

A Spatially Augmented Reality Sketching Interface for Architectural Daylighting Design

Yu Sheng, *Student Member, IEEE*, Theodore C. Yapo, *Member, IEEE*,
Christopher Young, and Barbara Cutler, *Member, IEEE*

Abstract—We present an application of interactive global illumination and spatially augmented reality to architectural daylight modeling that allows designers to explore alternative designs and new technologies for improving the sustainability of their buildings. Images of a model in the real world, captured by a camera above the scene, are processed to construct a virtual 3D model. To achieve interactive rendering rates, we use a hybrid rendering technique, leveraging radiosity to simulate the inter-reflectance between diffuse patches and shadow volumes to generate per-pixel direct illumination. The rendered images are then projected on the real model by four calibrated projectors to help users study the daylighting illumination. The virtual heliodon is a physical design environment in which multiple designers, a designer and a client, or a teacher and students can gather to experience animated visualizations of the natural illumination within a proposed design by controlling the time of day, season, and climate. Furthermore, participants may *interactively* redesign the geometry and materials of the space by manipulating physical design elements and see the updated lighting simulation.

Index Terms—Spatially augmented reality, global illumination, radiosity, and daylighting design

1 INTRODUCTION

THE particular area of architectural design we explore is daylighting: the use of windows and reflective surfaces to allow natural light from the sun and sky to provide effective and interesting internal illumination (Figure 1). Appropriate daylighting strategies can reduce energy consumption for electric lighting and create more aesthetically interesting and comfortable architectural spaces. The overall aim of a successful daylighting design is to increase the amount of useful daylight in an architecturally satisfying way while avoiding the problematic aspects of natural illumination including the risk of glare or overheating.

A *heliodon* is a traditional daylighting analysis device in which a small-scale physical model (often 1/4" = 1') is affixed to a platform and rotated relative to a fixed light source that represents the sun (Figure 2a). Alternatively, in more complex devices, the light source may be mechanically moved around a stationary platform. By studying the distribution of light within the model, the designer gains instantaneous and intuitive qualitative feedback on direct sun penetration and the corresponding indirect illumination.

Our new *virtual heliodon*, shown in Figure 2b&c, shares many of the features of the traditional heliodon and includes a number of important additional advantages. With a heliodon the user must awkwardly peer through windows (possibly blocking light) or place tiny video cameras within the model; in contrast, the virtual heliodon does not require a physical ceiling on the model, allowing easy inspection of the inte-

rior spaces. Furthermore, initial construction and revision of models for the virtual heliodon is faster than for heliodon models because the corners of the model need not precisely align. In the virtual heliodon we “fill-in” cracks between walls that in a traditional heliodon would allow light to incorrectly leak into the model. Editing surface materials for a heliodon model is labor intensive and impossible if appropriate scale versions of the material are not available (e.g., Venetian blinds), but with our system surface and window materials can be changed digitally. Traditional heliodons only simulate direct illumination from the sun and do not automatically adjust the intensity with altitude. Our system performs a quantitatively-accurate simulation of illumination from both the sun and sky and also models climate variations.

We begin with an efficient, interactive, and accurate global

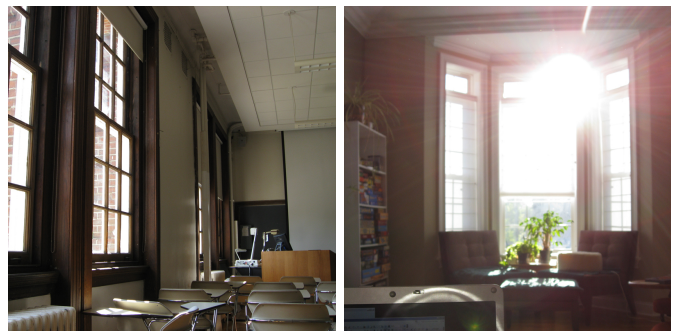


Fig. 1. Large, north-facing windows can be effectively used to illuminate spaces such as the classroom shown in the left image. However, glare (when the brightest part of the room is more than seven times brighter than the darkest part of the room) can reduce contrast and visibility. Direct sun far exceeds this range.

• All authors are with Department of Computer Science, Rensselaer Polytechnic Institute, 110 Eighth Street, Troy, NY USA 12180.
E-mail: shengyu@cs.rpi.edu, yapot@rpi.edu, youngc@rpi.edu, cutler@cs.rpi.edu

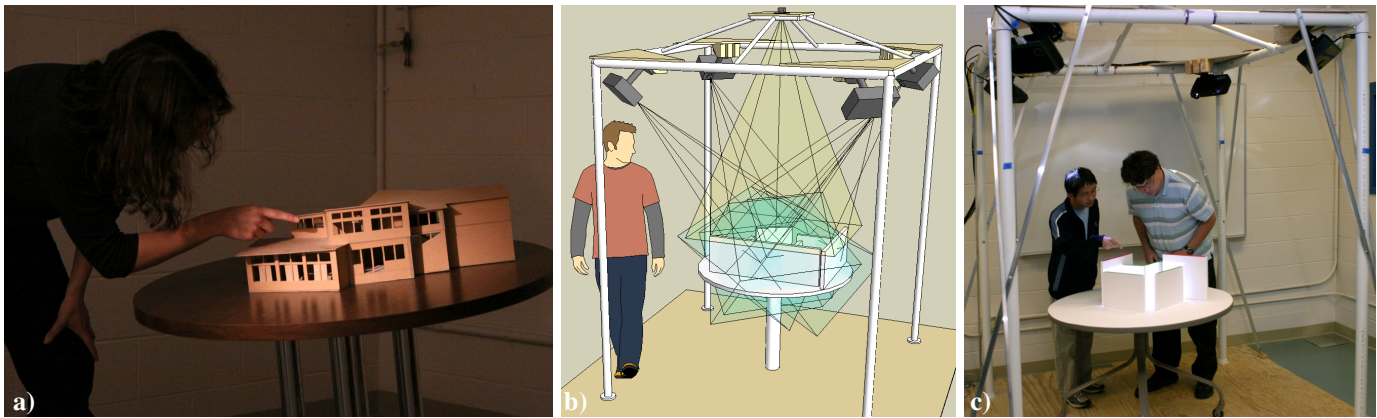


Fig. 2. a) The depth of direct light beam penetration into an architectural model is easily ascertained with the traditional *heliodon*. The relative orientation of the model and light source is adjusted to represent different sun positions. b & c) Our *virtual heliodon* allows interactive redesign and enhanced visualization and exploration of the interior spaces.

illumination algorithm for daylighting. We draw from the body of literature on virtual and augmented reality to build a novel daylighting tool based on this rendering method. Our interactive tangible daylighting design system can be used by novice or experienced designers who need not be experts in daylighting technology or advanced graphical simulations.

2 BACKGROUND AND RELATED WORK

Static, two-dimensional photographs or digital pre-renderings of a space are often insufficient for full appreciation of a scene. For example, only after visiting concert spaces with spectacular natural lighting did the clients and architects of the Schermerhorn Symphony Hall in Nashville choose to incorporate daylighting into their project (a significant expense acoustically) [1]. The virtual heliodon is capable of providing similarly engaging visualizations of the dynamic nature of natural illumination.

2.1 Virtual Environments

For many years, virtual reality environments have been dominated by head-mounted displays and CAVE-style environments [2], [3]. Pertinent examples involving architectural and/or lighting design include navigation through hospital operating room designs [4], interior lighting design [5], and accurate global illumination from daylighting for car interiors [6].

Research in Spatially Augmented Reality (SAR) [7] addresses some of the physically-immersive criteria that we wish to leverage. Examples include the Office of the Future [8] and Everywhere Displays [9], [10], which use existing wall and desk surfaces to expand the area of the traditional computer interface. With Shader Lamps [11], [12], complex physical models are “animated”, for example, by projecting a video of a spinning wheel onto a stationary car model or a facade texture onto a physical architectural model [13]. The geometry of the physical objects is known *a priori* and the surfaces are assumed to be a uniform diffuse white material. The implementation of most SAR systems involves multiple

automatically-calibrated cameras and projectors [14]. The projectors may also be used to generate patterns of structured light to reconstruct the geometry of a multi-planar display [15], [16]. Different mechanisms for interacting with these projected user interfaces have been proposed, prototyped, and evaluated for effectiveness. Examples include using a finger or laser pointer to indicate a position on the projected surface [10], [17] or gesturing with a handheld projector [18]. One or more cameras monitor the projection surface to detect and interpret the user’s actions.

Two obvious problems when using projection onto real-world objects are occlusions and shadows cast when the user approaches the projection surface. This problem can be partially alleviated by positioning the projectors on the ceiling at shallow angles to the target surface. By using multiple overlapping projectors and automatic detection of the user and dynamic scene content, many of these shadows can be removed or reduced [19]. The angle and distance between the projector and surface must be taken into account when determining the focal length and relative pixel intensity. Color correction is necessary for multi-projector displays due to slight color mismatches between different projectors and projection onto surfaces with different material properties or textures [20]. Finally, when projecting onto multi-planar scenes or non-planar objects, there will be secondary scattering of the projected light. If the scene geometry and surface reflectance properties can be estimated, Raskar et al. [11] note that inverse global illumination could be applied to handle and undo this scattering (with some important restrictions), which is related to work on reverse radiosity [21], [22] and “common illumination” between synthetic and real objects [23], [24], [25], [26].

2.2 Interactive Global Illumination for Daylighting

The complexity of daylighting simulation, which accounts for slow rendering speeds and thus a decreased potential for interactive creative exploration of daylighting design, arises from two factors. First, natural illumination in the built environment is provided not only from direct parallel rays of sunlight, but

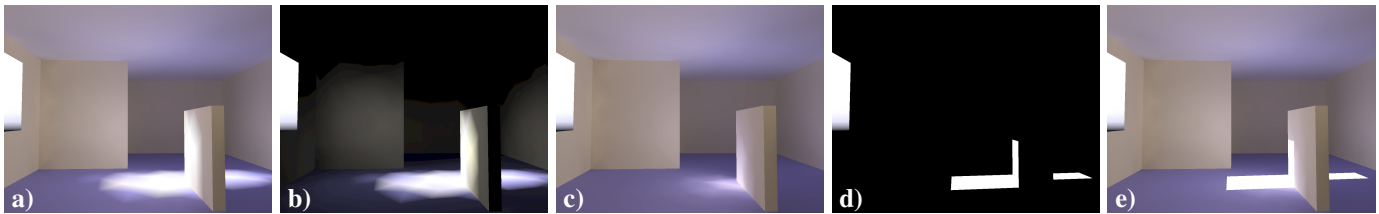


Fig. 3. a) The classic radiosity solution does not capture hard-edged shadows. We factor the radiosity solution into b) first bounce direct illumination and c) indirect illumination by subtracting b) from a). d) Shadow volumes are used to generate per-pixel hard shadows. e) A hybrid radiosity/shadow volumes rendering is generated by adding c) and d).

also from omni-directional illumination from the non-uniform, seasonally-varying sky hemisphere. The hemispherical distribution and relative intensity of the sky for different weather conditions (clear sky \rightarrow overcast sky) are calculated from standard models [27].

Second, surfaces receive illumination not only from direct light from the sun and sky, but also from *indirect* illumination that first reflects off one or more other surfaces in the scene. Real-time rendering systems based on the classic graphics rasterization pipeline (e.g., OpenGL and DirectX) are unable to accurately simulate indirect illumination. Physically-accurate patch-based radiosity [28] or Monte Carlo ray tracing [29] methods can simulate these effects, but require significantly more computational resources.

To facilitate interactive design, daylighting analysis tools must have adequate response time when the user edits the geometry or materials of the model. Decreasing the rendering time for global illumination is an active area of research [30], [31], [32], [33]; unfortunately, without explicit knowledge of these rendering algorithms, most architects are not able to appropriately prepare their models or tune parameters for these rendering software.

In most architectural scenes involving daylighting, light transfer due to diffuse reflection from surfaces dominates the indirect lighting. Additionally, hard shadows from the direct sun provide important visual cues that are necessary to understand the aesthetics of the space. Furthermore, the possibility of glare due to high contrast in the illumination values at the shadow boundaries must be considered. Per-pixel hard shadows greatly improve the perceived visual quality, but are usually not critical for computing accurate indirect illumination in diffuse-dominant scenes. Thus, we use a hybrid technique (Figure 3) [34] that uses radiosity [28] to compute the diffuse reflection between faces on a coarse per-face basis, and replaces the direct illumination from the sun with per-pixel computations using multi-pass stencil shadow volumes [35].

2.3 Daylighting Simulation Software

A wide array of architectural CAD lighting design software tools are currently available. Tools for use in the early stage of design are either quantitative in output and highly restrictive in model complexity [36], [37] or limited to qualitative renderings of direct illumination only [38], [39]. In other words, these tools cannot be used to determine if the daylighting illumination levels in the current design will meet task-specific recommendations [40]. At the other end of the

spectrum lie high-end rendering tools (e.g., Radiance [41]), which allow arbitrary model complexity at the expense of computation time. These tools are not suitable for use at the early stage of design because the calculation and rendering times involved (from minutes to hours) restrict exploration of alternative designs and annual variations. By exhaustively pre-computing the lighting solution for simple fixed geometries, the various daylighting analysis programs which use Radiance as the rendering engine are able to allow limited interactive visualization and evaluation.

A critical gap in daylight modeling remains: for moderately complex geometries such as those found in schematic design, no program today enables climate-based, time-varying performance analysis capabilities in combination with an interactive, highly visual, and creativity-promoting design exploration process. Thus, daylighting simulation software is *very seldom used in academic or professional practice to inform design*. As a result, we are left with poorly designed spaces that exhibit uneven natural illumination conditions. These conditions are often corrected by blocking out nearly all of the daylight and relying primarily on electric lighting (Figure 4). Our project aims to fill this gap in the available daylighting design tools by providing a tangible and interactive system that allows iterative analysis and modification of the geometry during the design process.

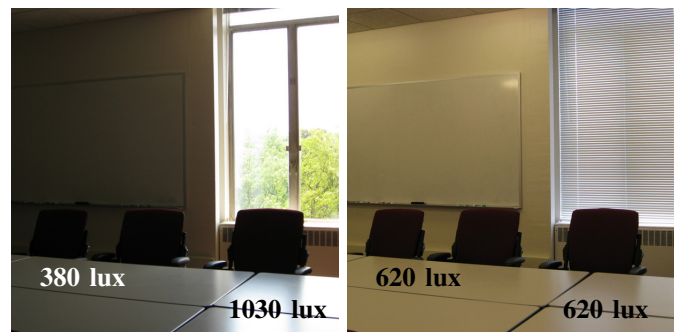


Fig. 4. On a sunny day with the blinds up (left image), parts of this conference room table are too bright (> 1000 lux) for comfortable reading while other parts are too dark (< 500 lux). By lowering the blinds we can partially correct the situation by removing excess illumination from the center of the table. Altogether too often in practice, lighting engineers settle for the least creative and least efficient solution, selecting light fixtures for night or when the blinds are fully closed (right image).

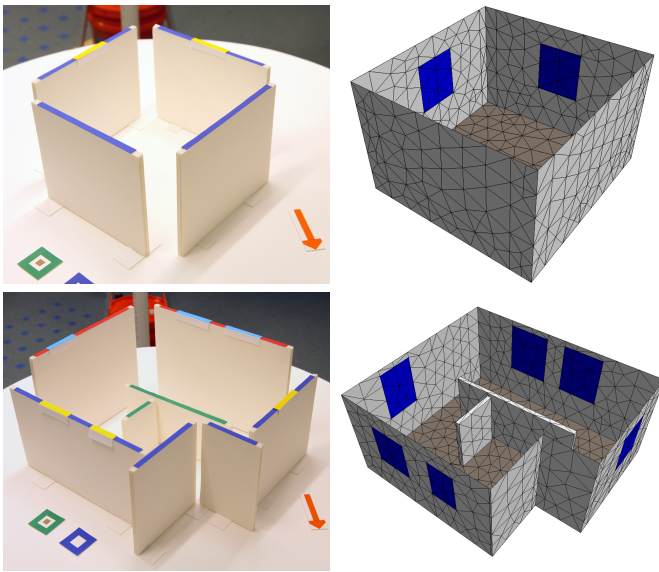


Fig. 5. The user *sketches* a design with physical walls (left images), and the system interprets the sketch as a closed watertight architectural model (right images). Different types and sizes of windows are specified with colored markers on the top edges of the walls. Cracks between the exterior walls are filled in. The ceiling polygons of the digital model are omitted for this visualization.

3 OVERVIEW

We have used spatially augmented reality to build a virtual heliodon system [42], [43] that complements modern desktop architectural daylighting design software tools. The user positions a set of small-scale physical walls within the workspace to “sketch” the 3D geometry of their design. Tangible walls are chosen to project daylighting simulation on instead of pure virtual reality solutions because they are low-cost, convenient, and easy for users to rebuild. Images captured by a camera mounted above the scene are processed to detect the wall positions. Gaps between the wall are filled to construct a closed 3D mesh (Figure 5). The daylighting solution in the virtual 3D building is computed with the hybrid radiosity/shadow volume global illumination rendering method (shown in Figure 3), accounting for illumination from both the sun and sky. This illumination solution is then displayed on the physical walls by four calibrated projectors.

The efficiency of the hybrid rendering method makes it possible for users to interact with daylighting simulations in our system. The user can explore the high-dimensional configuration space of the design by adjusting the position of wall modules to manipulate the geometry of the design. Through a wireless remote mouse, the user can vary the external conditions such as the sun position (time of day and day of the year) and weather conditions. The simulation results are recomputed and updated at interactive rates. In addition to visualizing the environment for a single point in time, the system allows time-lapse animations as the sun moves across the sky during the course of a day, throughout the season, or under different weather conditions.

The novel contributions of our project include:

- Dynamic and engaging visualization of daylighting variation for different moments (time of day and day of the year).
- Spatially augmented reality for visualization and exploration of natural illumination by multiple simultaneous users.
- Interactive redesign of the scene geometry and materials with online recomputation of the resulting lighting simulation.
- A rich language of wall primitives and other design elements for sketching a wide array of architectural spaces.

3.1 Physical Setup

Our table-top SAR system prototype (Figure 2b&c) centers around a standard 30” high, 42” diameter table. The surface of the table is covered with matte white museum board.

We have constructed a set of small-scale (1” = 1’) planar and cylindrical wall modules of three heights. All walls are made from lightweight, 3/8” thick matte white “foam core” and have small strips of museum board glued to the bottom of the wall for stability so they are easily balanced perpendicular to the table surface. The top of each wall is colored to identify its height: red walls are 10” tall, blue walls are 8” tall, and green walls are 4” tall. The shortest walls are used for interior partitions, similar to the cubicle walls in many modern offices. We have found that these wall modules are sufficiently complex and expressive to represent a wide variety of office and residential programs and styles.

A simple frame of 4” and 2” diameter PVC pipe with aluminum and plywood bracing for rigidity is built around the table. The frame is 6’ square at the base and 8’ tall allowing for easy movement around all sides of the table. A camera is mounted approximately 5’ above the center of the table and oriented downward to capture the table and walls. Four projectors are mounted at the upper corners of the frame and oriented to project to the center of the table at an approximately 45° downward angle. The distance from the projectors to the table center is approximately 5’. The frame is not necessary if the cameras and projectors are mounted from the ceiling instead. Four low-wattage fluorescent tubes mounted on the underside of the frame are used to illuminate the table during image capture.

3.2 Design Environment

The table top provides the canvas upon which a designer sketches an architectural design. Our sketching tools consist of a number of *primitives* that the user arranges on the table top to indicate a particular room or building design. The primitives allow for interior and exterior walls, cubicle dividers, different window styles, material properties, and building orientation.

Three different styles of window can be placed on any of the planar walls. Window primitives are constructed from simple U-shaped matte white cardstock with their top surface covered in one of three different colors. The window primitives slide over the top of any planar wall to indicate the horizontal placement of the window. Our initial primitives allow the

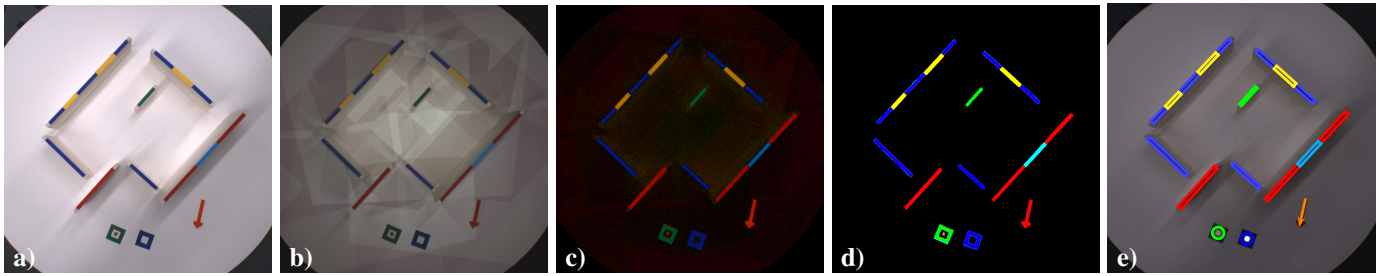


Fig. 6. a) A representative scene shown with the room lights on, appears b) dark and cluttered when captured during actual system conditions. The multiple shadows in the scene are caused by the low-wattage fluorescent working lights and the non-zero projector black level. c) Color contrast in the scene is enhanced by a non-linear transformation and d) pixels are classified by sequential thresholding. e) Vector primitives are robustly fit to connected components in the image, and subsequently used to determine 3D positions.

designer a choice of standard, picture (taller than standard), and transom (high, for privacy) window styles. Wide windows may be specified by overlapping two or more window primitives of the same color. As with the wall primitives, the color of the window markers is used to determine their type from images taken by the overhead camera. In addition to windows on walls, the designer may place skylights into the ceiling of the room by placing light blue colored markers on the table surface. The location of these markers is projected upward to the ceiling to indicate the actual skylight position. In addition to the planar walls and dividers, users may indicate curved wall surfaces. We allow curved physical wall primitives that are circular arcs of arbitrary radius and length. More complex curved wall surfaces (an “S”-curve, for example) can be constructed from an appropriate arrangement of circular arcs.

An important consideration for daylighting design is the overall orientation of a room or building with respect to the cardinal geographic directions. Thus, an orange “north arrow” primitive may be placed anywhere on the table surface. The direction of the arrow orients the model with respect to the simulated sun and sky conditions. Similarly, the designer exercises control over the surface reflectance properties of the space through a set of material “tokens”. In each case, a “paint chip” in the center of the token specifies the desired color. A green-outlined token is used to specify the floor material. Similarly, a blue-outlined token is used to select the global wall color, and red-outlined tokens can be used to indicate the color of the single wall closest to that token.

4 ALGORITHMS & IMPLEMENTATION DETAILS

4.1 Image Processing: Primitive Detection

The camera provides an overhead view of the model (Figure 6a). We considered using ARToolkit [44] for tracking, but found the required large marker size to be problematic in this application. Instead, we have color coded the top edges of the walls, which are easily visible from this vantage point. Since this single viewpoint precludes accurate photogrammetric determination of object heights, the tops of objects are color-coded to indicate distance above the table surface. The height and window coding colors are carefully selected to

allow robust detection in difficult lighting conditions: red, green, and blue indicate the three wall heights and cyan, magenta, and yellow are used for the different window markers. Furthermore, since our walls are constrained with respect to distance from the camera, the total error can be reduced by robustly fitting 2D primitives to the objects before back-projecting positions to obtain 3D coordinates.

Detection of primitives in an overhead image is complicated by the fact that we maintain a low level of illumination in the room to facilitate viewing and increase contrast in the projected lighting solutions. As shown in Figure 6b, the low ambient illumination combined with the relatively high black-level of the projectors results in a noisy image cluttered with shadows and uneven illumination. To accurately detect objects in the scene, we use a color-based classifier which is insensitive to illumination variations in the scene.

For an image of a new design, we first perform a nonlinear color transformation to enhance the contrast between foreground and background. For a given RGB color triplet, $P = (R_p, G_p, B_p)$, we first calculate the minimum and maximum components: $P_{max} = \max(R_p, G_p, B_p)$, $P_{min} = \min(R_p, G_p, B_p)$. We then transform the triplet according to $P' = \frac{P - P_{min}}{P_{max}}$. This transformation effectively moves near-neutral colors (the table surface, regardless of illumination) toward black, while increasing the saturation of colored pixels, as shown in Figure 6c. Since the resulting colors have at most two non-zero components, a simple ordering and thresholding of components accurately classifies pixels. Our initial object detections are performed in image space. Once we have assigned each pixel to a color class, we perform a connected-components analysis to separate the image into discrete objects. Since both wall markers and material tokens share the same red and green colors, material tokens are distinguished by their topological Euler number – material tokens contain a hole, while walls are solid regions.

Next, we estimate the 3D positions of planar wall primitives, which are assumed to be vertical and of the exact, fixed height specified by their color. Since the perspective transformation introduced by the camera preserves straight lines, estimation of wall surfaces can be performed in image space. To estimate a wall’s position, we first fit a single line to all of the edge pixels of a connected color region. This line, although a poor

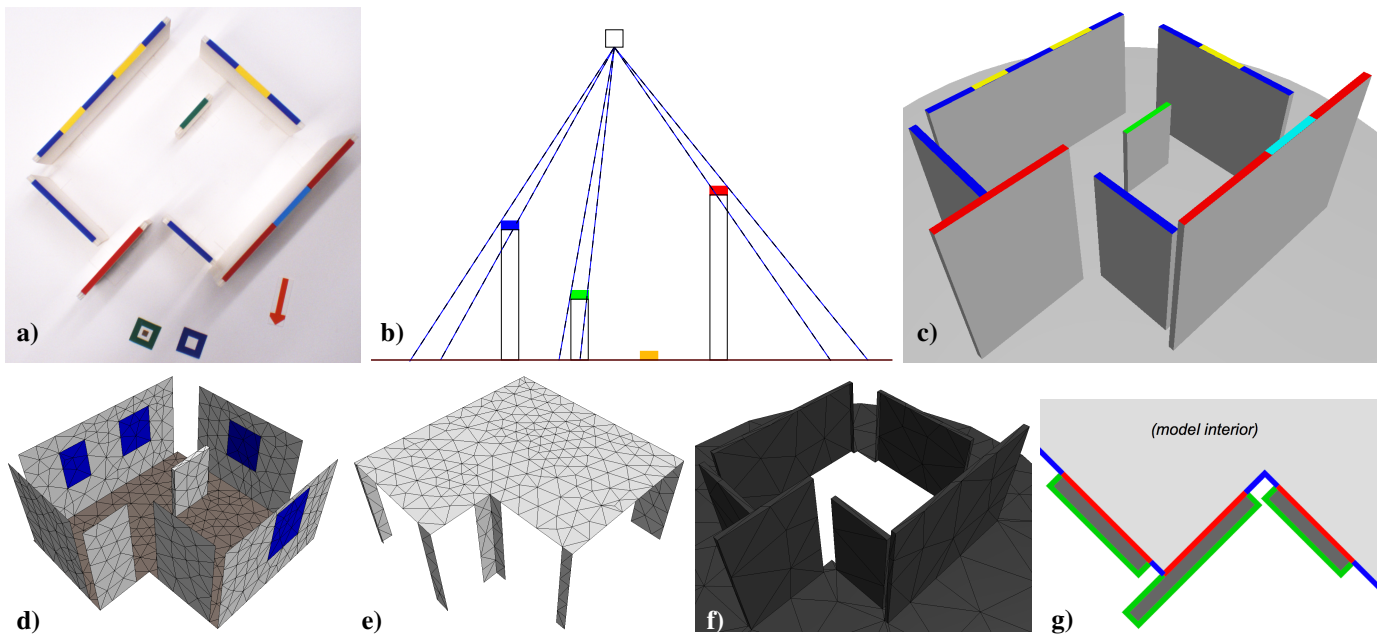


Fig. 7. 2D image processing of a) the image captured by the overhead camera is used to detect the tops of each wall. The camera calibration parameters are used to b) back-project rays from the camera to the known wall heights, yielding 3D positions of c) the physical wall geometry. Then, a closed 3D model of the intended room geometry is constructed and meshed with polygons that are roughly uniform in area. These polygons are classified as d) physical projection surfaces and e) fill-in polygons. Additionally, f) the full extent of each physical wall (the “extra” part) is also represented in order to correctly account for occlusions between the projectors and physical projection surfaces. The diagram in g) shows the projection surfaces in red, the fill-in surfaces in blue, and the extra surfaces in green. The daylighting simulation is performed on the union of the projection and fill-in geometry.

estimate of either inner or outer wall surface, separates the edge pixels into two sets that can then be individually fit to find both surfaces. In practice, we use an iterative line fitting that rejects points lying more than 2 pixels from the line as outliers; this produces a robust estimate of the wall surface, effectively ignoring noise pixels and the effect of pixels lying on the short wall “ends”. The length of the wall is estimated by projecting all colored pixels onto the wall surface lines to find the maximum extents. Since shadows and occlusions introduced by window markers can break connectivity of a single wall, a post-processing step merges parallel walls separated by less than a threshold distance.

To calculate the position of a wall in 3D world coordinates, the calibrated camera projection matrix is used to back-project rays through 2D wall corner points [45]. Dropping virtual “plumb lines” from these corners to the known table surface height allows full 3D models of the walls to be reconstructed (Figure 7b). A simple modification of this scheme improves robustness greatly: each colored wall top stops 1/2” from each end of the wall, allowing for easy separation of physically touching walls during the connected components step; the length of the virtual walls are subsequently extended to match. This scheme allows walls of arbitrary thickness and length and in practice, only a few fixed wall heights are required to model most architectural spaces.

Skylight detection is similar to that for walls, except that all four edges are required, instead of just the longest two. To estimate skylight edges, we use RANSAC [46] to estimate

each edge in turn. Starting with the full set of edge pixels for the object, we use RANSAC combined with a simple robust line fitting algorithm to fit a single line. Pixels which are considered part of this line (their distance falls within a specified threshold) are then removed, and the process is repeated to extract the remaining edges.

While linear objects in the scene can be estimated in image space, pixels corresponding to curved walls must first be back-projected into 3D, since circular arcs are not preserved under perspective transformations. To estimate the curved walls, we first fit a single circle to all edge points of the detected object, and then use the estimated radius to divide the edges into an inner and outer set, similar to our wall estimation algorithm. Once points corresponding to the two arcs are separated, we estimate new radii for each, using a common center point.

To detect the paint chips that specify material properties, we use the camera as a crude spectrophotometer to estimate the RGB values of these colors. To improve color accuracy, the white border around each color swatch is used to perform local white balancing and illumination compensation before an average RGB value for the swatch is computed. Finally, the north indicator arrow is distinguished from other objects in the scene by its asymmetry. Once classified, the centerline of the arrow is determined by a best-fit line, and one of two possible directions of the arrow chosen by comparing the center of mass of the object to the geometrical center of its extent.

4.2 Model Construction and Remeshing

The 3D wall geometry sketched by the user and detected in the previous image processing stage is not a “watertight” model. The walls may have gaps between them and the corners of the physical walls may overlap and stick beyond their natural intersection point. Furthermore, neither ceiling nor windows are explicitly included in the sketch and must be inferred. Thus, the geometry must be processed to build a closed 3D triangle mesh model that is appropriate for computing a radiosity global illumination solution.

Our initial implementation required that the exterior room geometry be star-shaped, that is, a central point within the volume of the room could “see” every face. This assumption allowed us to construct a closed model by sorting the faces in clockwise order by angle around this point and intersecting neighboring faces.

We subsequently developed the following algorithm to handle more general (non star-shaped) room designs. First, all edges of the detected wall and skylight geometry are used to build a 2D arrangement of lines on the plane of the table (third row of Figure 8). Cells of the arrangement that overlap the area of a wall primitive are labeled. Next we determine the *enclosure fraction* of each of the remaining cells, i.e., how much of the 360 degree “view” from the centroid of the cell is obstructed by walls. In a tightly constructed model, with very small gaps between the walls, the enclosure fraction for interior cells will be very high. In contrast, when the gaps are larger the enclosure fraction will decrease, especially for cells near the gap. Cells outside of the room will generally have enclosure fractions much less than 50%. Simply labeling all cells having enclosure fractions higher than a specified threshold works for many models, but requires careful parameter tuning. Thus, we iteratively select the unlabeled cell with the highest enclosure fraction and all of the cells that lie in the same combination of wall primitive halfspaces. Furthermore, rather than halting when an arbitrary enclosure fraction threshold is no longer met, the iteration process halts when all walls have been incorporated into the model.

Once all cells have been labeled, we systematically generate a closed, watertight model appropriate for CAD modeling packages and advanced simulations such as physically-accurate rendering. Each interior cell triggers the construction of floor and ceiling triangles. If the cell lies within a skylight, the material of the ceiling triangles is set to glass. The edges of interior cells that border non-interior cells trigger the construction of wall polygons. These wall polygons are subdivided into window polygons as necessary.

Once the closed model is constructed, it is remeshed with a combination of edge split, edge flip, edge collapse, and move vertex operations [47] to arrive at a triangle mesh with approximately 2000 polygons, an appropriate number for interactive radiosity simulations (fourth and fifth rows of Figure 8).

4.3 Projection of Daylighting Solution

The daylighting solution is computed on the closed model from the previous section, but must be displayed on the

physical non-closed model. Therefore, these gaps and unused portions of the walls are specifically labeled (Figure 7g). The patches corresponding to the *projection surfaces*, the implied *fill-in surfaces* necessary to make a closed model, and the additional unused *physical occlusions* are identified and processed to ensure that the model is correctly rendered. The fill-in surfaces include the ceiling and gaps between the walls. Although not displayed in the final rendering, these patches are part of the closed building model and are thus used in the computation of the radiosity solution (Figure 7e). Extra, unused portions of the physical walls that are not included as projection surfaces (Figure 7d) because they are taller than the ceiling or because they overlap with other walls are clipped from the closed model and labeled as physical occlusions (Figure 7f). They will not be taken into consideration for the radiosity computation, but they must be drawn black in the final rendering because they may cause real physical occlusions between the projectors and projection surfaces.

A hybrid technique that combines per-patch radiosity computation and per-pixel shadow volumes (Figure 3) is used to interactively render images that simulate the daylighting conditions in the room. A perspective projection matrix, which represents the projector’s intrinsic and extrinsic parameters, is used to render an image of the simulation results for each of the four projectors. Since the radiosity global illumination rendering method is view-independent, the lighting solution is calculated once and the illumination is correct when viewed from any angle [11].

The projectors have been positioned at an approximately 45° downward angle, which gives good coverage and visibility of the scene; however, there are still significant occlusions between the walls and projection surfaces that must be considered to produce correct and consistent projections. The display of each triangle is assigned to one or more projectors for display. When multiple projectors are available to cover a certain patch, they are appropriately blended to reduce visual artifacts due to slight miscalibration of the projector geometry and color. This is done by intensity blending with attenuating intensities in these regions [7].

We adopt the following strategy to smoothly illuminate the physical walls and floor of the model. First, we remesh the digital model along the visibility discontinuities [48], that is, the shadow edges formed between the projector centers, the physical walls, and the projector frusta (Figure 9). Then, each patch can be labeled with the projectors that can see that area by tracing a ray from the centroid of the patch back to the projector center to check for occlusions. To prevent projection at grazing angles, which magnifies errors in calibration and decreases the effective resolution, we require that the dot product between the projector and the surface normal be greater than 0.2. Thus, if the angle between the viewing direction and the surface is less than 12 degrees, the projector is discarded as a candidate for projection to that surface. Next, blending weights are assigned per vertex by taking the intersection of the projector visibility labels of the neighboring triangles and normalizing the blending weights so that they sum to one. In other words, the blending weight for a projector at a vertex is non zero only if that projector

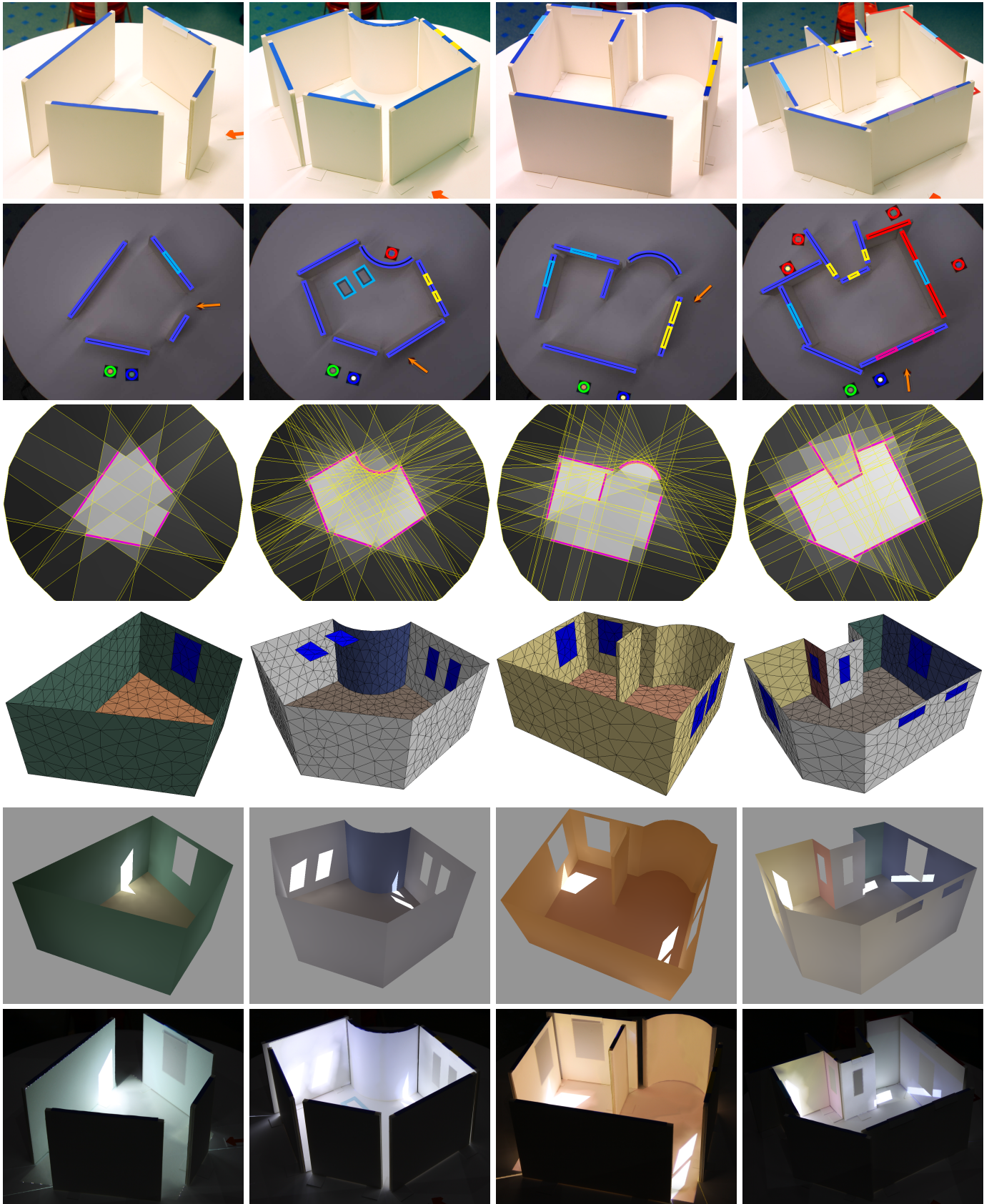


Fig. 8. Example designs produced with our system. Top to bottom: 1) Physical sketch input. 2) Overhead camera image with detected primitives annotated. 3) Arrangement with cells shaded by *enclosure fraction*. 4) 3D triangulated inferred design. 5) Rendering of daylighting simulation. 6) Image of daylighting solution being projected on physical walls.

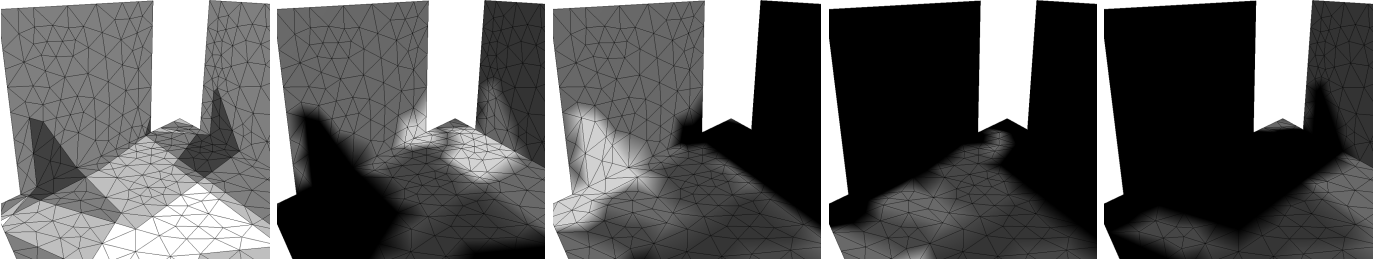


Fig. 9. To create a smooth transition between the coverage of multiple projectors we use intensity blending. The left image shows the model remeshed to follow projector visibility discontinuities (shadows) [48] (left image). The remaining images show the blending weights used for the four projectors.

can see all neighboring triangles. A demonstration of the final simulation projection with intensity blending is shown in the last row of Figure 8.

Treating the projector lamp as a point light source, the radiance reflected from the physical surface is related to the diffuse reflectance ρ of the surface, distance r between the surface and projector and the angle θ subtended by the surface normal and the vector from the surface to the projector. In order to achieve the same radiance in the physical world as in the virtual environment, radiance adjustment [11] is used. We apply the equation

$$I = E \frac{r^2}{\rho \cos \theta}$$

to each vertex that is rendered (I is the radiance value after adjustment, and E is the value from radiosity solution).

Our radiosity solution produces linear radiance values. Since the projectors apply gamma correction ($\gamma = 2.2$), we must apply the inverse gamma transformation to pixel intensities before display. This is achieved by rendering images in sRGB color space, which is a standard feature referred to as sRGB framebuffer objects in OpenGL 3.0 [49].

4.4 Camera and Projector Calibration

Calibration of our system comprises two individual components. The camera must be geometrically calibrated in order to produce accurate 3D models of the scene, and each projector must be geometrically calibrated to project correctly onto the wall surfaces.

The overhead camera is calibrated using Zhang’s algorithm [50]. We created a calibration target consisting of 212 black and white corner marks on a white background (Figure 10a). The target is printed directly on a 40” square sheet of “Gatorboard”, which provides a flat and rigid surface. Extraction of the corners from camera images is facilitated by the addition of four colored disks to the outer edge of the target. These disks are easily detected using a color-based classifier, and the centroids of the colored regions are used to estimate a planar homography between the measured physical target model and each captured image. Using this homography, we are able to efficiently scan the image for the grid of calibration corners. Isolated corners are used for calibration points since they are immune to bias induced by perspective and radial lens

distortion [51]. To extract the corner positions with sub-pixel accuracy, we fit a quadratic function to a blurred version of the image region surrounding each corner, and compute the location of the extremal points of the resulting polynomial [52]. Approximately 30 images of the calibration target in various poses are taken, roughly filling the active 3D volume of the model space. Zhang’s algorithm is used to estimate the intrinsic camera parameter matrix, K , from these images. Finally, a single image is taken of the calibration target lying flat on the table surface. The extrinsic parameters estimated for this image are used to establish the world coordinates for the system. This fixes the table surface as the x - z plane, with the y -axis normal to the table top. Once the images are captured, the calibration process is automatic and takes less than a minute.

The four projectors are also calibrated with a traditional camera model, using Tsai’s method [53]. Tsai’s algorithm requires a set of correspondences between projector image pixels and 3D points in our world coordinate system. We generate these correspondences using a structured light method. For each projector, we project a sequence of Gray-coded patterns (Figure 10b) onto a set of nine horizontal planes spaced at 24.6mm intervals within the model space, approximately spanning the active volume (Figure 10c). The matte white reverse side of the camera calibration target is used as a projection screen. We extract the 2D position of corners in the Gray-code patterns in the camera image with the same sub-pixel corner extraction algorithm used for camera calibration. Back-projection of the 2D camera location of the corner onto the known plane height above the table (Section 4.1) provides the necessary 3D coordinates. This process produces a set of 729 point correspondences for each projector. After calibration, we achieve a reprojection error of $0.4 \pm 0.3\text{mm}$ (1σ).

4.5 System Integration and Implementation Details

In our setup, we employ a Point Grey Flea2 1394b camera coupled to an 8mm $f/1.4$ lens to capture the images at a resolution of 1392×1032 pixels. The projectors are DLP-based Optoma EP727 units, with an image resolution of 1024×768 . We run the projectors on two dual-head video cards, and run a standard monitor for the user interface on a third card. The computer runs Linux on an Intel Core 2 Quad Q9450 processor.

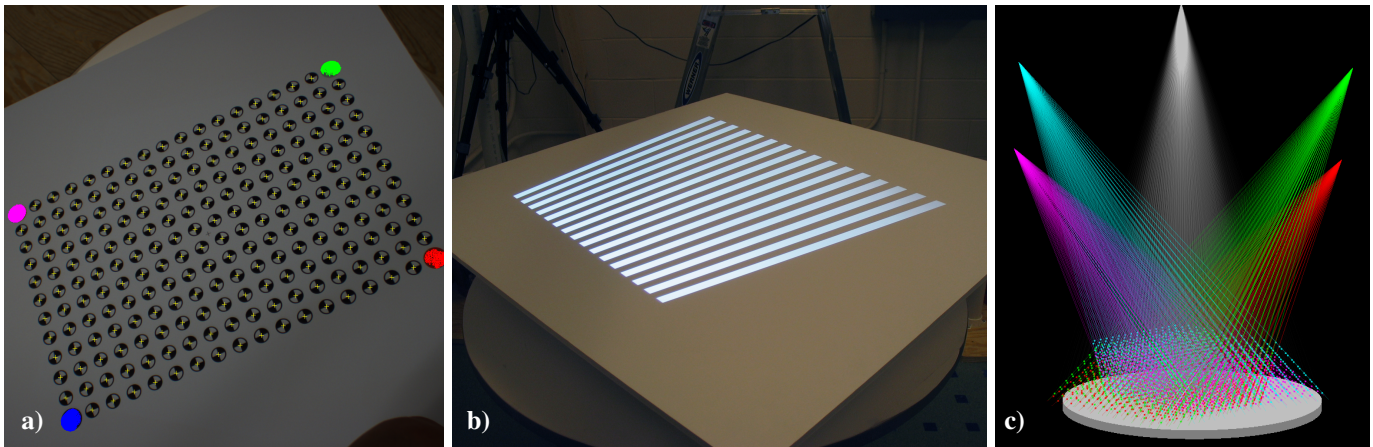


Fig. 10. Calibration targets for camera and projector. On the left, a processed image of the camera calibration target; on the right, a projector calibration image. Detected calibration points are marked with yellow crosses. Visualizing the active model volume. Rays are shown from the camera (white) and four projectors (red, green, cyan, magenta) projected into the modeling volume. Calibration points are drawn as small spheres.

5 RESULTS

The current system runs sufficiently fast for interactive daylighting visualization during iterative architectural design. Using unoptimized code, it takes approximately 12.5 seconds to generate the projection images for a model with 6 walls and 2 windows. Of this, approximately 0.05 seconds are used to process the captured image, and then additional 3.5 seconds are used to remesh the wall coordinates into a closed triangle mesh. After remeshing, it takes approximately 9 seconds to calculate form factors for the mesh with about 2000 triangles and compute an initial lighting solution. Because the most time consuming part of the system (remeshing and form factor calculations) need only be performed once for a particular wall geometry, additional lighting solutions for this geometry (changing the materials of walls and floors, time of day, day of the year, weather conditions, or site orientation) can be done in about 1 second. This facilitates interactive time-lapse animations of the daylighting solution. Figure 11 shows still frames from an animation of the daily variation in lighting conditions for a fixed model geometry and Figure 12 shows several images from an iterative design session.

Preliminary feedback from architecture students about the system has been positive. We plan to conduct formal user studies with our virtual heliodon to compare it with both the traditional heliodon and state-of-the-art daylighting analysis software packages. We believe our system will result in an improved understanding of the dynamic nature of daylighting and increased awareness of and attention to sustainable architecture design.

6 LIMITATIONS AND FUTURE WORK

Although our current system with four overhead projectors can be used to model many room configurations, tall, narrow room designs can result in areas occluded from all four projectors. Although this results in a presentation that does not reflect the true lighting solution, such shadow areas are

easily distinguished visually and we have not found them to be overly distracting.

Accurate representation of the full dynamic range of daylight is problematic. Projectors have a limited range of intensity: our four projectors together can produce a maximum 12,000 lux on the table surface; direct sunlight is $10\times$ brighter. Even the brightest projector commercially available (15,000 lumens [54]) projected onto a $3'\times 4'$ screen would result in only one ninth the brightness of direct sunlight. To compensate for the limited dynamic range we will extend tone mapping research [55] to tangible displays. As an alternative, we plan to explore false-color representations which can indicate overall lighting levels, or highlight areas that experience significant glare or inadequate illumination during some parts of the day.

While our current wall detection and tracking method has proven to be very robust, the algorithm may become less reliable as many more colors are used to code different wall heights or styles. Accordingly, we plan to explore other options for wall labels, such as: bar-code sequences on wall tops, a magnetic position tracking system [56], visible or infrared LEDs with unique colors or on/off sequences, and active systems where our projectors display structured light patterns captured with the camera or synchronized sensors that uniquely determine their 3D position [16]. Cost, robustness, ease-of-use, accuracy, speed, and how to minimize visual distractions for the designer will all be considered.

To correct for unintentional scattering of projected illumination between physical wall surfaces [11], we will perform inverse global illumination and color/intensity compensation [20] to determine the color that should be projected onto each wall so that the resulting intensity matches the software simulation. A solution to the general problem does not necessarily exist – we cannot emulate a scene with a white wall adjacent to a fully-absorptive black floor because we cannot prevent scattered light from striking and reflecting off the real-world floor. However, several characteristics of our application will allow us to compensate for most of

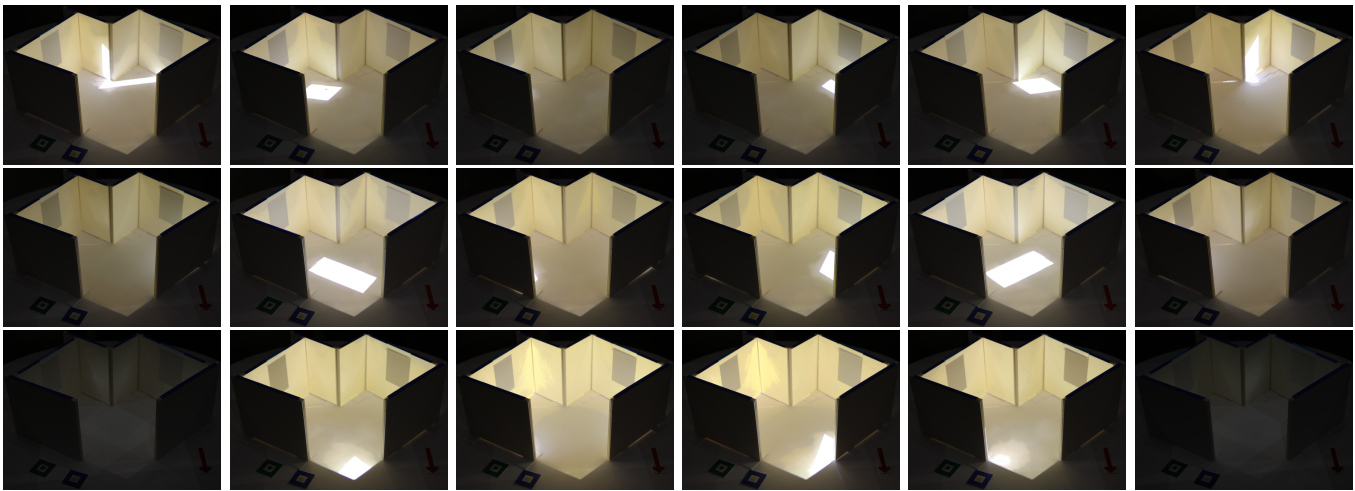


Fig. 11. Top row: lighting simulation for a room design for June 21 (summer solstice) at 2 hour intervals between 7AM and 5PM. Middle row: same simulation for March/September 21 (vernal & autumnal equinox). Bottom row: same simulation for December 21 (winter solstice).

the scattering: the diffuse surface assumption holds for most architectural scenes; neutral mid-range tones are used in most designs; gaps between the wall panels reduce surface inter-reflection; and the use of tone mapping will effectively reduce the desired contrast of the projections.

The mechanisms for user interaction can be further enhanced. We envision extending the system with physical control elements or gestures similar to the north arrow to adjust other aspects of the geometry or simulation; for example, a cloud shape could be used to indicate an overcast sky condition.

7 CONCLUSIONS

We have presented the *virtual heliodon*, a novel application of spatially augmented reality for architectural daylighting design. This small-scale, low-cost physical design environment is practical for academic architectural studios or professional design firms and suitable for installation in a multi-purpose space such as a conference room. Our system has a number of advantages over the traditional heliodon including ease of visualizing the interior lighting conditions and ease of editing the model geometry. This new design tool supports enhanced communication between client and architect and provides a platform for education in sustainable architectural design practice. The system can improve the architect's ability to make effective use of daylighting and reduce the need for supplemental electric lighting, reducing consumption of non-renewable energy resources. The same framework can be applied to other architectural design problems that require complex physical simulation and visualization, including passive solar heating and cooling, acoustics, aerodynamic building envelopes, and structural analysis.

ACKNOWLEDGMENTS

The authors wish to thank Marilyne Andersen from the Department of Architecture at MIT and Melissa Schroyer from the School of Architecture at RPI for their expertise in

daylighting and architectural design. We also thank Joshua Nasman for assisting with system calibration and the students in the computer vision and computer graphics research groups at RPI for their help with construction of the virtual heliodon. This work was supported in part by grants from IBM and NSF CMMI-0841319.

REFERENCES

- [1] E. Donoff, "Schermerhorn symphony center," *Architectural Lighting Magazine*, March 2007, <http://www.archlighting.com/industry-news.asp?articleID=460542§ionID=1331>.
- [2] C. Cruz-Neira, D. Sandin, T. DeFanti, R. Kenyon, and J. Hart, "The cave: Audio visual experience automatic virtual environment," *Communications of the ACM*, vol. 35, no. 6, pp. 65–72, June 1992.
- [3] O. Staadt, M. Gross, A. Kunz, and M. Meier, "The Blue-C: Integrating real humans into a networked immersive environment," in *Proc. ACM Collaborative Virtual Environments 2000*, 2000, pp. 201–202.
- [4] P. Dunston, J. McGlothlin, and L. Arns, "An immersive virtual reality mock-up for design review of hospital patient rooms," in *Conference on Construction Applications of Virtual Reality*, 2007.
- [5] C. Mackie, J. Cowden, D. Bowman, and W. Thabet, "Desktop and immersive tools for residential home design," in *Conference on Construction Applications of Virtual Reality*, 2004.
- [6] K. Dmitriev, T. Annen, G. Krawczyk, K. Myszkowski, and H. Seidel, "A cave system for interactive modeling of global illumination in car interior," in *ACM Symposium on Virtual Reality Software and Technology*, 2004, pp. 137–145.
- [7] O. Bimber and R. Raskar, *Spatial Augmented Reality: Merging Real and Virtual Worlds*. A K Peters, Ltd., 2005.
- [8] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs, "The office of the future: A unified approach to image-based modeling and spatially immersive displays," in *Proceedings of SIGGRAPH 98*, ser. Computer Graphics Proceedings, Annual Conference Series, Jul. 1998, pp. 179–188.
- [9] J. Underkoffler, B. Ullmer, and H. Ishii, "Emancipated pixels: Real-world graphics in the luminous room," in *ACM SIGGRAPH*, August 1999.
- [10] C. Pinhanez, "The everywhere displays projector: A device to create ubiquitous graphical interfaces," in *Ubiquitous Computing (Ubicomp)*, September 2001.
- [11] R. Raskar, G. Welch, K.-L. Low, and D. Bandyopadhyay, "Shader lamps: Animating real objects with image-based illumination," in *Rendering Techniques 2001: 12th Eurographics Workshop on Rendering*, Jun. 2001, pp. 89–102.

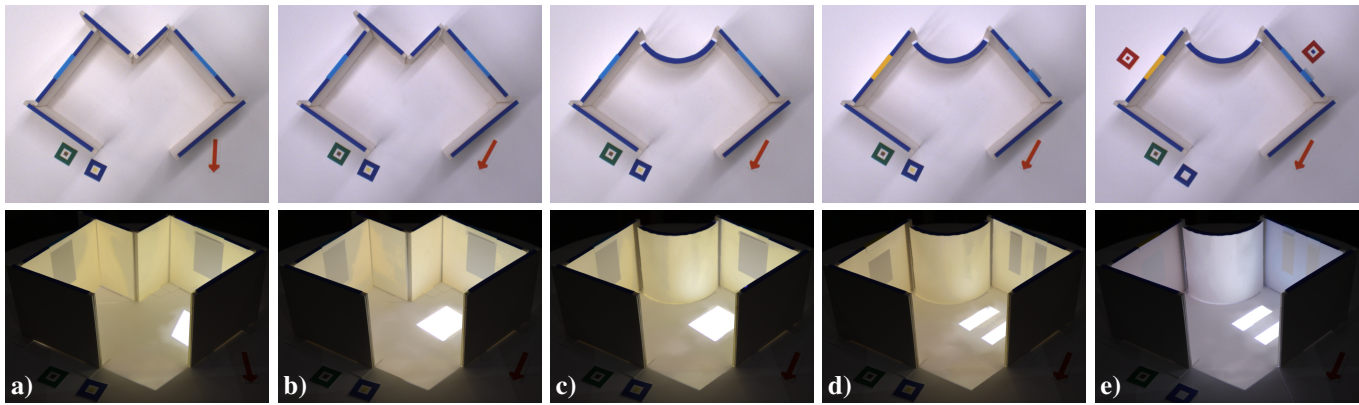


Fig. 12. Images from an interactive design session beginning with a) the initial design for a simple room configuration of 6 walls (also shown in Figure 11). b) The user adjusts the orientation of the entire building on the site by rotating the North indicator arrow. c) The geometry of the room is changed by replacing two planar walls with a curved wall module. d) The number and shape of windows within the design is changed by editing the corresponding markers. e) Finally, the reflectance properties of the surfaces are edited by changing the tokens that indicate these materials. All images in this figure show the daylighting solution for March 21st at 1pm.

- [12] R. Raskar, R. Ziegler, and T. Willwacher, "Cartoon dioramas in motion: Apparent motion effects on real objects with image-based illumination," in *International Symposium on Nonphotorealistic Animation and Rendering (NPAR)*, June 2002.
- [13] S. H. Han, J. H. Kim, T. S. Yun, and D. H. Lee, "Extensible interface using projector-based augmentation," in *International Conference on Computer Graphics & Virtual Reality (CGVR)*, 2006, pp. 132–137.
- [14] R. Raskar and P. Beardsley, "A self-correcting projector," in *2001 Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, Dec. 2001, pp. 504–508.
- [15] M. Ashdown, M. Flagg, R. Sukthankar, and J. M. Rehg, "A flexible projector-camera system for multi-planar displays," in *2004 Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, Jun. 2004, pp. 165–172.
- [16] J. C. Lee, P. H. Dietz, D. Maynes-aminzade, and S. E. Hudson, "Automatic projector calibration with embedded light sensors," in *ACM Symposium on User Interface Software and Technology (UIST)*, 2004.
- [17] R. Sukthankar, R. G. Stockton, and M. D. Mullin, "Smarter presentations: Exploiting homography in camera-projector systems," in *Proceedings of International Conference on Computer Vision*, 2001.
- [18] P. Beardsley, J. V. Baar, R. Raskar, and C. Forlines, "Interaction using a handheld projector," *IEEE Computer Graphics and Applications*, vol. 25, no. 1, pp. 39–43, January 2005.
- [19] S. Audet and J. R. Cooperstock, "Shadow removal in front projection environments using object tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.
- [20] A. Grundhofer and O. Bimber, "Real-time adaptive radiometric compensation," *IEEE Transactions Visualization and Computer Graphics*, vol. 14, no. 1, pp. 97–108, January-February 2008.
- [21] O. Bimber, A. Grundhofer, T. Zeidler, D. Danch, and P. Kapakos, "Compensating indirect scattering for immersive and semi-immersive projection displays," in *VR '06: Proceedings of the IEEE conference on Virtual Reality*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 151–158.
- [22] Y. Mukaigawa, T. Kakinuma, and Y. Ohta, "Analytical compensation of inter-reflection for pattern projection," in *VRST '06: Proceedings of the ACM symposium on Virtual reality software and technology*. New York, NY, USA: ACM, 2006, pp. 265–268.
- [23] A. Fournier, A. S. Gunawan, and C. Romanzin, "Common illumination between real and computer generated scenes," in *Graphics Interface*, 1993, pp. 254–263.
- [24] G. Drettakis, L. Robert, and S. Bounoux, "Interactive common illumination for computer augmented reality," in *Eurographics Rendering Workshop 1997*, Jun. 1997, pp. 45–56.
- [25] S. Gibson and A. Murta, "Interactive rendering with real-world illumination," in *Eurographics Workshop on Rendering (Rendering Techniques)*, June 2000.
- [26] O. Cossairt, S. Nayar, and R. Ramamoorthi, "Light-field transfer: Global illumination between real and synthetic objects," *ACM Transactions on Graphics*, vol. 27, no. 3, Jul. 2008.
- [27] CIE International Commission on Illumination, "Spatial Distribution of Daylight - Luminance Distributions of Various Reference Skies," 1994, technical Report CIE 110 - 1994.
- [28] C. M. Goral, K. E. Torrance, D. P. Greenberg, and B. Battaile, "Modelling the interaction of light between diffuse surfaces," in *Computer Graphics (Proceedings of SIGGRAPH 84)*, vol. 18(3), Jul. 1984, pp. 213–222.
- [29] G. J. Ward, F. M. Rubinstein, and R. D. Clear, "A ray tracing solution for diffuse interreflection," in *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM Press, 1988, pp. 85–92.
- [30] P.-P. Sloan, J. Kautz, and J. Snyder, "Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments," *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 527–536, Jul. 2002.
- [31] M. Nijasure, S. Pattanaik, and V. Goel, "Interactive global illumination in dynamic environments using commodity graphics hardware," in *PG '03: Proceedings of the 11th Pacific Conference on Computer Graphics and Applications*. Washington, DC, USA: IEEE Computer Society, 2003, p. 450.
- [32] P. Gautron, J. Krivánek, K. Bouatouch, and S. Pattanaik, "Radiance cache splatting: A gpu-friendly global illumination algorithm," in *Rendering Techniques 2005: 16th Eurographics Workshop on Rendering*, Jun. 2005, pp. 55–64.
- [33] C. Dachsbacher, M. Stamminger, G. Drettakis, and F. Durand, "Implicit visibility and antiradiance for interactive global illumination," *ACM Transactions on Graphics*, vol. 26, no. 3, pp. 61:1–61:10, Jul. 2007.
- [34] B. Cutler, Y. Sheng, S. Martin, D. Glaser, and M. Andersen, "Interactive selection of optimal fenestration materials for schematic architectural daylighting design," *Automation in Construction*, vol. 17, no. 7, pp. 809–823, September 2008.
- [35] T. Heidmann, "Real shadows, real time," *Iris Universe*, vol. 18, pp. 28–31, 1991, silicon Graphics, Inc.
- [36] C. Reinhart, D. Bourgeois, F. Dubrous, A. Laouadi, P. Lopez, and O. Stelescu, "Daylight 1-2-3 - A state-of-the-art daylighting/energy analysis software for initial design investigations," *Building Simulation*, 2007.
- [37] M. A. Lehar and L. R. Glicksman, "Rapid algorithm for modeling daylight distributions in office buildings," *Building and Environment*, vol. 42, pp. 2908–2919, 2007.
- [38] S. Bund and E. Y.-L. Do, "Spot! fetch light interactive navigable 3d visualization of direct sunlight," *Automation in Construction*, vol. 14, pp. 181–188, 2005.
- [39] Google, "SketchUp: 3D modeling software," 2008, <http://www.sketchup.com>.
- [40] M. Rea, Ed., *The IESNA Lighting handbook - Reference & Application*, 9th ed. Illuminating Engineering Society of North America, 2000.
- [41] G. W. Larson and R. Shakespeare, *Rendering with Radiance - The Art*

- and Science of Lighting Visualization*. Morgan Kaufmann Publishers, Inc., 1998.
- [42] Y. Sheng, T. C. Yap, C. Young, and B. Cutler, "Virtual heliodon: Spatially augmented reality for architectural daylighting design," in *Proceedings of IEEE Virtual Reality 2009*, 2009.
 - [43] B. Cutler, T. C. Yap, and Y. Sheng, "Inferring architectural designs from physical sketches: Application to daylighting analysis," in *Intelligent User Interfaces (IUI) Sketch Recognition Workshop*, February 2009.
 - [44] H. Kato and M. Billinghurst, "Marker tracking and hmd calibration for a video-based augmented reality conferencing system," in *Proceedings of the 2nd International Workshop on Augmented Reality (IWAR 99)*, San Francisco, USA, Oct. 1999.
 - [45] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.
 - [46] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
 - [47] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Mesh optimization," in *Proceedings of SIGGRAPH 93*, ser. Computer Graphics Proceedings, Annual Conference Series, Aug. 1993, pp. 19–26.
 - [48] F. Durand, G. Drettakis, and C. Puech, "Fast and accurate hierarchical radiosity using global visibility," *ACM Transactions on Graphics*, vol. 18, no. 2, pp. 128–170, Apr. 1999.
 - [49] Khronos Group, "The OpenGL Graphics System: A Specification," Aug 2008, <http://www.opengl.org/registry/doc/glspec30.20080811.pdf>.
 - [50] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1330–1334, 2000.
 - [51] J. Mallon and P. F. Whelan, "Which pattern? biasing aspects of planar calibration patterns and detection methods," *Pattern Recognition Letters*, vol. 28, no. 8, pp. 921–930, June 2007. [Online]. Available: <http://dx.doi.org/10.1016/j.patrec.2006.12.008>
 - [52] L. Lucchese and S. Mitra, "Using saddle points for subpixel feature detection in camera calibration targets," *Circuits and Systems, 2002. APCCAS '02. 2002 Asia-Pacific Conference on*, vol. 2, pp. 191–195 vol.2, 2002.
 - [53] R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 4, pp. 323–344, August 1987.
 - [54] Sanyo, "XGA Multimedia Projector PLC-XF47," 2007, <http://www.sanyo-lcdp.com/english/product/xf47/xf47.html>.
 - [55] G. W. Larson, H. Rushmeier, and C. Piatko, "A visibility matching tone reproduction operator for high dynamic range scenes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 3, no. 4, pp. 291–306, 1997.
 - [56] F. Raab, E. Blood, T. Steiner, and H. Jones, "Magnetic position and orientation tracking system," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. AES-15, no. 5, pp. 709–718, Sept. 1979.