions are weighted by half.  $E_v$  error is quantified by MSPD and MAPE.  $E_v$  values are grouped into eight bins split at (75, 150, 300, 600, 1200, 2400, 4800) such that the global mean error is weighted uniformly by the  $\log(E_v)$  from 75 lux to 4800 lux. Mathematically, the distribution normalized error can be described as:

$$\text{error}_{dn} = \sum_{j=1}^{nb} \frac{\sum_{i=1}^{n_j} \Delta x_i}{n_j nb}, \quad (2)$$

where  $nb$  is the number of bins,  $n_j$  is the number of cases in bin  $j$ ,  $i$  is the case,  $\Delta x_i$  is the deviation of the value at case  $i$ . For MAE<sub>dn</sub>,  $\Delta x_i = |x_{i,test} - x_{i,ref}|$  and for MSD<sub>dn</sub>,  $\Delta x_i = x_{i,test} - x_{i,ref}$ .

### 3.4. Efficiency analysis

Simulation time is heavily dependent on parameter selection, and a single parameter selection, which, even when carefully considered, may not yield the most efficient process. The parameters for both *raytraverse* and 2-phase were initially chosen to prioritize accuracy. To complete an efficiency analysis, both workflows were repeated with a steady degradation of parameters to generate an efficiency curve, plotting time vs. error. Some parameter sets that are not the fastest for a given accuracy level (what, in an optimization problem, would be called Pareto dominated) were tested, but discarded. For the 2-phase workflow, it turns out to be much more efficient to reduce the image resolution compared to any rendering parameters because this also speeds up evaluation. Table 1 gives the full range of included parameter sets. Options are sorted by

simulation time and labeled beginning with "A", from slowest to fastest. Corresponding letters between *raytraverse* and 2-phase do not correspond to similar simulation times.

## 4. Results

The reference, 2-phase, and *raytraverse* workflows are each evaluated for three metrics (UGP, DGP, and  $E_v$ ), four points, two view directions, and 4,294 sky conditions. Fig. 11 shows individual results, grouped by view, comparing the reference simulation to 2-phase and *raytraverse* for UGP and DGP. Additional per view scatter-plots can be found in appendix A of the supplemental materials. Similarly,  $E_v$  is shown only in the supplementary materials because its pattern of errors is quite similar to DGP, which because of the high glazing transmission, is dominated by saturation effects in a large majority of the included cases. For each method, three parameter sets are shown to illustrate the magnitude and nature of errors introduced by the parameter relaxation. For the *raytraverse* method, the overall shape of the scatter plots remains consistent, except with a wider spread and the noticeable cluster of low values for view Z2, representing missed reflections off of the interior of the glazing. The 2-phase parameter relaxation shows the emergence of clear systematic errors for both UGP and DGP, most visible for views O1 and Z2, where the direct sun and specular reflections are frequently visible.

In order to quantify what the scatter plots show and to insulate the reported error from the frequency of tested conditions, the data is binned and normalized according to the procedure described in Section 3.3. Fig. 12 shows the distribution, based on the reference results, of the test conditions for each metric. The distribution of DGP values in particular show why normalization is needed to get a useful overall error quantification. A third of all conditions have a DGP value below 0.2, which is outside the validated range of the metric and well below any standard glare detection thresholds. Certain combinations of view and metric values have few cases, but across all the views, the tested scenario has a minimum of 591 cases for each bin with contributing cases from at least five different views.

Table 2 shows the MAE and MSD for the same three parameter sets from each workflow shown in Fig. 11. The parameter sets include the highest quality parameters, then a middle run and fast run for each workflow. At all three levels the *raytraverse* runs are four to six times faster. These tables quantify what the scatter plots illustrate: *raytraverse* remains generally reliable throughout the parameter relaxation with a small increase in error across all metric magnitudes. There is a tendency towards a positive bias for illuminance values less than 600 lux and towards a negative bias for UGP and DGP. The 2-phase method is susceptible to negative bias across all metrics for the fastest parameters and is error prone for high magnitude values for all metrics. By averaging the error metrics calculated for each bin, a normalized error gives a summary metric describing the overall accuracy of the method/parameters independent from the frequency of conditions.

The reason for relaxing the parameter at the cost of accuracy is to reduce the simulation time. For an image (or LightPoint) based daylight coefficient process the total simulation is a combination of first: rendering the sky, direct sky, and sun contributions, and second: applying the sky matrix, combining the components, and evaluating the result for the metrics. The timing of both parts is dependent on the number of samples, hardware and software versions used, processor load, and executive scripts used to run the process. Only the rendering steps are dependent on the model size, scene complexity, and rendering parameters, except where the scene complexity requires a higher resolution of output samples (either automatically, in the case of *raytraverse*, or manually

**Table 1**

Table of tested *raytraverse* and 2-phase parameter sets. Ranges give initial and stopping resolutions, the grid side is the number of samples per square side of the sampling space, so 64 would be a 64x128 resolution to sample a sphere. src. res. is the resolution for solar source direction sampling. sky and sun ray res. is the resolution for directional sampling with each of those sources. *-ad* varies by parameter set and the *-lw* parameter is set to 0.4/ad. The shared *Radiance* parameters for both methods are: for the sky: *-u+ -ab 14 -av 0 0 0 -aa 0 -as 0 -dc 1 -dt 0 -lr -14 -ad -st 0 -ss 1* (with *-ab 1* for the DDC component) and for the sun: *-u+ -ab 0 -av 0 0 0 -dc 1 -dt 0 -st 0 -ss 1*.

label	accuracy (threshold)	sky-patches	src. res. (degrees)	ad	sky ray res. (grid side)	sun ray res. (grid side)
raytA	0.5	1296	10–1.25	130,000	64–1024	64–1024
raytB	1.0	324	15–1.88	65,000	64–512	64–1024
raytC	1.0	324	20–2.5	32,500	64–512	64–1024
raytD	1.0	144	20–2.5	14,500	64–512	64–1024
raytE	1.0	144	20–2.5	14,500	32–512	64–1024
raytF	1.0	144	20–5	7,250	32–512	64–1024
raytG	2.0	144	20–5	7,250	32–512	64–1024
raytH	3.0	144	20–5	5,000	32–256	64–512
		sky-patches	sun patches	ad	image res.	
2phsA		145	5186	8000	900	
2phsB		145	5186	1000	400	
2phsC		145	5186	1000	300	
2phsD		145	2306	1000	300	
2phsE		145	5186	1000	250	
2phsF		145	5186	1000	200	
2phsG		145	2306	1000	200	
2phsH		145	5186	1000	150	
2phsI		145	5186	1000	100	
2phsJ		145	2306	1000	100	
2phsK		145	5186	1000	50	
2phsL		145	2306	1000	50	

through knowing a higher resolution image is needed). **Table 3** gives the rendering and processing times for both methods in real clock time per view location (2 views), as each location in this case is an independent simulation. Both methods were run on the same computer: a 2018 MacBook Pro with Intel 2.9 GHz Core i9 processor (6 cores w/ hyperthreading), 16 GB RAM, and a solid state hard drive. Image based simulations were run using *Radiance* 5.4a (2021-02-07). *Raytraverse* simulations used *Radiance* 5.4a (2021-08-29)<sup>5</sup>. This is 2.1 times slower than the fastest entry on the benchmark page. Both processes were parallelized on 12 threads for rendering, combining, and evaluation, but not sky matrix generation. Wherever possible this was done using the *-n 12* flag of *rcontrib*, otherwise it was done using a *ProcessPoolExecutor* in Python 3.7. For the 2-phase image method, the Python code is just to start parallel os system calls using the *subprocess* module and does not introduce the overhead of running computations in Python. *Raytraverse* is written mostly in Python, so some calculation steps that do not use numpy or other C/C++ bindings, suffer the overhead of the uncompiled language.

The reference simulations were produced only as a benchmark for accuracy, so comparable computation time was not controlled for. The reference simulations were run on a remote cluster installed with *Radiance* 5.3a (compiled 2020-08-21), and evaluated locally. Based on running one image for each view locally with the reference settings, the timing to generate an image per processor was 10% faster on the server. As a rough approximation, the reference simulation and evaluation took 16 times longer than the full resolution 2-phase (2phsA) simulation and evaluation, adjusting for number of processors and single thread time for one image.

The summary normalized  $MAE_{dn}$  and  $MSD_{dn}$  are plotted against simulation time in **Fig. 13**. Both methods are able to reproduce the reference results with minimal bias and error using high quality parameters. The 2-phase method can maintain this accuracy

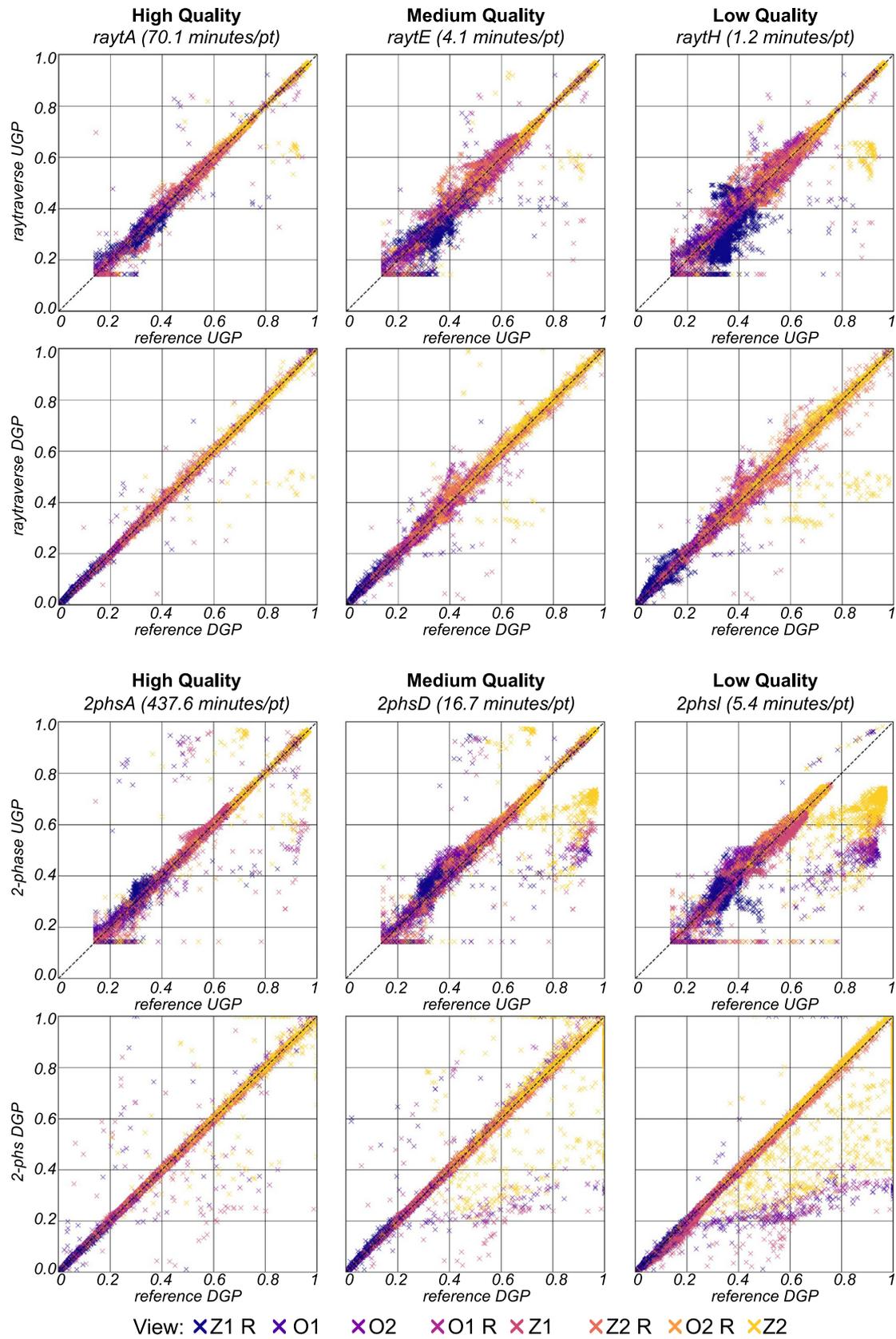
with a 14 times speedup by lowering the image resolution, but faster parameter sets introduce increasingly significant errors across all metrics with significant negative bias for UGP and DGP. *Raytraverse* avoids any bias with magnitude greater than  $MSD_{dn}$  0.02, and the introduced  $MAE_{dn}$  rises much more slowly than for 2-phase. For the fastest *raytraverse* parameters, similar accuracy can be achieved 17 times faster than 2-phase for UGP and DGP and eight times faster for  $E_{\nu}$ .

## 5. Discussion

This study assesses each workflow run with a number of parameter sets. Given the number of parameters exposed by *Radiance*, it is important not to think of any workflow as having a set simulation time or accuracy, but rather offering a trade-off between time and accuracy given a certain simulation scope. The reliability of the method is assessed by how well it maintains accuracy as parameters are deteriorated.

The 2-phase DDS method was chosen as the comparison test method for this study because it could reliably reproduce the results from the reference simulation with proper parameter selection. The problem is that this takes a very long time (7 h 20 min/point with the hardware previously described) to simulate. The parameter relaxation revealed that there are limits to how much the key parameters driving simulation time (*-ad* and image resolution) can be reduced before systematic errors are introduced. For the normalized distribution of conditions used in **Fig. 13**, this becomes noticeable for UGP and DGP at around 15 min. However, examining the 2phsD UGP scatter plots in **Fig. 11** shows a region of systematic under-prediction for two of the views. It is likely that for scopes with a large number of hours where a specular view of the sun is present, the 2phsB parameters (400x400 pixels) would yield a noticeable bias leading to under-prediction of both contrast and saturation. In scopes without any direct sun and generally diffuse conditions, the 2-phase DDS method is likely reliable at even the lowest parameter sets tested here, but in that case little image information is gained compared to a sensor point calculation which would offer further time savings.

<sup>5</sup> Using a previous 5.4a release, this hardware achieved a time of 174.61 s using 12 threads on the *Radiance SMP Benchmark Results* ([http://markjstock.org/pages/rad\\_bench.html](http://markjstock.org/pages/rad_bench.html), accessed: Feb. 23 2022)



**Fig. 11.** UGP and DGP scatter plots for high (raytA, 2phsA), medium (raytE, 2phsD) and low (raytH, 2phsL) quality parameters (see Table 1). Views are described in Fig. 10. Note that mathematically UGP cannot be less than 0.1435.

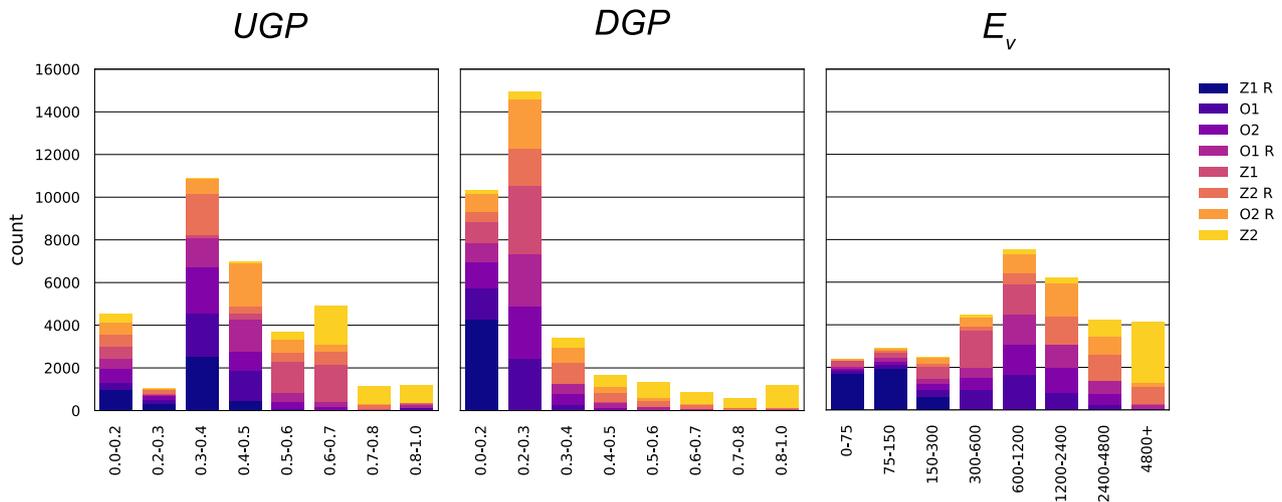


Fig. 12. cases per view and metric range based on reference simulation values.

Table 2

Error tables. As the average of all the rows the  $error_{dn}$  is the distribution normalized error defined in Eq. 2. Tables with error by view and magnitude can be found in Appendix A of the care defined in Table 1.

	MAE						MSD							
	raytA	raytE	raytH	2phsA	2phsD	2phsL	raytA	raytE	raytH	2phsA	2phsD	2phsL		
UGP	0.0-0.2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00		
	0.2-0.3	0.01	0.03	0.04	0.03	0.03	0.04	-0.01	-0.01	-0.02	0.00	-0.01	-0.02	
	0.3-0.4	0.01	0.01	0.03	0.01	0.01	0.01	0.00	0.00	-0.01	0.00	0.01	0.00	
	0.4-0.5	0.00	0.01	0.02	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.01	0.00	
	0.5-0.6	0.01	0.02	0.02	0.01	0.01	0.02	0.00	0.00	0.00	0.00	0.01	0.00	
	0.6-0.7	0.00	0.01	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	-0.01	
	0.7-0.8	0.00	0.01	0.01	0.01	0.01	0.02	0.03	0.00	0.00	-0.01	0.00	0.00	-0.03
	0.8-1.0	0.01	0.01	0.03	0.02	0.10	0.29	-0.01	-0.01	-0.02	-0.02	-0.10	-0.29	
$error_{dn}$	0.01	0.01	0.02	0.01	0.02	0.05	0.00	0.00	-0.01	0.00	-0.01	-0.04		
DGP	0.0-0.2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00		
	0.2-0.3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00		
	0.3-0.4	0.00	0.01	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00		
	0.4-0.5	0.00	0.01	0.02	0.01	0.01	0.02	0.00	0.00	0.00	0.00	0.00	-0.01	
	0.5-0.6	0.00	0.01	0.02	0.01	0.02	0.05	0.00	0.00	0.00	0.00	-0.01	-0.03	
	0.6-0.7	0.01	0.02	0.03	0.01	0.03	0.06	0.00	0.00	-0.01	-0.01	-0.02	-0.05	
	0.7-0.8	0.00	0.01	0.02	0.02	0.04	0.07	0.00	0.00	-0.01	0.00	-0.01	-0.05	
	0.8-1.0	0.01	0.00	0.02	0.01	0.05	0.17	0.00	0.00	-0.01	0.00	-0.04	-0.16	
$error_{dn}$	0.00	0.01	0.01	0.01	0.02	0.05	0.00	0.00	0.00	0.00	-0.01	-0.04		
$E_v$	0-75	1.6%	3.1%	3.1%	1.2%	2.7%	5.7%	0.9%	1.6%	0.5%	-0.1%	1.5%	-5.5%	
	75-150	1.8%	4.2%	4.2%	1.4%	1.3%	5.7%	1.1%	2.9%	2.4%	0.3%	0.2%	-5.0%	
	150-300	1.4%	3.1%	5.4%	2.1%	2.3%	5.9%	0.1%	0.6%	2.3%	0.2%	1.0%	-5.0%	
	300-600	1.0%	2.9%	3.1%	2.5%	1.8%	6.1%	-0.3%	-0.6%	-0.3%	1.0%	-0.2%	-5.6%	
	600-1200	1.2%	3.1%	3.7%	1.6%	1.8%	4.1%	-0.2%	-0.1%	0.5%	0.2%	0.1%	-3.1%	
	1200-2400	1.7%	4.7%	5.9%	2.0%	2.0%	2.9%	0.2%	0.7%	1.4%	0.4%	0.2%	-0.8%	
	2400-4800	1.9%	5.3%	6.3%	1.7%	2.5%	4.6%	0.3%	0.7%	0.3%	-0.5%	-0.4%	-0.1%	
	4800+	1.3%	2.3%	3.5%	3.2%	8.9%	13.6%	0.3%	0.5%	0.0%	0.6%	2.5%	-5.0%	
$error_{dn}$	1.5%	3.6%	4.4%	2.0%	2.9%	6.1%	0.3%	0.8%	0.9%	0.3%	0.6%	-3.8%		

The *raytraverse* coupled 2-phase workflow appears to be reliable for calculating UGP and DGP even at the fastest parameter sets tested, yielding distribution normalized low-error unbiased results for the complete set of conditions in around 1.2 min/point. However, two issues suggest interpreting this result cautiously. First, the scatter plot for raytH UGP in Fig. 11 shows a systematic under prediction for cases with reflections off of the interior of windows (view Z2). Second, most obvious for view Z1 R (the most interior view), the scatter plots have noticeable clumps of conditions. These are due to systematic under or over predictions driven by the sky

coefficient results. Overall, it appears that the adaptive sampling approach used by *raytraverse* can reproduce the results of high-resolution grid based approaches to calculate contrast and saturation based metrics. Coupled with the means for storing and evaluating non-uniform sampling data, described in Section 2.3, it enables the accurate evaluation of large spatial-temporal domains with a high effective resolution.

Both of the tested workflows, *raytraverse* and 2-phase, are built on top of the same models and parameters as the reference simulation workflow. While this setup introduces some important lim-

**Table 3**

Simulation processes in minutes/point. Times reported separately for rendering and evaluation. Speed-up is relative to full resolution 2-phase simulation (time 2phsA/time), which is itself approximately 16.3 times faster than the reference. Parameter sets are defined in Table 1.

raytraverse	Speed-up	Total	Rendering	Evaluating
raytA	6	70.1	63.5	*6.6
raytB	22	19.5	17.6	1.9
raytC	40	10.9	9.2	1.7
raytD	72	6.1	4.7	1.4
raytE	106	4.1	2.9	1.2
raytF	152	2.9	1.8	1.1
raytG	275	1.6	1.0	0.6
raytH	378	1.2	0.7	0.5
<b>2-phase</b>				
2phsA	1	437.6	352.0	85.7
2phsB	14	32.3	16.3	16.1
2phsC	22	19.9	9.6	10.4
2phsD	26	16.7	10.0	6.7
2phsE	27	16.0	9.7	6.3
2phsF	35	12.7	8.7	4.0
2phsG	44	9.9	6.9	3.0
2phsH	44	10.1	7.8	2.3
2phsI	52	8.4	7.3	1.0
2phsJ	70	6.2	5.5	0.8
2phsK	70	6.2	5.9	0.3
2phsL	81	5.4	5.2	0.2

\* Evaluating raytA was limited to six threads due to memory constraints.

itations on what can be concluded, it also introduces opportunities to attribute the source of errors and therefore the broader applicability of the validation results.

5.1. Error attribution

For high quality settings across all bins of illuminance values, this validation found average errors of less than or equal to 2.0% for raytraverse and 3.2% for 2-phase. This error controls for light measurement errors as well as material, geometry, sky and observer models as well as parameter optimization. The remaining error is attributable to the differences in implementation. For the 2-phase DDS method, this includes the discretization of the sky into patches, a fixed grid of sun positions, and a sky patch to estimate the indirect contribution from the direct sun. In addition, raytraverse differs from the reference by adaptively sampling view rays and sun positions.

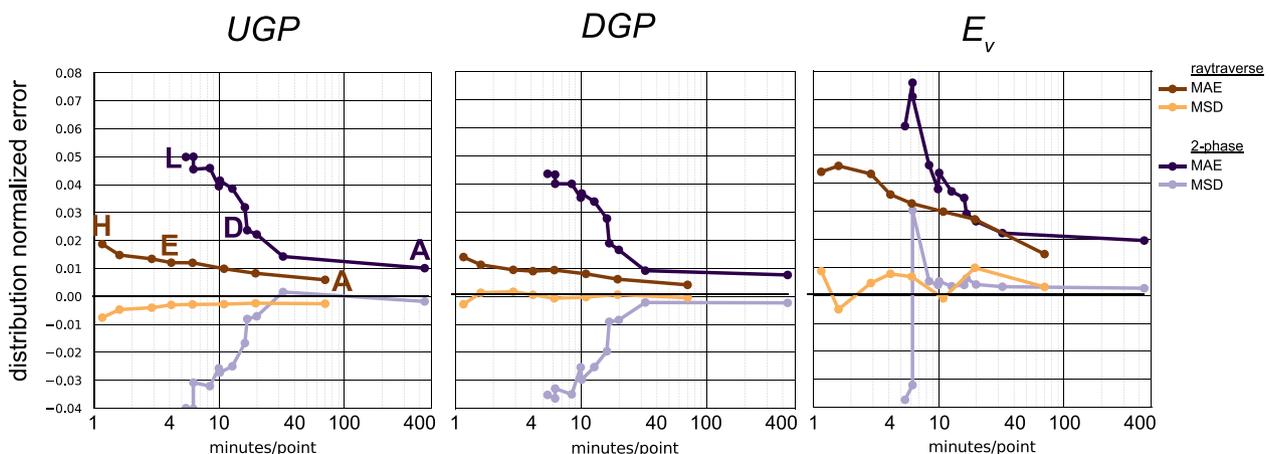
The errors found here are an order of magnitude smaller than the 10–20% relative errors found in physical measurement validations of similar CBDM methods [25,17,18,26,19,20] implying that most of the error found by those studies is introduced upstream from this validation, for example in the models and measurement. This isolation of the errors introduced by the 2-phase and raytraverse methods allows for pinpointing possible sources of these errors.

Within the scope of this validation, the two most difficult conditions to accurately simulate with shorter simulation times are specular reflections and deep interior views. For deep interior views, the errors are most apparent for  $E_v$ . The issues stem from under-sampling, both failing to resolve apparently small windows or patches of sun and finding sufficient indirect ray paths. At a certain point, resolving these areas in higher detail simply requires more sampling, but the lower error and bias observed for raytraverse compared to the fixed resolution 2-phase shows that even in these conditions, the adaptive sampling improves results.

Regarding reflections, the 2-phase method is only reliable for the highest quality settings. This is because, in order to ensure that a pixel sample will strike a reflection of the sun, which has a diameter of 0.533 degrees, the width of a pixel must be less than the diameter divided by  $\sqrt{2}$  (0.376 degrees). For an 180 degree angular fisheye, pixels begin to exceed this threshold when the image is smaller than 500x500 pixels. Raytraverse partially avoids this by sampling for potential reflections at a resolution of 512x512 samples per hemisphere, but only when the sky sampling provides a reliable guide of potential reflections. At view Z2, this proved difficult because the dominant contribution in the reflection direction is not the reflection, but the view to the sky beyond. At fast settings, this area is not reliably sampled. Because these conditions were run for the reference set with both -ps 1 and -ps 3, there is a useful benchmark to contextualize this error. For these 98 conditions, -ps 3 introduced a -0.05 MSD in UGP. A similar level of bias is maintained for 2phsA-C and raytA-C. Future work on raytraverse will look to explore additional techniques for marking potential reflections.

5.2. Limitations

Among the most important limitations, and true of all simulated reference validations, the observed deviation from the reference must be assumed to be additive with whatever error the reference has to the true conditions. The true error, especially in



**Fig. 13.** Efficiency curves plotting time (real minutes per point) vs. error for each metric. The left plot (UGP) labels the three parameter sets for each workflow included in other detailed figures and error tables. Parameter sets are defined in Table 1.

this case where the workflows use the statistically derived TMY data and the statistically modeled Perez sky model, cannot be known. However, this is an important use case as it is the most commonly used source sky data for CBDM. Results from TMY data do not match any observable reality, but they do provide a set of conditions that are likely sensitive to changes in the same way as reality. For this reason, simulation results are better interpreted as relative values (between options, sites, buildings) rather than absolute values [35].

Another limitation is that where a scene or observer model does not behave similarly the validation is not valid. For example, continuous and gradually changing distributions (such as the Perez model) may be accurately represented with a small number of sky patches and coarser indirect sampling (the *-ad* parameter in *Radiance*). These same settings would not be valid for discretizing a sky with more abrupt changes, such as an HDRi sky including bright clouds or a complex horizon line.

Because currently accepted daylight glare metrics do not depend on light spectrum, *raytraverse* stores all results in a single luminance channel. Recent research suggests that color does play a role in discomfort glare [36], and ongoing research under daylight conditions also suggests that color is a factor [37]. Depending on how this is accounted for in future daylight glare metrics, *raytraverse* can be extended to either maintain color information, or in the case that spectral rendering is needed and has a significant variation across a scene could even be adaptively sampled using the same wavelet based methods.

As covered in Section 1.1, efficient methods for calculating specific glare metrics exist that bypass finding a full solution to the daylight incident on a view. These methods can, for some cases, including most of the conditions in the case-study, generate accurate results in less time than the comparison simulation. On the other hand, those methods provide a more limited set of information. As *raytraverse* is intended as a more general purpose sampler, this study does not cover whether it would offer any benefit to workflows that directly target the inputs to specific glare metric equations, like eDGPs, *ClimateStudio* annual glare, and *dcglare*.

## 6. Conclusion

This paper presents a new simulation method, called *raytraverse*, for sampling daylight conditions in architectural spaces. Like other daylight coefficient approaches, the time dimension is treated as a collection of source directions, using a precomputed sky matrix to recover the time-steps. In addition to explaining how *raytraverse* works to adaptively sample the directional, source, and positional dimensions of an analysis scope, a simulated reference validation demonstrates the efficiency of these methods compared to a uniform sampling CBDM approach for the directional and source sampling. It remains for future work to validate the positional sampling, which will require high-resolution reference data in both positional and source direction domains.

To increase the robustness of this validation, which is done for a single real world case-study of a typically complex design model used in practice, new strategies are proposed. The reported error metrics are insulated against possible biases introduced by the particular distribution of the cases and the methods to be validated were run for a wide set of quality parameters to determine the efficiency of the method. By comparing the accuracy of metric values directly and not glare predictions by threshold, the validation is less limited by site and glazing transmission, it is only limited by the nature of the lighting pathways taken from source to eye. Coupled with the previous validation looking across a range of transmitting and reflecting behaviors [1], we have shown that *raytraverse* can produce reliable annual results across a wide range

of daylight conditions, including those particularly challenging for adaptive sampling. Assessed as a comparison method, the 2-phase DDS for image-based glare metrics is most limited by image resolution, both in terms of accuracy and time. Results are reliable for image resolutions greater than 400x400 pixels for a 180 degree angular fisheye.

The simulated reference validation of *raytraverse*, shows that view-based metrics can be calculated with reasonable computational expense, with less than 0.02 for UGP and DGP and less than 5% MAPE<sub>dn</sub> for  $E_v$ . These lower errors are a more appropriate benchmark for simulated reference validations than the higher values of 10%-20% reported by physical measurement reference validations, where the captured error includes many factors outside the scope of the simulation algorithms. Compared to the uniformly sampled comparison method, *raytraverse* achieved these benchmarks in one-seventeenth the total time. This is the same as the speed-up gained by using a 2-phase approach instead of the per time-step simulations used by the reference method, the foundational method enabling CBDM. For the cases tested here, this translates to simulation times of 75 seconds per point on a modern laptop. Given the complexity of the modeled geometry and open plan requiring a high number of ambient bounces for convergence, it is expected that the models often used as examples in other simulation method research and early stage design will take even less time. Extending *raytraverse* to the equivalent of a zone-based analysis should offer even greater time savings, both from the application of the same adaptive sampling techniques to positional variance and the potential to couple the sampling with rendering methods that take advantage of previously computed values, such as ambient caching or photon mapping. Although, it is possible that some of these savings are overlapping, as they both achieve efficiency by re-using precomputed samples. Perhaps more promising is the potential to couple *raytraverse* with faster hardware solutions, like GPU rendering, as the time savings of these two approaches should be complimentary: *raytraverse* can reduce the number of samples, and GPU rendering can reduce the time needed to calculate these samples. There is now a practical path to developing tools for the zonal analysis of high-accuracy contrast based glare and other visual comfort metrics that currently require image generation, or are limited to a subset of metrics and potential glare inducing scenarios.

Future work should explore the extent of these additional time savings for zonal analysis, as well as the performance of the method under different conditions. These include: simpler models typical of early stage design, dynamic shading controls, the inclusion of electric lighting, and spectral rendering.

## Funding

This research was supported by the Swiss National Science Foundation as part of the ongoing research project "Light fields in climate based daylight modeling for spatio-temporal glare assessment" (SNSF #179067) and, as part of the research collaboration "Phos 4D: Affordance-based evaluation of daylight in antique residential buildings" with Technische Universität Darmstadt and Universität Leipzig, by the German Federal Ministry of Education and Research.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

Thanks to Loisos + Ubbelohde for providing the building geometry and materials as *Radiance* scene files.

## Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at <https://doi.org/10.1016/j.enbuild.2022.112141>.

## References

- [1] S. Wasilewski, L.O. Grobe, R. Schregle, J. Wienold, M. Andersen, Raytraverse: Navigating the lightfield to enhance climate-based daylight modeling, in: 2021 Proceedings of the Symposium on Simulation in Architecture and Urban Design.
- [2] A. Nabil, J. Mardaljevic, Useful daylight illuminances: A replacement for daylight factors, *Energy and Buildings* 38 (7) (2006) 905–913, <https://doi.org/10.1016/j.enbuild.2006.03.013>.
- [3] E. Brembilla, J. Mardaljevic, Climate-Based Daylight Modelling for compliance verification: Benchmarking multiple state-of-the-art methods, *Building and Environment* 158 (2019) 151–164, <https://doi.org/10.1016/j.buildenv.2019.04.051>, <http://www.sciencedirect.com/science/article/pii/S0360132319303038>.
- [4] S. Wilcox, W. Marion, Users Manual for TMY3 Data Sets, Tech. rep. (2008). <https://www.nrel.gov/docs/fy08osti/43156.pdf>.
- [5] J. Mardaljevic, L. Heschang, E. Lee, Daylight metrics and energy savings, *Lighting Research and Technology* 41 (3) (2009) 261–283.
- [6] S. Hoffmann, E.S. Lee, A. McNeil, L. Fernandes, D. Vidanovic, A. Thanachareonkit, Balancing daylight, glare, and energy-efficiency goals: An evaluation of exterior coplanar shading systems using complex fenestration modeling tools, *Energy and Buildings* 112 (2016) 279–298, <https://doi.org/10.1016/j.enbuild.2015.12.009>.
- [7] IESNA, IES LM- 83-12, Approved Method: IES Spatial Daylight Autonomy (sDA) and Annual Sunlight Exposure (ASE) (2012).
- [8] L. Heschang, V.D. Wymelenberg, M. Andersen, N. Digert, L. Fernandes, A. Keller, J. Loveland, H. McKay, R. Mistrick, B. Mosher, et al., Approved method: IES spatial Daylight Autonomy (sDA) and Annual Sunlight Exposure (ASE) Tech. rep., IES-Illuminating Engineering Society, 2012.
- [9] K.V.D. Wymelenberg, M. Inanici, A critical investigation of common lighting design metrics for predicting human visual comfort in offices with daylight, *Leukos* 10 (3) (2014) 145–164, <https://doi.org/10.1080/15502724.2014.881720>.
- [10] J. Wienold, T. Iwata, M. Sarey Khanie, E. Erell, E. Kaftan, R. Rodriguez, J. Yamin Garreton, T. Tzempelikos, I. Konstantzos, J. Christoffersen, T. Kuhn, C. Pierson, M. Andersen, Cross-validation and robustness of daylight glare metrics, *Lighting Research & Technology* (2019), <https://doi.org/10.1177/1477153519826003>, 1477153519826003.
- [11] M.B. Hirling, G.L. Isoardi, V.R. Garcia-Hansen, Prediction of discomfort glare from windows under tropical skies, *Building and Environment* 113 (2017) 107–120, <https://doi.org/10.1016/j.buildenv.2016.08.005>, <http://www.sciencedirect.com/science/article/pii/S0360132316303006>.
- [12] G.J. Ward, T. Wang, D. Geisler-Moroder, E.S. Lee, L.O. Grobe, J. Wienold, J.C. Jonsson, Modeling specular transmission of complex fenestration systems with data-driven BSDFs, *Building and Environment* 196 (2021), <https://doi.org/10.1016/j.buildenv.2021.107774>, <https://www.sciencedirect.com/science/article/pii/S0360132321001815> 107774.
- [13] J. Wienold, Dynamic daylight glare evaluation, in: 11th International IBPSA Conference, Glasgow, Scotland, 2009, pp. 944–951. [http://www.ibpsa.org/proceedings/BS2009/BS09\\_0944\\_951.pdf](http://www.ibpsa.org/proceedings/BS2009/BS09_0944_951.pdf).
- [14] G.J. Ward, The RADIANCE lighting simulation and rendering system, in: Proceedings of the 21st annual conference on Computer graphics and interactive techniques, SIGGRAPH '94, Association for Computing Machinery, New York, NY, USA, 1994, pp. 459–472. doi:10.1145/192161.192286. doi:10.1145/192161.192286..
- [15] N.L. Jones, Fast Climate-Based Glare Analysis and Spatial Mapping, in: Proceedings of Building Simulation 2019: 16th Conference of IBPSA, 2019, p. 8..
- [16] R. Perez, P. Ineichen, R. Seals, J. Michalsky, R. Stewart, Modeling daylight availability and irradiance components from direct and global irradiance, *Solar Energy* 44 (5) (1990) 271–289, [https://doi.org/10.1016/0038-092X\(90\)90055-H](https://doi.org/10.1016/0038-092X(90)90055-H), <http://www.sciencedirect.com/science/article/pii/0038092X9090055H>.
- [17] J. Mardaljevic, The bre-idmp dataset: a new benchmark for the validation of illuminance prediction techniques, *International Journal of Lighting Research and Technology* 33 (2) (2001) 117–134, <https://doi.org/10.1177/136578280103300209>.
- [18] C.F. Reinhart, O. Walkenhorst, Validation of dynamic RADIANCE-based daylight simulations for a test office with external blinds, *Energy and Buildings* 33 (7) (2001) 683–697, [https://doi.org/10.1016/S0378-7788\(01\)00058-5](https://doi.org/10.1016/S0378-7788(01)00058-5), <http://www.sciencedirect.com/science/article/pii/S0378778801000585>.
- [19] C.F. Reinhart, M. Andersen, Development and validation of a radiance model for a translucent panel, *Energy and Buildings* 38 (7) (2006) 890–904, special Issue on Daylighting Buildings. doi: 10.1016/j.enbuild.2006.03.006. <http://www.sciencedirect.com/science/article/pii/S0378778806000703>.
- [20] A. McNeil, E.S. Lee, A validation of the Radiance three-phase simulation method for modelling annual daylight performance of optically complex fenestration systems, *Journal of Building Performance Simulation* 6 (1) (2013) 24–37, publisher: Taylor & Francis \_eprint: doi: 10.1080/19401493.2012.671852. doi:10.1080/19401493.2012.671852. <https://doi.org/10.1080/19401493.2012.671852>.
- [21] E.S. Lee, D. Geisler-Moroder, G. Ward, Modeling the direct sun component in buildings using matrix algebraic approaches: Methods and validation, *Solar Energy* 160 (2018) 380–395, <https://doi.org/10.1016/j.solener.2017.12.029>, <http://www.sciencedirect.com/science/article/pii/S0038092X17311118>.
- [22] N.L. Jones, C.F. Reinhart, Experimental validation of ray tracing as a means of image-based visual discomfort prediction, *Building and Environment* 113 (2017) 131–150, <https://doi.org/10.1016/j.buildenv.2016.08.023>, <http://www.sciencedirect.com/science/article/pii/S0360132316303183>.
- [23] C. Pierson, J. Wienold, M. Bodart, Daylight Discomfort Glare Evaluation with Evalglare: Influence of Parameters and Methods on the Accuracy of Discomfort Glare Prediction, *Buildings* 8 (8) (2018) 94, <https://doi.org/10.3390/buildings8080094>, <https://www.mdpi.com/2075-5309/8/8/94>.
- [24] L.O. Grobe, Photon mapping in image-based visual comfort assessments with BSDF models of high resolution, *Journal of Building Performance Simulation* 12 (6) (2019) 745–758, publisher: Taylor & Francis \_eprint: doi: 10.1080/19401493.2019.1653994. doi:10.1080/19401493.2019.1653994. <https://doi.org/10.1080/19401493.2019.1653994>.
- [25] J. Mardaljevic, Simulation of annual daylighting profiles for internal illuminance, *International Journal of Lighting Research and Technology* 32 (3) (2000) 111–118, publisher: SAGE Publications. doi:10.1177/096032710003200302. <https://journals.sagepub.com/doi/abs/10.1177/096032710003200302>.
- [26] D. Bourgeois, C.F. Reinhart, G. Ward, Standard daylight coefficient model for dynamic daylighting simulations, *Building Research & Information* 36 (1) (2008) 68–82, publisher: Routledge \_eprint: doi: 10.1080/09613210701446325. doi:10.1080/09613210701446325. <https://doi.org/10.1080/09613210701446325>.
- [27] R. Schregle, L. Grobe, S. Wittkopf, An out-of-core photon mapping approach to daylight coefficients, *Journal of Building Performance Simulation* 9 (6) (2016) 620–632. arXiv:https://doi.org/10.1080/19401493.2016.1177116. doi:10.1080/19401493.2016.1177116. doi: 10.1080/19401493.2016.1177116..
- [28] R.S. Overbeck, C. Donner, R. Ramamoorthi, Adaptive wavelet rendering, *ACM Trans. Graph.* 28 (5) (2009) 1–12, <https://doi.org/10.1145/1618452.1618486>.
- [29] P. Shirley, K. Chiu, A Low Distortion Map Between Disk and Square, *Journal of Graphics Tools* 2 (3) (1997) 45–52, <https://doi.org/10.1080/10867651.1997.10487479>.
- [30] P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S.J. van der Walt, M. Brett, J. Wilson, K.J. Millman, N. Mayorov, A.R.J. Nelson, E. Jones, R. Kern, C.J. Carey, I. Polat, Y. Feng, E.W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E.A. Quintero, C.R. Harris, A.M. Archibald, A.H. Ribeiro, F. Pedregosa, P. van Mulbregt, SciPy 1.0 Contributors, SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, *Nature Methods* 17 (2020) 261–272, <https://doi.org/10.1038/s41592-019-0686-2>.
- [31] D.P. Mitchell, The antialiasing problem in ray tracing, *Advanced Topics in Ray Tracing, In ACM SIGGRAPH Course Notes*, 1990.
- [32] S. Subramanian, Daylighting Simulations with Radiance using Matrix-based Methods, Tech. rep., LBNL (Jun. 2017). <https://www.radiance-online.org/learning/tutorials/matrix-based-methods>.
- [33] R. Perez, R. Seals, J. Michalsky, All-weather model for sky luminance distribution—Preliminary configuration and validation, *Solar Energy* 50 (3) (1993) 235–245, [https://doi.org/10.1016/0038-092X\(93\)90017-I](https://doi.org/10.1016/0038-092X(93)90017-I), <http://www.sciencedirect.com/science/article/pii/0038092X9390017I>.
- [34] S. Wasilewski, L.O. Grobe, J. Wienold, M. Andersen, A Critical Literature Review of Spatio-temporal Simulation Methods for Daylight Glare Assessment, *SDAR\* Journal of Sustainable Design & Applied Research* 7 (1). doi:https://doi.org/10.21427/87r7-kn41. <https://arrow.tudublin.ie/sdar/vol7/iss1/4>.
- [35] P.R. Tregenza, Uncertainty in daylight calculations, *Lighting Research & Technology* 49 (7) (2017) 829–844, <https://doi.org/10.1177/1477153516653786>.
- [36] Y. Yang, R.M. Luo, W. Huang, Assessing glare, part 3: Glare sources having different colours, *Lighting Research & Technology* 50 (4) (2018) 596–615. arXiv:https://doi.org/10.1177/1477153516676640, doi:10.1177/1477153516676640. doi: 10.1177/1477153516676640..
- [37] S. Jain, Does glazing color influence our perception of discomfort glare from daylight? <http://infoscience.epfl.ch/record/290187>..