Caustic Object Construction Based on Multiple Caustic Patterns

Budianto Tandianus
Nanyang Technological University, Singapore
budi0010@ntu.edu.sg

Henry Johan
Nanyang Technological University, Singapore
henryjohan@ntu.edu.sg

Hock Soon Seah
Nanyang Technological University, Singapore
ashsseah@ntu.edu.sg

ABSTRACT

Inverse caustic problem, that is computing the geometry of a reflector and/or refractor based on a given caustic pattern, is currently not widely studied. In this paper, we propose a technique to solve the inverse caustic problem in which we compute the geometry of a semi-transparent homogeneous refractive object (caustic object) given a directional light source and a set of caustic patterns (each pattern is considered to be formed at a specified distance from the caustic object). We validate the results by using mental ray (software rendering). The novelty of our research is that we consider a set of caustic patterns whereas existing techniques only consider one caustic pattern. We employ a stochastic approach to simulate the refracted light beam paths that can approximately reconstruct the input caustic patterns. Working backward, from the computed refracted light beam paths we compute the geometry of the caustic object that can produce such light beam paths. Due to having multiple caustic patterns as the inputs, it is a challenge to reconstruct the input caustic patterns because of the differences in their shapes and intensities. We solve this problem by using a two-step optimization algorithm in which we adjust the position and size of the caustic regions in the first step and we adjust the caustic shapes in the second step. Our technique is able to construct a caustic object for a various types of input caustic patterns.

Keywords: caustics, photon, reconstruction, inverse problem, stochastics

1. INTRODUCTION

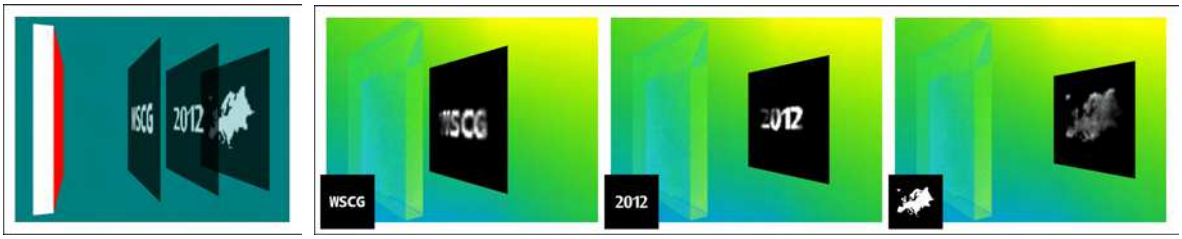
Recently, there is a growing interest in inverse problem research in Computer Graphics due to the possibility of controlling the creation of visual effects. By using the inverse techniques, the design process becomes easier as the artists can just specify the intended effects directly instead of performing the iterative trial-and-error process. However, inverse problem is generally difficult as in most cases there is no unique bijective relationship between the output and the input (i.e., given an output, there are many input possibilities that can generate such output).

In the inverse caustic problem, given an input caustic pattern (shape, intensity, and location from the caustic object) and a light source, we have to compute the geometry of the caustic object that can produce a caustic pattern similar to the input caustic pattern. Inverse caustic problem is hard to solve because the input caustic pattern only contains the irradiance magnitude and it does not have incident light direction information (and also the reflected and/or refracted light paths). Up to now, the inverse caustic problem is not widely studied and the existing work only consider a single input caustic pattern.

In this paper, we propose a new inverse caustic problem, that is computing the caustic object given multiple input caustic patterns formed on a caustic receiver (diffuse and non-transparent surface) at various distances from the caustic object. We show an example in Figure 1.

Our basic idea for solving this problem is as follows. We subdivide one side of the caustic object and also the caustic patterns into regular cells. The light beam refracted by each caustic object cell will pass through one caustic cell of each caustic pattern. We try to compute the orientation of each caustic object cell such that the combination of the refracted light beams of all caustic object cells can approximately reconstruct the input caustic patterns.

We use a stochastic approach in our technique and we represent each input caustic pattern as a 2D probability mass function (pmf) by considering the brightness of a caustic cell as the probability of a light beam might pass through it (i.e., the brighter the caustic cell is, the higher probability or the more likely a light beam is considered to pass through it). Hence, for each cell of the caustic object, we use the pmfs of the caustic patterns to determine to which direction the caustic object cell refracts a light beam. From the determined refracted light beam



(a) Scene setup. (b) Computed caustic object and approximately reconstructed caustic patterns.

Figure 1: (a) Scene setup. We compute a caustic object (the leftmost box), specifically the surface geometry of the side (shown in red color) facing the caustic patterns, given three caustic patterns (WSCG, 2012, and Europe) to be formed on a caustic receiver at three distances from the caustic object, with a directional light source orthogonal to the caustic object illuminates from the left. (b) mental ray renderings of caustics produced by our computed caustic object (final output). Input caustic patterns are shown in the insets at each image. The computational time is 9.0 hours.

direction, we can compute the orientation of the caustic object cell.

Due to differences of the input caustic patterns (in terms of shapes and intensities), it is hard to compute the caustic object that can satisfy all the input caustic patterns. Thus, we relax the input requirements by slightly adjusting the sizes, positions, and shapes of the non-zero intensity regions of the input caustic patterns. Moreover, we also allow a small amount of light beam which has passed through several caustic patterns to miss or overshoot the rest of the caustic patterns. We compute these adjustments by using optimization techniques. We validate our results by performing rendering simulation using mental ray [men12a], a robust industry standard rendering engine.

2. RELATED WORK

Unknown Input Given only the output, the input that can produce such output is computed. This is a hard problem since the *a priori* knowledge of the input is not available. One example is the work presented by Bottino et al. [Bot01a] and Mitra et al. [Mit09a]. They compute a 3D geometry that can satisfy the inputs which consist of a set of silhouettes or shadow patterns.

Inverse Caustics One of the earliest work in inverse caustic is presented by Patow and Pueyo [Pat04a]. They compute the reflector shape in an optical set (consists of a reflector, light source, and diffusor) given the radiance distribution as an input. They represent the reflector as grids and they iteratively adjust the grid vertices such as positions and number of vertices based on the similarity with the intended radiance distributions. The whole process took many days even though they can obtain radiance distribution similar to the input. They improve the work by allowing the user to set the range of the solution space [Pat07a] (the lower bound and the upper bound of the reflector shape). Hence, a user has more control in determining the reflector shape. They later increase the performance by using GPU [Mas09a] and they can reduce the processing time into magnitude of hours.

In parallel with the aforementioned work, Anson et al. [Ans08a] represent the reflector as a NURBS surface and Finckh et al. represent the reflector as a B-Spline surface [Fin10a]. As a result, during the optimization they optimize the control points instead of grid vertices which in the end can produce smooth reflectors in a relatively fast speed (due to the small number of parameters to be optimized). However, as Papas et al. also mention [Pap11a], the parameterized technique has a difficulty with highly complex caustic images, thus Finckh et. al [Fin10a] cannot reproduce all frequencies of the caustic pattern and Anson et. al [Ans08a] assume the shape of the caustic pattern to be circular.

Weyrich et al. generate a microgeometry reflector given a single reflected caustic pattern input [Wey09a]. The caustic object is subdivided into uniform cells (facets), and they compute the optimized orientation of each cell that can produce a caustic pattern similar to the input pattern. Papas et al. [Pap11a] improve the work of Weyrich et al. [Wey09a] by generating a refractor caustic object on a larger scale. Moreover, they are able to prevent noise on the reconstructed caustic pattern by computing the surface of each facet based on the Gaussian distribution. Similar to Weyrich et al. [Wey09a], they employ several optimization costs in order to generate the caustic objects. In the most recent development, Yue et al. [Yue12a] emphasize on modularity by reconstructing an input caustic pattern from a caustic object which consists of many smaller pieces of caustic object cells. Their caustic object cells are divided into ten types with each type refracts light to a predefined direction.

Comparison In all these work, the input is only a single caustic pattern. On the other hand, in this paper we propose a new challenge in which we compute the geometry of a caustic object based on a set of input caustic patterns.

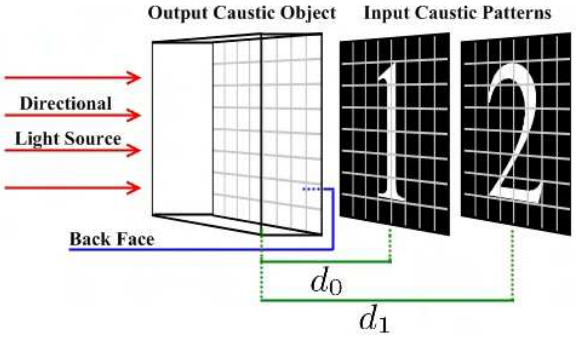


Figure 2: Scene setup. Our algorithm computes the normal/orientation of each caustic object cell. Caustic pattern '1' is formed when the caustic receiver is at distance d_0 from the caustic object and similarly caustic pattern '2' at d_1.

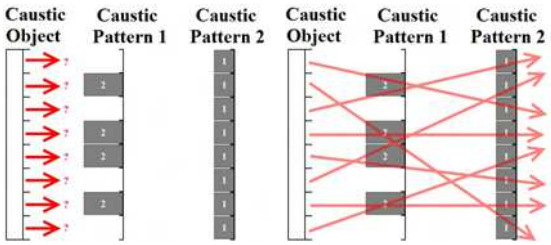


Figure 3: Problem formulation. (a) Given two caustic patterns at two different locations (with the intensity of each caustic cell is denoted by the size of the cell), compute light refraction direction (red arrow) of each caustic object cell such that the refracted light collectively can generate caustic patterns similar to the input caustic patterns. (b) Light refraction combination that can satisfy the input caustic patterns.

3. BASIC IDEA OF OUR METHOD

Scene Setup The scene setup is shown in Figure 2. The scene consists of three components, a caustic object (of a box shape), a caustic receiver (a planar surface where the caustic patterns are formed), and a directional light source (whose direction is orthogonal to the caustic object). Both the caustic receiver and the caustic object are positioned coplanar, with the caustic receiver is on one side of the caustic object (assumed to be the **back face** of the caustic object, facing (0, 0, -1) direction) and incoming light direction is on the other face of the caustic object (assumed to be the front face of the caustic object, facing (0, 0, 1) direction). We assume the caustic receiver and the caustic object to have the same spatial dimension (i.e. same width and height) and orientation. Therefore, the extent of the region of interest of each caustic pattern is bounded by the shape of the caustic receiver

The caustic patterns and the back face of the caustic object are subdivided into a regular grid of cells. Each cell of a caustic pattern stores the total caustic intensity on that particular cell. We call cells of caustic patterns which have non-zero caustic intensity as **caustic cells** and cells with zero caustic intensity as **empty cells**. The collection of caustic cells of a caustic pattern is collectively called as a **caustic region**. For each cell of the caustic object (**caustic object cell**), we compute its orientation such that it can produce a refracted light beam to a specific direction. In the rest of this paper, we refer to each refracted light beam as light and we represent each light beam in the following Figures 3 and 4 as an arrow.

Problem Formulation We compute the back face of a caustic object C given a set of p grayscale caustic patterns such that each caustic pattern j will be formed on the caustic receiver when the receiver is located at the user-input distance d_j from the caustic object. Specifically, we compute the orientation of each caustic object cell at the back face of the caustic object such that the cell refracts the incoming light into a direction that passes through parts of the caustic regions. Collectively, the light refracted from all caustic object cells is expected to pass through all the input caustic cells thus reconstructing the input caustic patterns. As mentioned in Section 1, the input caustic patterns only provide the estimate of the amount of refracted light arriving at caustic receiver cells, not the light directions. Hence, the main challenge is to compute refracted light paths that can approximately reconstruct all the given caustic patterns. This problem is illustrated in Figure 3. The orientation of each cell can then be determined based on the path of its refracted light.

Solution As explained above, the task is to compute refracted light direction combinations such that they can approximately reconstruct the input caustic patterns. In this case, more light is expected to pass through brighter caustic cells compared to darker caustic cells. Hence, to solve this, we simulate the direction of the refracted light of each caustic object cell by using a stochastic approach. The idea is to use the caustic intensity in each caustic cell as the probability that we will refract a light to that caustic cell (i.e. the brighter the input caustic pattern is, the more likely it is chosen as a refracted light target).

We represent the set of caustic patterns as a set of normalized 2D probability mass functions $\mathbf{P} = \{f_{P_0}, f_{P_1}, f_{P_2}, \ldots f_{P_{p-1}}\}$ (each in $\mathbf{P}$ is the pmf of the user-input caustic pattern on the caustic receiver when the receiver is located at the user-input distance from the caustic object). The pmf of each caustic pattern is defined by using the grayscale value (intensity) of the caustic pattern in which the probability at each caustic cell is the grayscale value of that particular cell in the caustic pattern. Each pmf f_{P_j} is normalized by

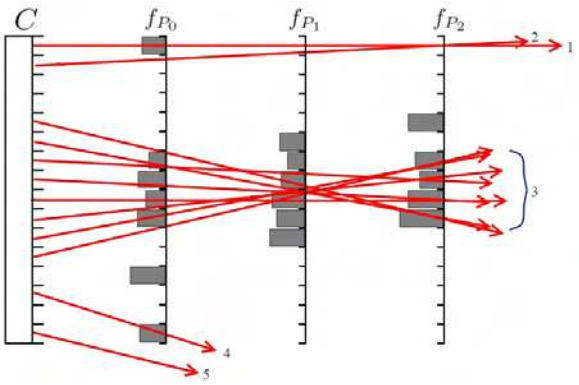


Figure 4: Each numbered arrow denotes the refracted light from the cells of the caustic object C and the gray blocks denote the probability of caustic cells. Light #1 and #2 have joint pmf of zero since their paths pass through at least one empty cell. Each light in #3 has the probability greater than zero since the light pass through non-zero cells. Light #4 is allowed even though it misses some of the caustic patterns. We will explain this further in Section 4.2. Light #5 is not valid because it does not intersect any caustic patterns

dividing the probability of each of its cell with the total probability of all cells of f_{P_j}.

We assign a random variable X_i for each i-th cell of the caustic object. For each X_i, the probability value of each possible refracted light direction $\mathbf{x}$ is computed by multiplying the probability of each caustic cell passed by the light refracted to direction $\mathbf{x}$ (see Figure 4), as shown in Equation 1.

$$f_{X_i}(\mathbf{x}) = \prod_{j=0}^{p-1} f_{P_j}(g_j^i(\mathbf{x})), \tag{1}$$

with $g_j^i(\mathbf{x})$ is a mapping function. The mapping function is basically a ray casting function in which the light is shot from the i-th caustic object cell to the caustic pattern j with the direction of $\mathbf{x}$ and return the intersected cell (of caustic pattern j). Then, for each caustic object cell or each X_i we assign a refracted light direction by using the Acceptance-Rejection method [vN51a] with the distribution based on the sampled joint probability mass function of all caustic patterns (Equation 1).

Once we obtain the refracted light direction for each caustic object cell, we compute its normal or orientation based on the user-input index of refraction of the caustic object, incoming light direction (orthogonal to the caustic object), and the obtained refracted light direction by solving the Snell's Equation (see Appendix A). Afterward, we perform rendering simulation using the mental ray to asses the approximate reconstructed caustic patterns (as shown in Figure 6a).

Note that our technique can also be applied to point light sources and directional light sources non-orthogonal to the caustic object. In the rest of the paper, the terms caustic patterns and pmfs are interchangeable as we use the term pmfs when we emphasize on the mathematical representation of the caustic patterns.

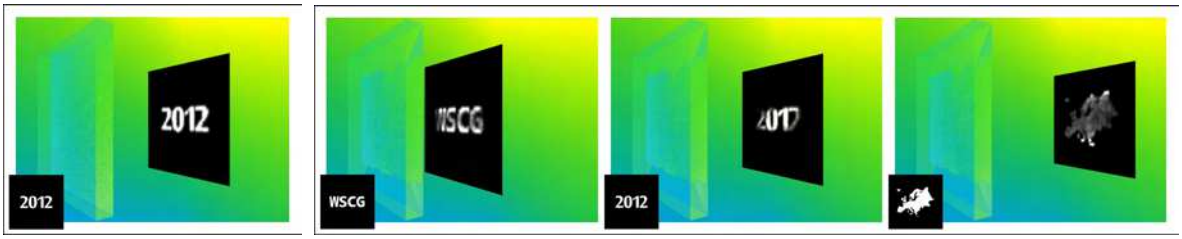
4. IMPROVING THE RECONSTRUCTED CAUSTICS

The solution in Section 3 may not be able to reconstruct the caustic patterns very well if the input consists of multiple patterns. If we only have a single caustic pattern (shown in Figure 5a), then we can reconstruct it very well. However, if we add two additional caustic patterns, then some parts of the input caustic patterns are missing (Figure 5b).

Reconstruction problem As explained in Section 3 (and shown in Figure 4), some refraction directions have zero joint pmf when they pass through at least one empty caustic cell. As a result, if all possible refraction directions from every caustic object cell pass through a caustic cell of a caustic pattern but they also pass through the empty cells of other caustic patterns, then the aforementioned caustic cell cannot be reconstructed (we call such cell as a **missing caustic cell**). As seen in Figure 4, the top and bottom caustic cells in f_{P_1} are missing caustic cells since the refracted light that pass through these caustic cells also pass through empty cells in the other caustic patterns.

Proposed solution Based on the given input caustic patterns and their configurations (positions and sizes), it might not be possible to compute the caustic object that can well reconstruct the original input caustic patterns. Thus, we propose a method to relax the input requirement by allowing slight changes to the positions, sizes, and shapes of the caustic regions. Our proposed method consists of two steps. In the first step, we optimize the size and position of the caustic regions by slightly adjusting the size and position given by the user (Section 4.1). In the second step, the boundaries of each caustic region are adaptively extended such that they enable the reconstruction of the missing caustic cells on the other caustic patterns (Section 4.2). We also compute the amount of light that is allowed to overshoot or to miss some caustic patterns.

In both optimization steps, we use Simulated Annealing [Kir83a]. The main reason we use Simulated Annealing is because the problem cannot be solved analytically. However, there are also some possible optimization techniques such as Particle Swarm, Ant Colony, and Genetic Algorithm. However, those techniques require keeping the record of multiple possible solutions at once, hence it is not efficient for our case (as seen in Equation 3, the cost computation requires reconstruction of the caustic patterns multiple times using the adjusted input caustic patterns). Moreover, in some of the related work [Wey09a, Fin10a, Pap11a], Simulated Annealing is also used. After applying our proposed



(a) (b)

Figure 5: (a) Only a single caustic pattern and it can be reconstructed very well. (b) Two additional caustic patterns cause some parts of the input caustic patterns to be missing.

solution, the reconstructed caustic patterns from the test case in Figure 5b are improved as seen in Figure 1b.

Cost computation In every iteration of both optimizations, we use the root mean square to compute the cost or degree of possibility that the adjusted input caustic pattern can be approximately reconstructed (in order to guide the simulated annealing). The root mean square is computed as the difference between the normalized reconstructed caustic patterns $\mathbf{Z} = \{f_{Z_0}, f_{Z_1}, \dots f_{Z_{p-1}}\}$ and the normalized adjusted input caustic patterns $\mathbf{D} = \{f_{D_0}, f_{D_1}, f_{D_2}, \dots f_{D_{p-1}}\}$. Only in the cost computation here, the normalization of caustic patterns in $\mathbf{Z}$ and $\mathbf{D}$ are computed by dividing the value of each caustic cell $\mathbf{t}$ with the maximum caustic cell value of the caustic pattern $\mathbf{t}$ belongs to. We compute the cost in this way such that the maximum cost (which is in the worst case scenario, for example the input caustic patterns are not reconstructed at all) is 1.0. The cost computation is shown in Equation 3.

$$Cost = \frac{1}{p}\sum_{j=0}^{p-1}\sqrt{\frac{1}{n(\mathbf{W}(j))}\sum_{i=0}^{\beta}(f_{D_j}(\mathbf{t}_i) - f_{Z_j}(\mathbf{t}_i))^2}, \quad (2)$$

with

$$\mathbf{W}(j) = \{\mathbf{t} | f_{D_j}(\mathbf{t}) + f_{Z_j}(\mathbf{t}) > 0\}, \quad (3)$$

and $\mathbf{t}$ is the caustic cell, p is the number of caustic patterns, $n(\mathbf{W}(j))$ is the number of elements of a set of caustic cells contributing to the cost computation, and β is the total number of cells of each caustic pattern (in our experiments, $\beta = 64 \times 64 = 4096$). For more accurate computation of $\mathbf{Z}$, we approximately reconstruct the caustic patterns 32 times and accumulate their caustic cell values, and finally we divide the value of each caustic cell by 32 in order to get $\mathbf{Z}$.

We reconstruct the caustic patterns by computing the refracted direction as explained in Section 3 and then for each caustic cell we accumulate the amount of refracted light that intersects it. We use the modified pmfs (which are adjusted in each optimization step) to compute the joint pmfs (Equation 1).

4.1. Adjusting the Size and Position

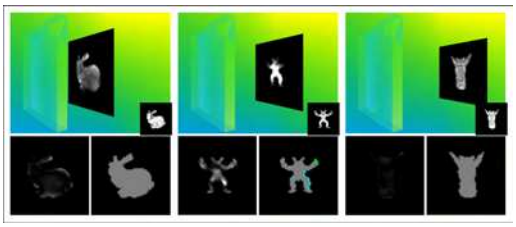
In this first optimization step, we relax the input caustic pattern configurations by iteratively adjusting the size and position of the input caustic region. Adjusting the position is basically translating the caustic regions in 3D space (translation in x, y, z). This means we also adjust the input distance (translation in z) between the caustic object and the caustic pattern (caustic receiver).

In every iteration, we adjust the size and positions of the input caustic regions and compute the cost by using Equation 2 in order to guide the optimization iterations. The adjusted input caustic patterns are used as the input to the next optimization step (Section 4.2) and they are also the target caustic patterns for every iteration of the second step.

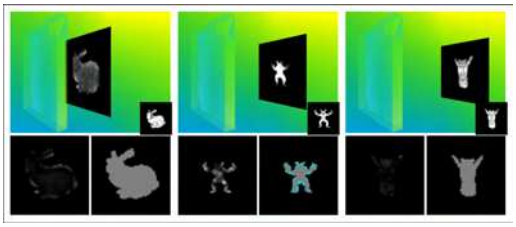
4.2. Extending Caustic Regions and Overshooting Refracted Light

After performing the first optimization step, there might be some missing caustic cells left. Missing caustic cells are the caustic cells that cannot be reconstructed. To reconstruct some of these missing caustic cells, we slightly extend the shape of all input caustic regions. For example, in Figure 4, the middle-top and middle-bottom caustic cells of f_{P_1} cannot be reconstructed since all possible refracted light that passes through these cells in f_{P_1} have to pass through empty cells in either f_{P_0} or f_{P_2}. Hence, to solve this, we can extend the middle caustic regions in f_{P_0} and f_{P_2} up and down by one cell.

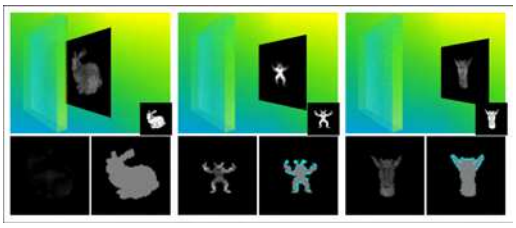
Some caustic cells especially around caustic patterns borders are hard to reconstruct as most light passing through these cells can miss other caustic patterns in behind. Thus, we relax this requirement by enabling some of the refracted light that passes through caustic cells of one or several caustic patterns to miss the rest of the caustic patterns (or region of interests of the rest of the caustic patterns). This is beneficial especially for the caustic cells on the border of the caustic patterns. For example, in Figure 4, if we enable light #4 to pass the bottommost caustic cell of f_{P_0} and miss the rest of caustic patterns (we call this **overshoot**), then



(a) Results without optimization. Cost : 4.45×10^{-1}



(b) Results after 1st optimization (Section 4.1). Cost : 4.08×10^{-1}

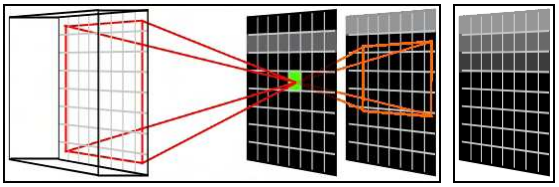


(c) Results after 1st and 2nd optimization (Section 4.2). Cost : 2.64×10^{-1}

Figure 6: Mental ray rendering results of the optimization steps. Input caustic patterns are shown at the bottom right of each screenshot in (a). We also show the missing caustic cell maps at the below right of each image (green cells show the missing caustic cells, gray cells show the caustic cells that can be reconstructed, and cyan cells show the extended caustic cells). For the visualization of the differences between the target and the reconstructed caustic patterns, we also show the caustic irradiance difference maps (assuming the total irradiance of each target caustic pattern is 1.0 and the total light emitted to the scene is 1.0, i.e. each caustic object cell refracts the light with the amount of 1.0 divided by the number of caustic object cells) at the below left of each image (from the darkest pixels with the least errors to the brightest pixels with the most errors). For the sake of visual clarity, we scale up the difference values by 5000. The computational time is 5.7 hours.

the bottommost caustic cell of f_{P_0} can be reconstructed. However, we still do not allow the refracted light of one caustic object cell to miss all of the caustic patterns (as in light #5).

Fully extending the caustic regions can deform the original caustics too much, and likewise if we allow too much light to overshoot the caustic patterns then the approximate reconstructed caustic patterns will have very low intensity. Hence, in this step, we apply an optimization to determine the appropriate caustic regions extensions amount $\mathbf{k} = \{k_0, k_1, \ldots k_{p-1}\}$ and light overshoot amount $\mathbf{o} = \{o_0, o_1, \ldots, o_{p-1}\}$ with $\mathbf{k}$ and $\mathbf{o} \in [0,1]$ (i.e., a k and an o value for each caustic pattern).



(a) (b)

Figure 7: (a) A simple example of missing caustic cell projection (is explained in Section 4.2). Gray cells are the caustic cells and the green cell is the missing caustic cell. (b) We extend the second caustic pattern (two cells away) with gradually decreasing intensity.

Extending Caustic Regions To enable the missing caustic cells of a caustic pattern j to be reconstructed, we have to firstly compute at most how many s_b unit cells away the caustic region boundaries of the other caustic patterns b ($0 \leq b \leq p-1, b \neq j$) have to be extended. Afterward, for every caustic pattern b, we extend its caustic region with the amount of $s_b \cdot k_b$. In order to enable smooth extension, we extend the caustic regions with linearly decreasing intensity (or probability value).

To do this, for every missing caustic cell of the caustic pattern j, we project it from every caustic object cell to the empty cells of other caustic patterns b ($0 \leq b \leq p-1, b \neq j$). We perform this projection for the missing caustic cells of all caustic patterns. This projection example is shown in Figure 7a in which we project the missing caustic cell (shown in green color). Afterward, for each caustic pattern b, we obtain the maximum distance (s_b) between its caustic region boundaries and its empty cells that receive the projections of the missing caustic cells (of other caustic patterns). In Figure 7a example, it is five cells ($s_b = 5$) away for the second caustic pattern and in Figure 7b the caustic region boundary is extended two cells away (if $k_b = 0.4$).

Some of the missing caustic cell projections might miss the other caustic patterns b. For example, if the missing caustic cell in Figure 7a is one or two cells to the right, then some of the projections will overshoot or will not hit the second caustic pattern. We use this information to control the possibility of the refracted light to overshoot each caustic pattern.

Overshooting Refracted Light To improve the results, we also enable the refracted light to overshoot some of the caustic patterns. Thus, during the missing caustic cells projections, we also compute the ratio (e_b) between the amount of these projections that do not hit caustic pattern b and the total amount of these projec-

tions toward caustic pattern b (i.e. sum of the missing caustic cell projections that hit and do not hit caustic pattern b). e_b is essential since it provides the information on the amount of probability that the refracted light miss plane b. Hence, the probability h_b that we will refract the light to miss the caustic pattern b is shown in Equation 4.

$$h_b = e_b \cdot o_b \cdot f_{P_b}(\mathbf{t}_{max}), \tag{4}$$

with o_b is the coefficient to control the probability of overshooting b-th caustic pattern and $\mathbf{t}_{max}$ is a cell of f_{P_b} with the highest probability value.

We show the optimization progression in Figure 6.

5. GEOMETRY CONSTRUCTION

As explained in Section 3 we use the joint pmf (Equation 1) of the caustic patterns to compute the refraction direction of each caustic object cell. From the refraction direction, we can obtain the normal of the particular caustic object cell. However, if we perform the optimizations in Section 4, then we use the modified pmfs (output from both optimization steps) to compute the joint pmf (and ultimately the normal of each caustic object cell).

Based on the computed orientation of each caustic object cell, we can obtain the caustic object geometry by computing the x, y, z coordinates of the four corners of each caustic object cell. The x, y coordinates of each caustic object cell corner can be easily found as the caustic object is uniformly subdivided. To compute the z coordinates of the caustic object cell corners, we firstly assume that the z coordinate of all caustic object cell middle points to be the same ($z = 0.0$). Next, we use the dot product operation, i.e. dot product between the normal of the caustic object cell and the vector from the caustic object cell middle point (we know its x, y coordinates from the uniform subdivision and also its z coordinate which is 0.0) to each caustic object cell corner (we know its x, y coordinates) must be equal to zero. In this case, the only unknown value is z of the caustic object cell corner.

Since the edges of the neighbouring caustic object cells do not have the same slope or the same z coordinate on both endpoints, there are vertical open spaces between caustic object cells (shaded with lighter gray in Figure 8), we generate additional polygons to close those gaps.

6. RESULTS

We present some results computed using our technique in Figures 1, 6, 10. In all of the test cases, the index of refraction of the caustic objects are 1.5, the resolution of the caustic objects are 128×128 and the resolution of the caustic receiver is 64×64. Resolution refers to the number of cells. If we assume the spatial size to be 1.0×1.0, then the size of each caustic object cell is $1.0/128 \times 1.0/128$ and the size of each caustic cell is $1.0/64 \times 1.0/64$.

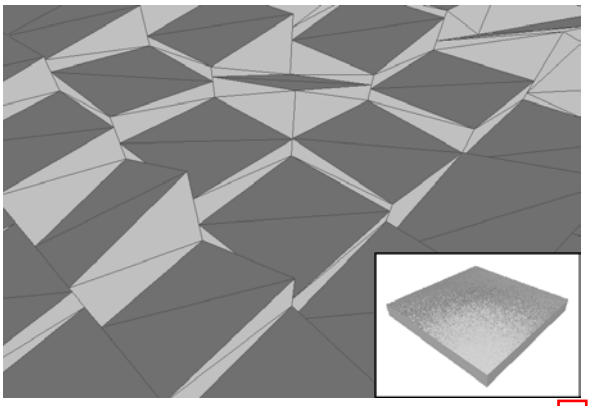
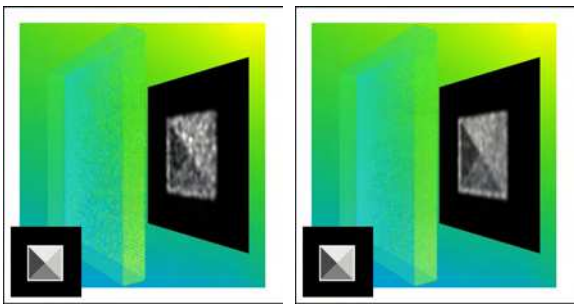


Figure 8: Caustic object geometry (inset) of Figure 6c with a zoom-in view. In the zoom-in view, each caustic object cell consists of two co-planar triangles shaded with darker gray. We also generate additional vertical polygons (shaded with lighter gray) to close the gaps between caustic object cells.

We use higher resolution for caustic object cells since we want to have more variations on the refracted light paths so that we can better reconstruct the contrast (or intensity variations) of the given caustic patterns. We show a difference example with a single caustic pattern case in Figure 9, a simple caustic pattern (resolution 64×64) reconstructed with a resolution 64 x 64 caustic object and a resolution 128 x 128 caustic object.



(a) (b)

Figure 9: A simple test case (one caustic pattern, with resolution 64×64) reconstructed with different caustic object resolutions. (a) Resolution 64 x 64 caustic object. (b) Resolution 128 x 128 caustic object.

We use the same parameters for the simulation annealing for both optimization steps in all experiments, i.e. 10 cycles of 10 iterations, Boltzmann's constant of 1.0, and temperature reduction factor of 0.5.

The experiments were performed on two comparable PCs. The specification of the first PC is Intel i7 920 2.67 GHz (CPU) with NVIDIA GeForce GTX 285

(GPU) and the specification of the second PC is Intel i7 880 3.07 GHz (CPU) with NVIDIA GeForce GT 330 (GPU). In the implementations, we calculate the joint pmf by rendering each caustic pattern and then we multiply them by using alpha blending (hence the use of GPU). For the rest of the computations such as Simulated Annealing and Acceptance-Rejection method, we perform them on CPU.

From the results, we can observe that the caustic objects generated using our technique can approximately reconstruct various types of input caustic patterns, especially for the WSCG (Figure 1b), fruits (Figure 10a), and rotating star (Figure 10c) test cases. The degree of difficulty in reconstructing the caustic patterns mostly depends on the number of caustic patterns, similarity between shapes, and the number of caustic cells in the input caustic patterns. As shown in Equation 4, due to the multiplication in computing the joint pmf, the probability of a particular refraction direction can become zero if the refracted light passes through empty cells. With the increasing number of caustic patterns especially the ones with different shapes, the chances that we have many refraction directions with zero probability also increase. We can see this from the results in Figures 1 and 6 (three input caustic patterns) where we can reconstruct better compared to the results in Figures 10 (four or more caustic patterns).

The shapes and orientations of caustic patterns can also affect the reconstruction difficulty. The test cases with similar caustic patterns, can be approximately reconstructed pretty well since similar refracted light paths are sufficient to reconstruct the caustic patterns. This is evident by comparing Figure 10a and Figure 10b. The caustic patterns in Figure 10a have near round shapes and approximately the same orientations. In contrast, the test case in Figure 10b have pretty different shapes (and orientations). Hence, there are few light that can pass through the endpoints of the bars compared to the middle regions of the bars.

Note the difficult test case shown in Figure 10b, four bar caustic patterns with alternating orientations. As we can see, the hardest parts to reconstruct are the ones near the two endpoints of the bars and on the other hand the parts around the centers are the easiest. This is due to the alternating shapes which cause the light paths to always pass through the center of the caustic regions. As we allow the refracted light to miss some of the caustic patterns, we are able to approximately reconstruct the top and bottom parts of the first caustic pattern. However, the consequence is that the last caustic pattern appears much dimmer.

In many cases, light tends to converge to the middle caustic patterns and as a result the center regions of these caustic patterns become relatively brighter compared to the other caustic patterns (for example, the Armadillo caustic pattern in Figure 6 exhibits this effect). This is due to two reasons. First, the caustic regions are positioned approximately at the center of the caustic patterns. Second, because the size of the caustic regions are mostly smaller than the caustic object. Therefore, some caustic object cells have to refract the light in the diagonal directions. This is illustrated in Figure 4 in which some of the light grouped to #3 have to be refracted in the diagonal directions.

Note that the relative depth of caustic patterns also affects the quality, as it is very difficult to reconstruct if the caustic patterns are located very near to each other. This is because the refracted light paths will intersect the patterns at similar locations and thus it is difficult to reconstruct caustic patterns with different shapes.

Please refer to the submitted **video** to see the progressive changes of the caustic patterns as the caustic receiver is moved.

7. APPLICATIONS

The inverse caustics has several potential applications.

Arts As shown in Figure 10, our technique can generate caustic objects that can produce several interesting caustic effects (similar to the intention in the inverse shadow [Mit09a]). Therefore, we hope that our work can encourage more exploration in caustic arts.

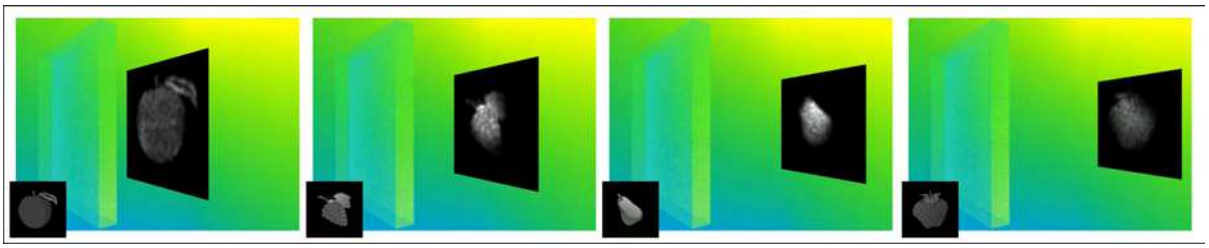
Information Encoding Information (such as serial numbers, passwords) can be encoded as caustic patterns of encrypted 2D images. Only when the requirements (such as light direction and caustic receiver distance) are known, we can recover the original information. We show an example in Figure 11 in which we encrypt **WSCG** and **2012** into two QR barcode patterns.

Validation Tests By using the computed caustic object, we can validate some processes such as rendering process (validating the correctness of caustics rendering algorithm) or manufacturing (validating the quality of produced glasses or light sources).

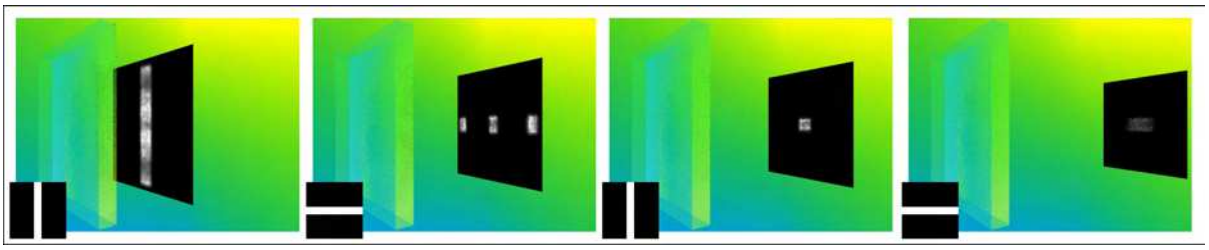
8. CONCLUSIONS AND FUTURE WORK

We have presented an inverse caustic problem and a novel technique which computes a caustic object given a set of caustic patterns with each pattern is positioned at a user-input distance from the caustic object. Our proposed technique is based on a stochastic approach, and it is augmented with two optimization steps that can alleviate the missing caustic problems. We have validated our results by performing physically rendering simulation using mental ray, and the caustic object generated using our technique can approximately reconstruct various types of input caustic patterns.

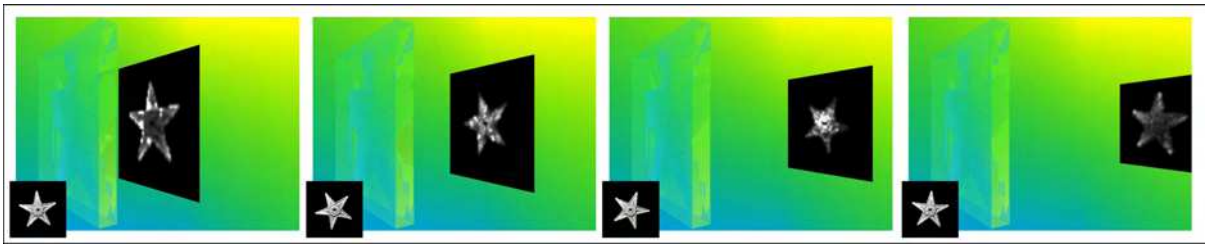
In the future, we would like to improve the quality of the reconstructed caustics in terms of smoothness. It



(a) Fruits (four caustic patterns). Computational time : 7.7 hours.



(b) Four bars (four caustic patterns). Computational time : 15.7 hours.



(c) Rotating Star (nine caustic patterns). Computational time : 27.6 hours.

Figure 10: More results. Note that the caustic pattern set in (c) contain similar patterns, as they are frames of a simple animation.

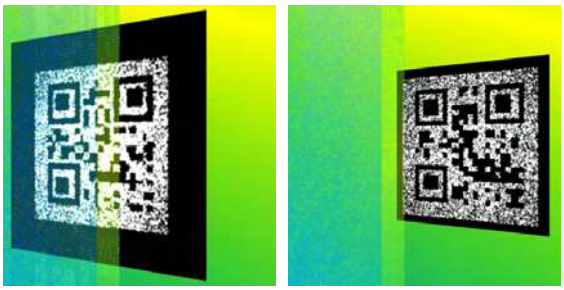


Figure 11: An application of information encoding. We encode **WSCG** and **2012** into two QR barcode patterns.

is also interesting to consider more complex light situations such as area light sources and dynamic light sources. It is challenging to use area light sources since they emit light to many directions from every point in the area light sources. As for the dynamic light sources, it is interesting to generate unique caustic pattern for each given light source direction (in this case, caustic object and caustic receiver are static). Last but not least, we would like to fabricate a real caustic object based on the computed geometry.

9. ACKNOWLEDGEMENTS

This work has been partially supported by the National Research Foundation grant, which is administered by the Media Development Authority Interactive Digital Media Programme Office, MDA (IDMPO). We would like to express our gratitude to Stanford Computer Graphics Laboratory for the 3D models (bunny, and armadillo) which are used to generate the input caustic patterns.

10. REFERENCES

[Ans08a] Anson, O., Seron, F.J., and Gutierrez, D.. NURBS-Based inverse reflector design. in CEIG08: Congreso Español de Informática Gráfica, Eurographics Association, Barcelona, Spain2008. pp. 65–74.

[Bot01a] Bottino, A., Cavallero, L., and Laurentini, A.. Interactive reconstruction of 3-D objects from silhouettes. in WSCG. 2001. pp. 230–236.

[Fin10a] Finckh, M., Dammertz, H., and Lensch, H.P.A.. Geometry construction from caustic images. in Proceedings of the 11th European Conference on Computer Vision: Part V, Springer-Verlag, Berlin, Heidelberg2010. ECCV'10. pp. 464–477.

[Kir83a] Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P.. Optimization by simulated annealing. Science. vol. 220, no. 4598, pp. 671–680, 1983.

[Mas09a] Mas, A., Martín, I., and Patow, G.. Fast inverse reflector design (FIRD). Computer Graphics Forum. vol. 28, no. 8, pp. 2046–2056, 2009.

[men12a] mental ray. http://www.mentalimages.com/products/mental-ray.html, 2012.

[Mit09a] Mitra, N.J., and Pauly, M.. Shadow art. ACM Trans. Graph.. vol. 28, pp. 156:1–156:7, 2009.

[Pap11a] Papas, M., Jarosz, W., Jakob, W., Rusinkiewicz, S., Matusik, W., and Weyrich, T.. Goal-based caustics. Computer Graphics Forum. vol. 30, no. 2, pp. 503–511, 2011.

[Pat04a] Patow, G., Pueyo, X., and Vinacua, A.. Reflector design from radiance distributions. World Scientific. vol. 10, no. 2, pp. 211–235, 2004.

[Pat07a] Patow, G., Pueyo, X., and Vinacua, A.. User-guided inverse reflector design. Computers & Graphics. vol. 31, no. 3, pp. 501–515, 2007.

[vN51a] von Neumann, J.. Various techniques used in connection with random digits. monte carlo methods. vol. 12, pp. 36–38, 1951.

[Wey09a] Weyrich, T., Peers, P., Matusik, W., and Rusinkiewicz, S.. Fabricating microgeometry for custom surface reflectance. ACM Trans. Graph.. vol. 28, pp. 32:1–32:6, 2009.

[Yue12a] Yue, Y., Iwasaki, K., Chen, B., Dobashi, Y., and Nishita, T.. Pixel art with refracted light by rearrangeable sticks. Computer Graphics Forum. vol. 31, no. 2, pp. 575–582, 2012.

A. NORMAL COMPUTATION

Given the Snell's Equation

$$\begin{aligned} \eta_1 \sin\theta_1 &= \eta_2 \sin\theta_2, \\ \eta\sqrt{1-\cos^2\theta_1} &= \sqrt{1-\cos^2\theta_2}, \\ \eta\sqrt{1-(-\mathbf{N}\cdot\mathbf{M})^2} &= \sqrt{1-(\mathbf{N}\cdot\mathbf{R})^2}, \\ A_1N_xN_x + A_2N_xN_y + A_3N_yN_y &= 1, \end{aligned} \tag{5}$$

with η_1 is the index of refraction of the caustic object, η_2 is the index of the refraction of air, $\mathbf{N}$ is the normal, $\mathbf{M}$ is the inverse incoming light direction, $\mathbf{R}$ is the refracted light direction, θ_1 is the angle between $\mathbf{M}$ and the inverse normal, θ_2 is the angle between $\mathbf{M}$ and the normal, and

$$\eta = \frac{\eta_1}{\eta_2} \qquad A_1 = \frac{\eta^2 M_x M_x - R_x R_x}{\eta^2 - 1}$$
$$A_2 = 2\frac{\eta^2 M_x M_y - R_x R_y}{\eta^2 - 1} \qquad A_3 = \frac{\eta^2 M_y M_y - R_y R_y}{\eta^2 - 1}$$

Since the normal vector $\mathbf{N}$ is normalized, the solution lies on a unit circle and as a result $N_x = \cos\phi$ and $N_y = \sin\phi$. Hence, Equation 5 can be simplified to

$$(2A_3 - 2)\tan^2\phi + 2A_2\tan\phi + (2A_1 - 2) = 0. \tag{6}$$

Equation 6 is basically a quadratic equation and the angle ϕ can be obtained by solving the quadratic equation.