

Rendering Issues in Pacioli's Rhombicuboctahedron

*Carlo H. Séquin
Raymond Shiau*



Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/Eecs-2015-169

<http://www.eecs.berkeley.edu/Pubs/TechRpts/2015/Eecs-2015-169.html>

June 24, 2015

Copyright © 2015, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Acknowledgement

This work is supported in parts by the Undergraduate Research Apprenticeship Program (URAP) at U.C. Berkeley.

Rendering Issues in Pacioli's Rhombicuboctahedron

Carlo H. Séquin and Raymond Shiao
EECS Computer Sciences, University of California, Berkeley
E-mail: sequin@cs.berkeley.edu

Abstract

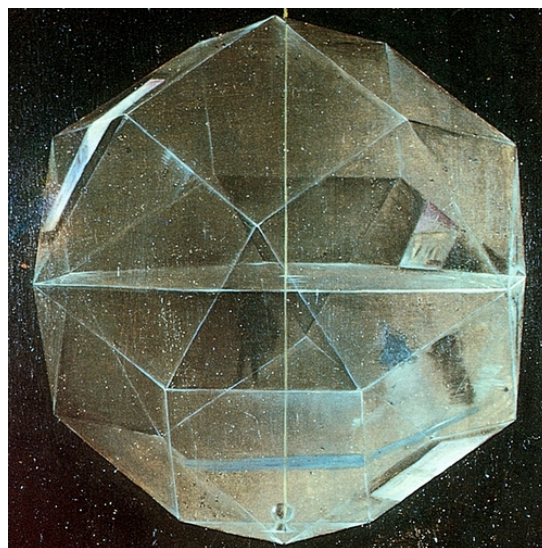
The depiction of the glass rhombicuboctahedron (RCO) appearing in a famous painting of Pacioli (1495) is analyzed as to how much it might agree with a physically correct rendering of a corresponding glass container half-filled with water. This investigation shows that it is unlikely that the painter of the RCO was looking at such a physical object. The question is then raised what a proper rendering of such an object might look like. Detailed computer renderings are presented that take into account multiple internal and external reflections and refractions. A warning is issued to non-experts in the use of computer graphics tools, showing that one cannot simply plug in the geometry of the RCO into a readily accessible rendering program and expect to obtain photo-realistic results. Care must be taken to model properly the interfaces where water is in contact with the glass container; the inappropriate handling of total internal reflections will give erroneous results. Detailed rendering studies on simpler glass and water geometries are recommended to assure that the resulting renderings do not deviate too much from physical reality. Nevertheless, the painter of the RCO has clearly succeeded in providing a rendering that appears very plausible to almost all observers.

1. Introduction

The rhombicuboctahedron (RCO), which is the focus of our attention in this article, appears in the painting “Ritratto di Fra’ Luca Pacioli” (1495) exhibited in the Museo e Gallerie di Capodimonte [15] in Naples, Italy. The central person (Figure 1a) is Fra’ Luca Pacioli, a famous mathematician of the Renaissance period, most likely lecturing on some topic concerning the regular Platonic solids or the semi-regular Archimedean polyhedra. There is some speculation that the second person in the painting might be Albrecht Dürer who also had an interest in symmetrical polyhedral objects. The great fascination with such objects at that time eventually culminated with “De Divina Proportione” written by Pacioli around 1497. This book on mathematical and artistic proportions was illustrated by Leonardo da Vinci.



(a)



(b)

Figure 1: (a) Pacioli painting; (b) Enlarged and enhanced view of the rhombicuboctahedron (RCO).

The suspended RCO, shown in the top left of the painting, is one of the thirteen semi-regular Archimedean solids. Its surface is composed of 18 squares and 8 equilateral triangles, and each vertex is shared by three squares and one triangle. The painting implies that this object has been realized with 26 glass plates fused together well enough, so that this shell can be filled with some liquid up to its centroid.

Mackinnon suggests [13][12] that the RCO, with its square and triangular faces, represents four of the five regular Platonic solids, and by filling it partially with water it might evoke associations with the four corresponding elements according to the chemical theory of Plato's *Timaeus*: *Earth* (cube) by the physical glass shell, *Water* (icosahedron) and *Air* (octahedron) by the media contained in this shell, and *Fire* (tetrahedron) by the depicted bright reflections. The fifth regular solid, the dodecahedron, which for Plato represented the *Universe*, is shown in a model on the right hand side of the painting.

It is now generally assumed that the painting was done by Jacopo de' Barbari. However, there is some speculation that the depiction of the transparent RCO was added by some other artist, since in style and detail it looks quite different from the other objects in this room. Many admirers of this painting have issued glowing comments about how wonderfully the painter seems to have captured the reflections and refractions in this object (Figure 1b). Mackinnon [13] attributes this part of the painting to Leonardo da Vinci himself. Even though Pacioli had not yet met Leonardo when this panel was painted, and he started to collaborate with him only after 1495, there is a possibility that the RCO was added later to the portrait of Pacioli when they were both in Milan [12]. More up-to-date references and in-depth discussions of this painting can be found in [3], [6], [9], and [19].

Clearly the RCO in this painting stands out in a special way. George Hart on his page devoted to Luca Pacioli's Polyhedra states [11]:

The polyhedron in the painting is a masterpiece of reflection, refraction, and perspective. (Davis states that the bright region on its surface reflects a view out an open window, showing the Palazzo Ducale in Urbino.) Certainly an actual glass polyhedron was used as a model. (Pacioli states in his books that he constructed several sets of glass polyhedra, but I know of no other information about them.) The polyhedron in the painting is beautifully positioned, suspended with a 3-fold axis vertical, out of physical contact with the other objects in the scene. I suspect that Pacioli chose it for the portrait because he discovered this form and was quite proud of it. (Presumably Archimedes first discovered it, but that wasn't known in Pacioli's time.) The painting is the earliest known image of the rhombicuboctahedron.

But on a closer, more critical inspection some things seem not quite right: The reflection of the window (not seen in the painting) on the upper left of the RCO seems more like a pasted-on sticker image that bends around one of the polyhedron edges, rather than like two separate reflections in the two differently angled RCO facets. Already in 2007, Joost Rekveld raised doubts in a publication [17] and on his website [18]:

Last week in Napoli I revisited the Capodimonte museum and its amazing collection of paintings. At some point I found myself face to face with this canvas, attributed to Jacobo de Barbari, a portrait of the mathematician Luca Pacioli painted in 1495.

... In said painting I suddenly noticed the mysterious reflections in the gorgeous mathematical shape at the top left, a rhombicuboctahedron, made of glass and half filled with water. I took a picture, and when I zoom in on these painted reflections we see buildings and sky, as if to suggest that an open window in the room is being reflected in the glass facets. The direction doesn't seem right though; am I wrong or do the reflections show that the window is high up towards the left? Or rather towards the bottom right? Both do not really make sense. Also in either case the light in the painting does not seem to be much affected by the source of these reflections.

... To me the way the same window seems to be reflected three times within this shape doesn't seem terribly true to the laws of optics. Or more precisely: they seem true to a textbook notion

of reflection and refraction of light in water, but the way the images are formed doesn't seem very realistic at all.

... Its depiction doesn't seem based too much on observation or the tracing of reality through a camera obscura, but then again, I've never actually seen a glass rhombicuboctahedron half filled with water, so who am I to judge?

Thus it seems worthwhile to investigate these issues. Just as disturbing as the “pasted-on” reflection of the window discussed above is the fact that there are no visible effects of refraction in the water body in the lower half! The RCO edges on the back surface appear in the painting in exactly the places where the computer rendering of a thin wire-frame object (Figure 2b) shows them.

In May 2011, Claude Boehringer [5], an artist working with various materials, produced a physical glass model of an RCO shell held together with lead, which was strong enough to be half filled with water. Under the guidance of Herman Serras [21] the model was suspended in the proper way to obtain the same orientation as the polyhedron in the painting. Serras then took the photograph shown in Figure 2a. This image looks quite different from the one in the painting, and so it is difficult to draw conclusions about the realism of the depiction of 1495. The surroundings where this model was photographed are entirely different, and there are no dominant reflections of a single, brightly lit window. Nevertheless, the photo of the Boehringer model (Figure 2a) confirms our intuition that those edges seen through the water body would be seriously altered in their rendered positions.

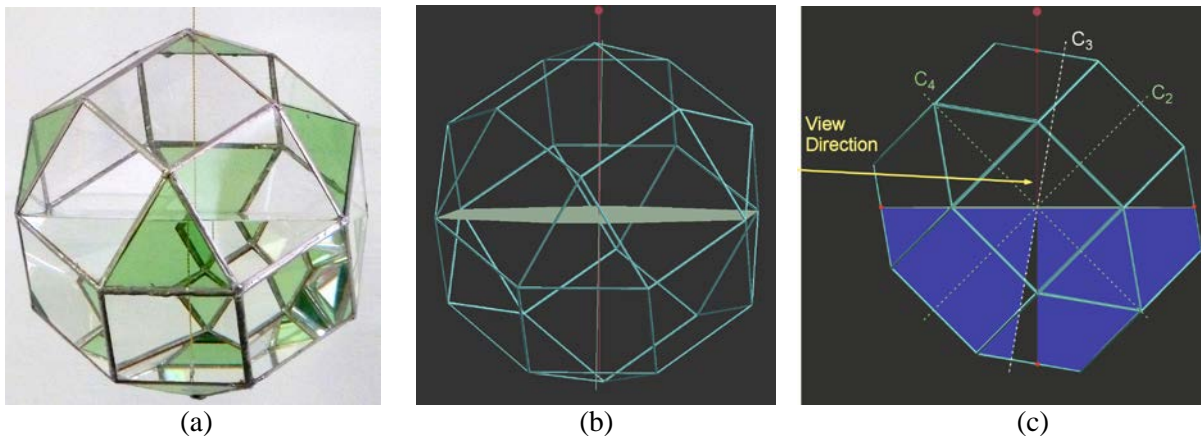


Figure 2: (a) Model by Claude Boehringer photographed by Herman Serras; (b,c) viewing geometry.

In the discussions prompted by the website [20], the question arose what an actual RCO glass shell half-filled with water really would look like, and several people have spent some time to use computer graphics programs to model such an object with its various reflections and refractive effects. However, different people came up with quite different results, driving home the point that modeling a complex object with multiple volumes with different refractive indices abutting against one another is not a trivial task. When making a model of the object to be rendered, the idiosyncrasies of the rendering program to be used need to be taken into account carefully. One model does not fit all possible renderers! Even for people experienced in using computer graphics rendering programs, it is advisable to run through a series of simple, and then progressively more complex test models, when switching to a new renderer.

In most of our efforts we have used Autodesk Maya [2] as our modeling tool and Mental Ray [14] as our rendering engine using a basic ray-tracing algorithm [10],[16]. First we constructed simple geometrical test models for which the correctness of the renderings could readily be verified, and made sure that the renderer properly interpreted the desired geometry in the various refraction events and reflections at external and internal boundaries. In the end we repeated this process once more for the open-source rendering program Blender [4]. Overall, these computer simulations confirm that it is highly

unlikely that the painter was observing a physical glass container half-filled with a clear liquid when painting the RCO. Our computer simulations look quite different from the painted RCO, proving that the latter is not a physically correct rendering. We thus conclude that the artist's most likely objective was to create the most "plausible" and "convincing" depiction of such an object for a broader public.

2. Viewing Geometry

To create a realistic rendering that can be compared with the painted RCO, we first have to figure out how this RCO has been suspended, and where to place the observer's eye. If the RCO were indeed suspended along one of its 3-fold symmetry axes piercing the centers of the top and bottom triangles, as stated in [11], then these two faces would be truly horizontal. In the painting we see both of these triangles from below, which then would imply that the eye of the observer must lie below the bottom of the RCO. On the other hand, the water surface clearly is seen from above: The more strongly depicted edges of the RCO are belonging to its front facets, and the foreshortened left and right edges of the polygon depicting the water surface have a vanishing point that lies somewhere in the upper right. This view geometry requires an eye point above the center of the RCO.

Careful inspection of the enlarged view of the painted RCO (Figure 1b) shows that the water surface passes through 4 of the 24 vertices of the RCO and cuts the vertical, square facets on the left and right sides along their horizontal diagonals (Figures 2b, 2c). This implies that the suspension line must form an angle of 45° with the plane of the square immediately in the back of the top triangle, as well as with the square in front of the top of the RCO. The C_3 symmetry axis deviates by $54.74^\circ - 45^\circ = 9.74^\circ$ from this suspension line. A few trigonometric calculations then reveal that this suspension line cuts the height line of the top triangle in the ratio $1:\sqrt{2}$, i.e., a fraction of 0.4142 away from the top vertex in the painting. Herman Serras [21] also had calculated the distance of the suspension line penetration from the closest triangle vertex as $\sqrt{3} * (\sqrt{2}-1)/2 = 0.3587$ times the RCO edge length. Fortunately these two calculations agree. In addition, the water boundary intersects two more pairs of triangle edges at a fraction of $\sqrt{2}-1 = 0.4142$ away from the shared vertex (red dots in Figure 2c).

With the angle of suspension unambiguously resolved based on the boundary of the water level, we now can try to locate the eye point of the observer. The viewer must be looking towards the center of the RCO with a slight downward angle that must lie between 0° and 9.74° . The best match between an appropriately tilted computer model and the rendering in the painting indicates a downward angle of about 3° (Figure 2c). Comparing the resulting view of a wire-frame RCO (Figure 2b) with the polygon edges depicted in the painting (Figure 1b) confirms that we have found the proper viewing geometry.

Now let's review this in the context of the complete painting. First we try to find the viewer's eye level with respect to the painting. Our intuition tells us that the observer's eyes are about at the height of Pacioli's nose or eyes, and that the person on the right is looking slightly down on the viewer. This would place the view center properly above the water level. We can also try to find a horizon compatible with the objects on the table (assumed to be horizontal), and this is what we found:

- 1.) a vanishing point on top of Pacioli's head for the slate tablet;
- 2.) a vanishing point somewhat above the middle of the RCO (but way out to the left) for the open book (with substantial error margins);
- 3.) a vanishing point slightly above the top of Pacioli's head for the red box (also with substantial error margins).

All of these vanishing points are higher than the center of the RCO, and this is compatible with our view of the water surface from above. But then, the fact that we can see the lower side of the bottom triangle implies that it must be slanted upwards towards the viewer. This confirms the tilted suspension of the RCO established above.

Now let's take a closer look at the projection used in the painting: Clearly there is some perspective involved; the back-face triangle (point down) is a few percent smaller than the size of the front triangle

(point up). The same interactive computer rendering program that lets us find the best match for the view angles also lets us find the parameters for the perspective projection in which all the rendered edge crossings of the wire frame model best match the locations in the painting; this occurs when the eye is about 14 RCO diameters away from its center. Now, based on its position and comparing it to the hands and head of Pacioli, we may estimate that the RCO is about 8 to 10 inches in diameter. Thus, based on its own perspective, it must have been drawn as it would look from about 10-11 feet away. But Pacioli in the painting seems only about 5 to 6 feet away from the viewer, and the object, if it is indeed above the table, would then be only 4 to 5 feet away. Thus the perspective of the rhombicuboctahedron is not compatible with the rest of the scene. The RCO should exhibit much stronger foreshortening, or it should have been depicted at only about half its current scale. This raises a strong suspicion that this RCO was sketched out quite carefully, but separately, – perhaps from a smaller model or from a geometrical construction – before it was copied into the Pacioli painting.

3. Detailed Computer Rendering of the RCO

So let's assume that the RCO was painted in a separate sitting, possibly in a room with an open window in the left wall, offering a view of a palazzo and some bright sky, and perhaps with some more local illumination coming from the lower right. Can we find an environment and some suitable material constants and illumination levels that will produce an image closely resembling the depiction in the painting? What would a physically more realistic rendering of such a glass container half-filled with water look like?

The first task is to create some geometrical model of the object to be rendered. To have some control over the appearance of the edges of the glass container, we flesh out the wire-frame shown in Figure 2b into properly mitered prismatic beams that are composed into a polyhedral object of genus 25 (i.e., a shell with 26 facet openings). Figure 3(a) shows an enlarged partial view of this framework; to make its geometry more clearly visible, all the cross sections have been enlarged by a factor of 2. In our model, the thickness of these struts is parameterized and set equal to the thickness of the glass plates shown in green in Figure 3(b); this allows us to emphasize or de-emphasize the visibility of these seams. In most of our renderings, we have set the material of the struts to be some gray, diffusely reflective material. However, there is the option to assign them the same material parameters as are used for the glass plates, in order to simulate a completely fused contiguous glass container. Finally, the water volume is modeled as a separate polyhedron that fits snugly against the inside surfaces of the glass plates in the lower half of the RCO (Figure 3c).

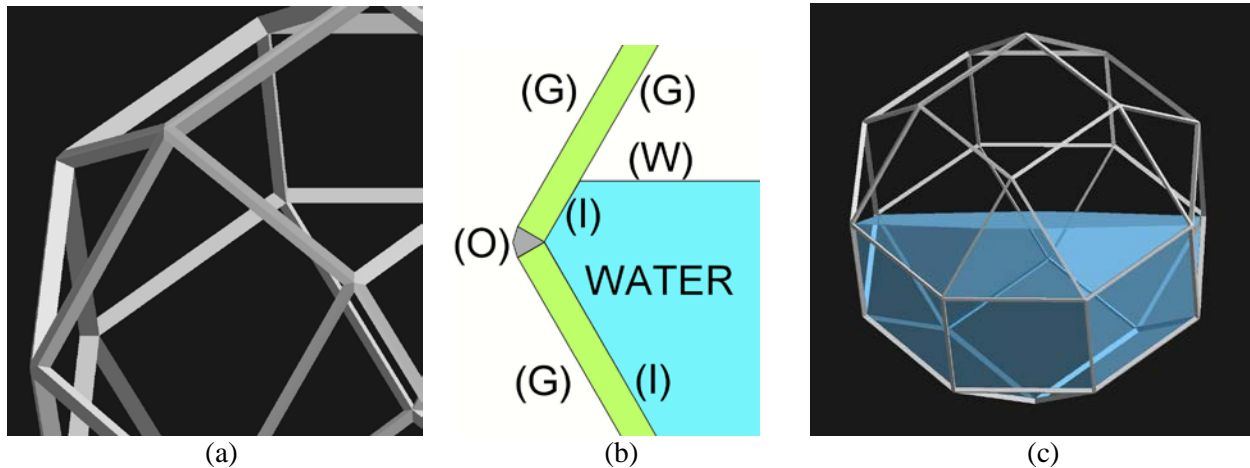


Figure 3: Modeling the complete RCO: (a) Filler framework; (b) cross section showing the types of interfaces; (c) a transparent, non-refractive water polyhedron inside the opaque filler framework.

One might now expect that simply sending this geometry to some free on-line rendering program (i.e., Blender [4]) or to a more sophisticated commercial computer graphics tool (i.e., Autodesk Maya [2]) would yield a photo-realistic rendering of this object in whatever environment this model is placed in. However, several attempts by Berkeley students as well as remote collaborators have produced quite different-looking results, and it became clear that producing physically and optically correct renderings is far from trivial. Some efforts led to fairly convincing results concerning the reflections, but left the refraction calculations incomplete. Others showed reflections as well as refractions, but it was unclear whether they all appeared in the right places and with the right intensities. Therefore, in our work reported here, we took a very careful incremental approach to model each physical effect first with extremely simple geometries and in an environment that made it possible to trace every reflected or refracted feature through the test object back to a location in the surroundings. Thereby we could maintain confidence that the obtained results were indeed indicative of physical reality.

To obtain photo-realistic renderings, a good technique is to use ray-tracing [10],[16]. In this approach geometrical rays are traced from the eye (or camera) through each pixel of the display screen into the virtual scene represented in the computer, with the goal to find out what each ray hits at the end of its journey, so that then each pixel can be displayed with the corresponding hue and shading. When rendering a scene with transparent media and partially reflecting surfaces, the situation becomes more difficult. As the probing, geometrical ray hits such a surface, it has to be split into two components: one that passes into the transparent medium and gathers possible light energy that may have come this way, and a second ray that leaves the intersected surface at the proper angle of reflection to collect the light contributions from this other direction. These various ray components may again intersect surfaces of the same kind and may have to be split again. Overall, we have to keep track of a whole tree-structured network of rays zig-zag-ing through the scene in order to determine the proper composite color that needs to be assigned to one particular pixel. This article is not the place to give a tutorial on ray-tracing in general. However, we have observed that the typical, non-expert user of commercial or freely available rendering tools may fall into the trap of naively trusting the first results they obtain when they send the proper detailed geometry and materials constants to such a rendering program. In the following we will outline an approach that a careful user should take to establish confidence that the observed results do not deviate too much from physical ground truth.

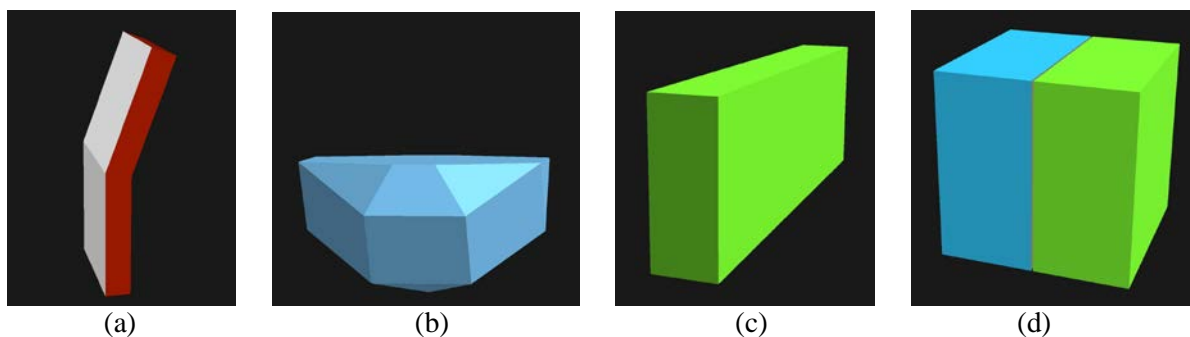


Figure 4: *Test objects used in preparatory rendering experiments: (a) angled mirrors (Figure 5); (b) RCO water body (Figure 6); (c) glass brick (Figure 7); (d) water/glass half-cubes (Figure 8).*

In most of our efforts we have used Autodesk Maya [2] as our modeling tool and Mental Ray [14] as our rendering engine. A great deal of effort was devoted to understanding and verifying the correct behavior of these tools to the extent we needed them for our studies. The photo-realistic renderings were produced in a ray-tracing mode, where we could control the number of consecutive photon/matter interactions for all the surfaces intersected by all the rays cast. Since the water-filled glass RCO is a complex object in which multiple reflection/refraction events occur between different interfaces, we first rendered simple prisms and thick glass plates (Figure 4) immersed in a “Cornell Box” [7] with distinct, high-contrast textures on its inside walls. For these test objects, we started out with simple ray-casting, having disabled

all secondary ray branches in the above-mentioned ray-tracing tree. Then we gradually increased the number of allowable reflections and/or refractions and watched extra features appear in the rendered image. This incremental approach allowed us to identify each newly appearing feature by its sequence of refractions and internal or external reflections. We also built a 2-dimensional simulator program that allowed us to see how different light rays travel through our test-setup and are split into different reflected and refracted components (Figure 9). In Section 4 we describe some of these preliminary tests; and in Section 5 we then show this incremental approach applied to the complete RCO geometry. In this article we can only show a small fraction of all the images produced, but we hope that they are sufficient to describe the approach taken.

4. Preliminary Rendering Tests

We started by studying the reflections and refractions occurring in simple rectangular prisms (thick glass plates). Our techniques employed to understand the rendered images included: Limiting the depth of the tree of reflection/refraction events, comparing side-by-side two images that differ in only one parameter; gradually and interactively varying the angle under which one of these simple objects appeared to the observer; and creating rich, textured backdrops (environments) to determine the final landing spots of the rays traced from the eye point. In this context, the Mental Ray documentation [14] proved to be very useful for understanding how rays move through different interfaces and for controlling the behavior of refracted/reflected rays up to some stated recursive depth limit.

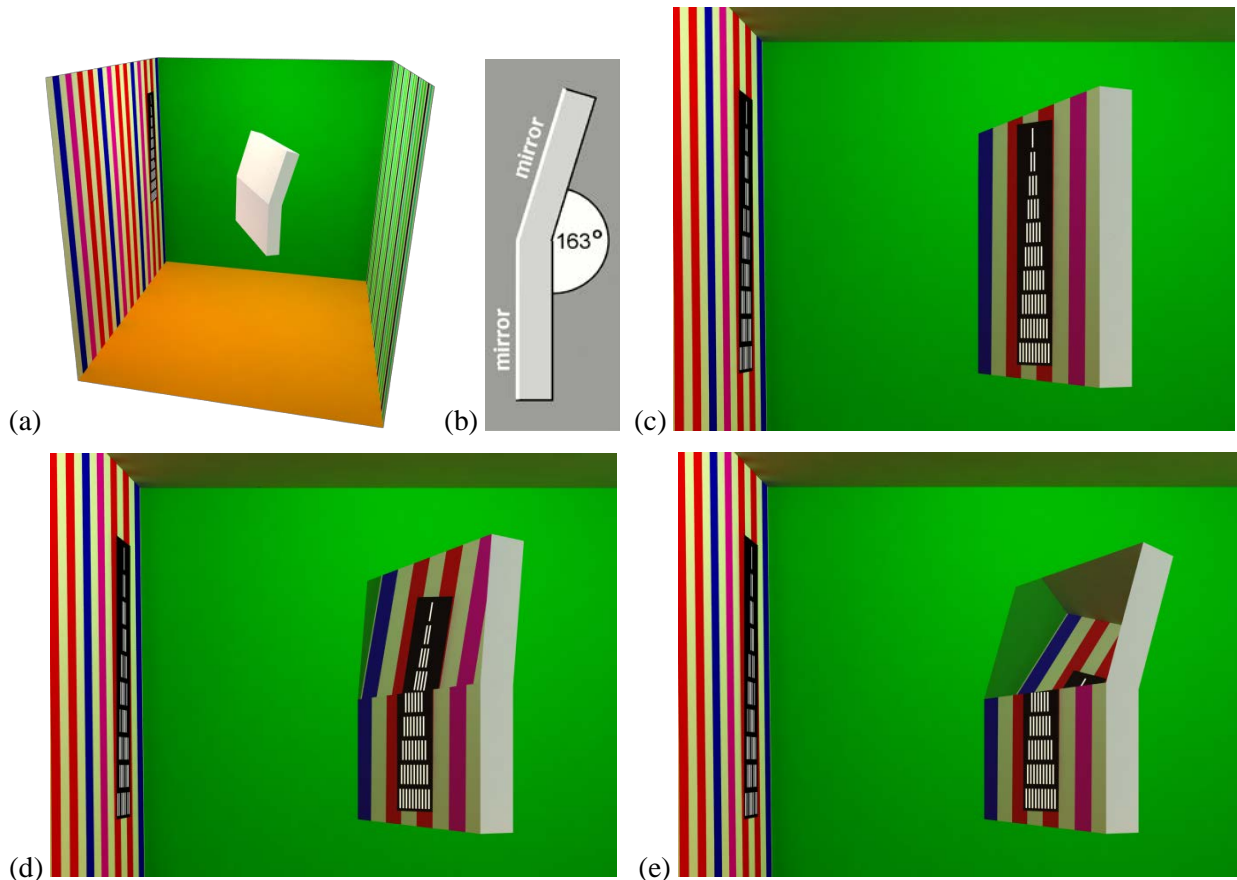


Figure 5: Reflections of a rectangular image in an angled mirror: (a) general set-up; (b) cross-sectional view of mirror; results for a dihedral angle of: (c) 180° , (d) 175° , (e) 163° .

4.1 Surface Reflections

Pure reflections on mirror surfaces are easy to understand and do not require intensive examination. But it is still useful to set up a demonstration that models the reflection of a simple rectangle across a convex polyhedral edge, to show how unrealistic the reflections are that were painted onto the top left of the RCO. For this demonstration we just use two mirrors sharing a common edge (Figures 4a, 5a,b) and orient them so that the rectangular, black&white picture on the left wall of the Cornell Box gets reflected across that edge.

In Figure 5(c) the two facets are coplanar and the window is reflected as a single rectangle across the common mirror edge. In Figure 5(d) the dihedral angle between the two facets has been reduced slightly to 175° , and the reflected rectangle is broken into two disjoint elements, which get sheared farther apart as the dihedral angle gets reduced further. The reader is probably familiar with this effect from looking into a medicine cabinet with more than one mirrored door. Two rays hitting the two faces close to the hinge-edge and close to one another travel off in quite different directions and can no longer hit the same spot on the reflected object, e.g., the rectangular picture.

In Figures 5(e) the dihedral angle has been reduced to 163° , and already it becomes rather difficult to make one and the same window edge appear as a reflection in both mirror facets. In order to have a window reflected in two adjacent facets of the RCO, which incur a dihedral angle of 144.74° between them, the window would have to be very close to the RCO, or it would have to be very large. But under no circumstances would the scene reflected through that window appear as a contiguous, folded image of what one would see when looking out the window.

4.2 Refraction in the Water

Simple views through a block of parallel-plate glass (Figure 7) or through a triangular prism, involving just two refractive events (entering and leaving the transparent glass body), are also simple to model and to understand. Thus we could progress quickly to studying the primary effect of refraction in the water: i.e., the displacement of the locations of the polyhedron edges in the back wall of the RCO shell. For this purpose we just used the geometry of the water body alone (Figure 4b), properly surrounded by the filler framework for easy reference to the overall geometry of the RCO. Figure 6(a) shows a non-physical depiction in which the water surface is marked with a thin, reflecting glass plate and the volume below it has been given a bluish transparency but has the same refractive index as air. The struts of the back surface of the RCO thus appear tinted, but in the same locations as they are seen in the empty filler framework (Figure 2b).

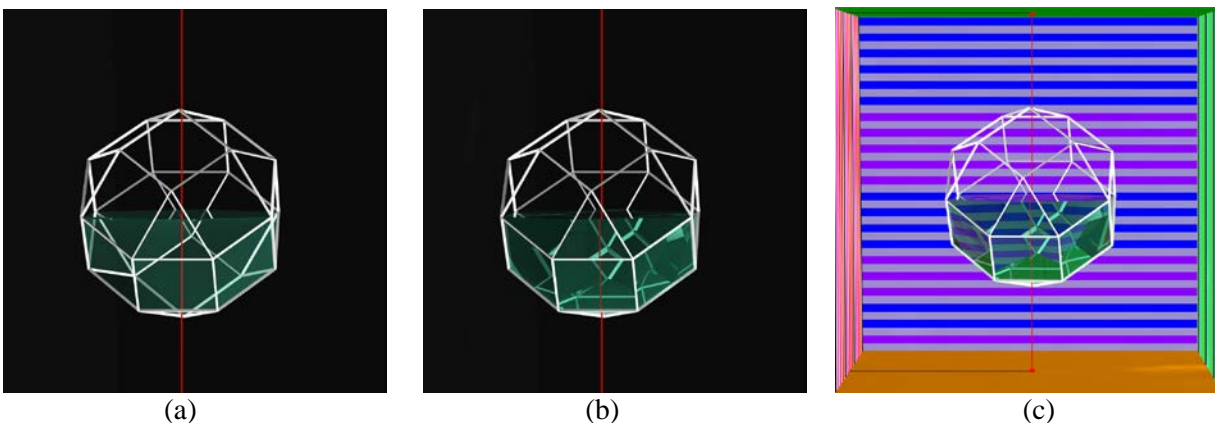


Figure 6: Refractions in the water-body: (a) no refraction, water: $n=1$; (b) proper refractions in the water body showing displacement of the backside filler struts; (c) same in striped Cornell box.

In Figure 6(b) water is assigned the known index of refraction ($n=1.333$), and the refractive processes are modeled properly; two levels of refraction are needed to see the opaque back struts; two consecutive refractive events are also sufficient to see the displaced stripes on the back wall of the Cornell Box (Figure 6c). If we also include the glass plates, then four consecutive refractive events are needed to see the back wall through the water-filled half of the RCO (Figure 11). Note that the back struts appear in quite different locations from Figure 2(b) and from what is seen in the painting (Figure 1b). This raises serious doubts that the painter actually had a chance to observe a glass polyhedron partly filled with water. It should also be noted how closely the geometry of the rendered back edges of the RCO in Figure 6(b) resembles the geometry in Figure 2(a).

4.3 Multiple Refractions and Reflections

Even in scenes of modest complexity, it is difficult to anticipate where certain features in the rendering will appear when there are multiple refractive events and many possible total internal reflections (depending on the exact angle under which some interface is hit by the ray). Selectively restricting the number of reflection and refraction events admitted and processed by the ray-tracer is a crucial tool to understand what is going on. Figure 7 shows different renderings of a simple rectangular glass brick (Figure 4c) placed inside a textured Cornell Box (Figure 7a). In Figure 7(b) the ray tracing tree has been limited to just the first mirror event ($M=1$), and a faint reflection of the right wall can be seen in the large, frontal prism face. There are even fainter reflections (invisible in print) in the left-facing front face.

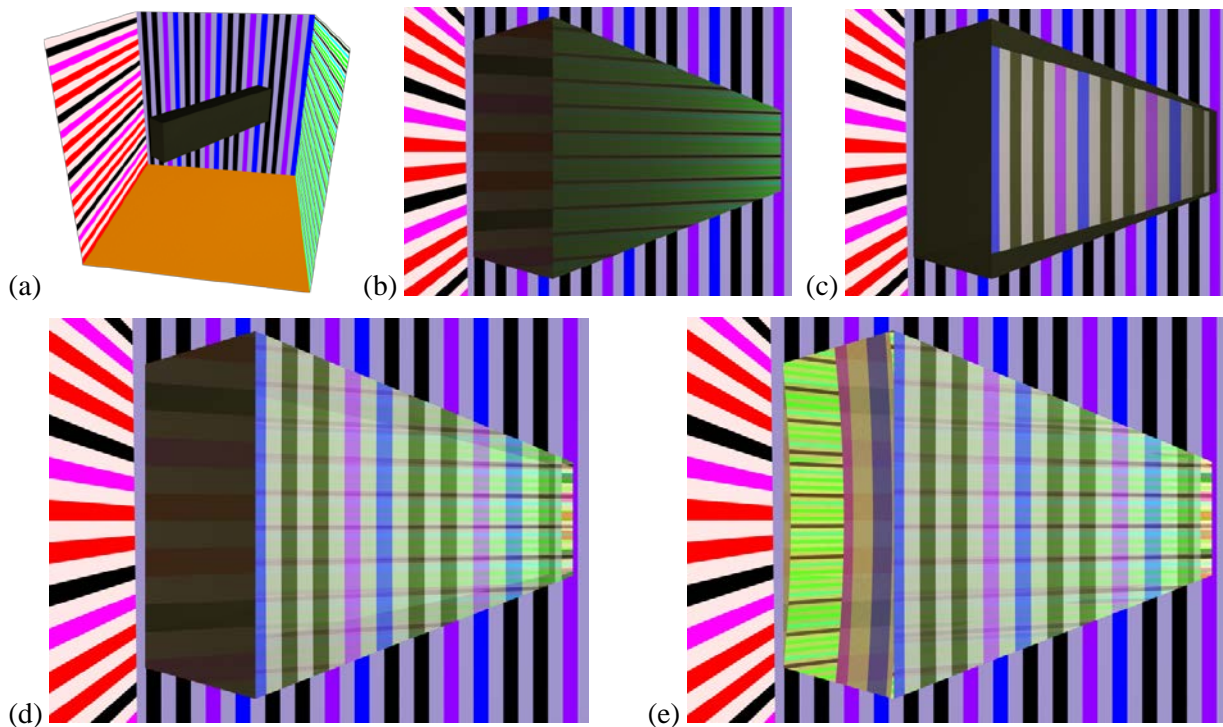


Figure 7: Refractions and reflections in a brick of glass with a varying depth of the ray-tree: (a) general set-up; results for: (b) $M=1$; (c) $R=2$; (d) $M=1$, $R=2$; (e) $M=8$, $R=8$.

In Figure 7(c) we have limited ray-tracing to the first two refraction events ($R=2$); this allows us to see through the prism and to observe the stripes on the back wall in horizontally shifted positions. In Figure 7(d) we allow one mirror event and two refraction events to take place. The most pronounced new effects are the total mirror reflections on the inside of the top and bottom surfaces of the glass brick, after which those rays also exit through the back face, thereby extending the visible vertical stripes of the back wall throughout the whole height of the front face of the brick. There are also total reflections occurring in the

right/back face of the brick, which now shows a thin sliver of the left room wall with its red and pink stripes. All of this is masked to some degree by the mirror effects in the front brick face. Finally, in Figure 7(e) we allow as many as 8 mirror and 8 refractive events; this now permits rays entering the front/left face of the prism to exit through the back/right face after three internal total reflections and to display the green and yellow stripes from the right-side room wall near the back corner. The displayed bars result from these rays that undergo true prismatic refraction, i.e., they exit from the glass in a direction different from the one that they entered; thus they show up slightly curved, because of the change in view angle with the height at which rays enter the glass brick.

4.4 Internal Reflections at the Glass/Water Interface

The real difficulties appear when we start to combine refractions and reflections. Key problems encountered in earlier modeling and rendering attempts were caused by the glass/water interfaces in the lower half of the RCO shell and by the handling of the possible total internal reflections at these interfaces. Often proper results cannot be obtained, if one just models a body of glass with a body of water next to it. It all depends on whether the ray-tracer will properly combine the exit event from the glass with the entry event into the water. If these two surfaces are not properly fused into a single interface, then under certain angles of incidence (more than 41.8° away from the normal) a total internal reflection in the glass will be introduced, based on the ratio of the refractive indices of glass ($n=1.5$) and vacuum ($n=1.0$). Such rays will then not be pursued into the water body, even though a proper calculation based on the ratio of the refractive indices of glass ($n=1.5$) and water ($n=1.33$) would allow rays to propagate, as long as they deviate from the interface normal by less than 62.5° . The situation is even worse when exiting from the water-body: At angles greater than 48.7° a ray cannot exit into air; but it can always exit into glass, since glass has a higher index of refraction.

To make sure that our rendering engine does the right thing, we stacked two thick transparent plates (half-cubes) of different materials behind each other (Figures 4d, 8b) and traced rays through this “sandwich” at various angles of incidence. The only reliable way to obtain the proper results expected from physics is to model this composite of two bricks of different material with a single shared interface between the two half-cubes – rather than as two separate, self-contained bodies. This interface must then be properly tagged with the relative index of refraction between the two materials, i.e., the ratio of the individual indices on either side of the interface. To demonstrate these effects, a simple cube shape has been suspended in a Cornell Box with differently colored striped walls, with one edge of the cube pointing towards the camera (Figure 8b). In a first experiment the whole cube is made out of solid glass, and the results are shown in Figure 8(a). In both front faces, near the front edge we can see the horizontal stripes of the back wall; they result from rays that pass through the cube as if it were a thick glass plate, with the exiting ray parallel to the incident ray. In the top and bottom portions of these areas on the front faces we can also see bands that result from total internal reflections on the top and bottom faces of the cube. Roughly half-way towards the left and right outer edges of the two front faces, the images change abruptly into sets of vertical stripes; they result from rays that experience one internal total reflection on one of the back faces and then exit towards the opposite side wall through one of the cube’s back faces. The color of the displayed stripes clearly indicates which wall is being seen in this indirect manner.

In Figure 8(b) the cube has been split into two half-cubes. The left/back half-cube is a water-body that has been placed close to the front/right half-cube, which is made of solid glass, as before. The display in the right front plane of Figure 8(c) has not changed much, except that it has taken on a greenish tint because all the rays, whether internally reflected or not, pass also through the water half-cube. We use a volumetric tinting model (see Section 4.6); water adds a bluish tint and the glass has a greenish-yellow color.

At the far left end of the left/front panel we can now see some of the horizontal stripes on the back wall. It is not immediately obvious what exact ray paths produce those stripes. It might be that the lower refractive index of water creates a small zone where that edge of the water cube acts as a 90° prism. On

the other hand, it might also be that rays entering in this zone get reflected on the back wall and then experience a second reflection at the water/glass interface so that they still can exit towards the back wall from the water body. By judiciously controlling the number of allowed reflections (as in Section 4.3), and by verifying the various ray paths with our 2D ray tracker (Section 4.5), we could determine that the latter explanation is indeed correct. A little further forward, the rays experience total internal reflection on the back wall of the water half-cube and then exit through the right back face of that cube towards the right wall (red beam in Figure 9b).

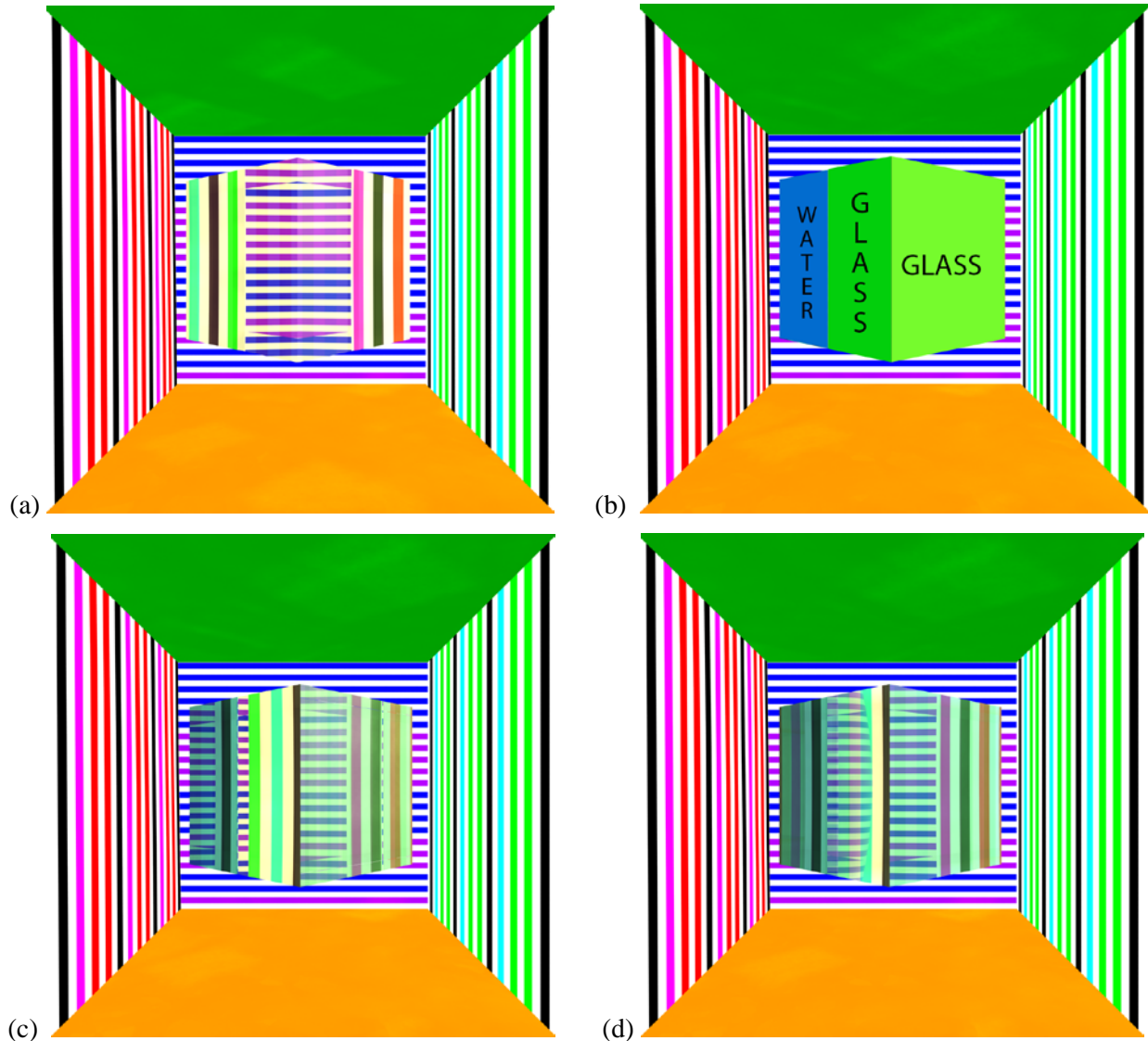


Figure 8: (a) Refraction through a solid glass cube. (b) Placement of two half-cubes in Cornell box. (c) View through a glass half-cube in front of a water half-cube; (d) view through the two half-cubes properly merged with a single shared interface.

In the front half of this left face, rays enter the glass half cube, in the same way as illustrated in Figure 9(e); but here all of the rays experience total reflection at the back glass wall – which is not acting as a proper interface to the water body because of the small air gap between the two half-cubes. Again we see two subdomains: In the left domain rays experience a second internal reflection at the right front face and

exit through the right back face towards the back wall (green beam in Figure 9e). In the whole domain in front of this, the glass body acts as a prism with one internal reflection: Rays experience one total reflection at the back of the glass body and then exit the half-cube through the right/back face towards the right wall (yellow and red beams in Figure 9e).

Figure 8(d) shows the result when the glass/water interface has been properly merged, so that rays can transition directly from glass into water; how a proper model can be obtained in Maya is explained in Appendix A. Nothing significant changes in the right front face, because none of the rays there have experienced an inappropriate internal reflection at that shared interface. But in the left face we see some significant changes: In the front half, a wide swath of rays hit the glass/water interface at a steep enough angle, so that they can now exit through the water body and reach the back wall. In this domain, where we now see rays passing properly through the glass/water interface, just short of avoiding total internal reflection, we see distinct signs that the result is a superposition of two sub-trees of rays pursued by the rendering system: We are still seeing some weakened internal reflection showing the green and yellowish stripes of the right wall. Note that the boundary showing the onset of total internal reflection is curved, since the critical angle where this happens also involves the elevation angle of the ray. Meanwhile, in the back half of the left face, we see two sub-regions: The right sub-region demonstrates the same effect (illustrated by the red beams in Figures 9b and 9d) as seen in the corresponding region in Figure 8(c), whereas the left sub-region shows an entirely different effect. This is due to rays that experience one internal reflection at the back water face, transition through the interface, experience two more internal reflections at the back/right and front/right glass faces, transition again through the shared interface, and exit through the front/left water face towards the left wall – in total, a series of 4 refractions and 3 reflections (Figure 9c, and yellow beam in Figure 9d).

4.5 Two-Dimensional Ray-Path Visualizer

In order to help us understand the complex paths that rays can take even in a simple physical object made of a half-cube of glass in front of a half-cube of water, we also programmed a simple 2-D visualizer application. This application renders the ray patterns occurring in a horizontal plane through the “eye” (or camera lens) and some test object. Figure 9 shows consolidated outputs from our visualizer for the glass/water cube; the camera is at the bottom of the frames and the cube near the top. The water/air boundaries of the water half-cube are shown in blue; the glass/air boundaries of the glass half-cube are shown in yellow; and the merged glass/water interface is depicted in green. White/grey lines correspond to individually traced rays, and their brightness indicates the amount of light energy carried by that particular ray. More details on this program can be found in Appendix B.

Figures 9a and 9b correspond to a composite of two separate half-cubes with a thin air gap between them. Figure 9(a) shows the path of a single ray and its refracted and reflected sub-trees; the ray is being totally reflected at the water surface lying closest to the glass half-cube. Figure 9(b) shows a whole swath of such internally reflected rays (yellow beam), all ending in a beam exiting towards the back wall.

The subsequent panels correspond to the situation where the water touches the glass surface and a properly fused interface has been created at that location. Figure 9(c) shows that the same incoming ray that was displayed in Figure 9(a) can now transition through the green interface into the glass half-cube, where it undergoes two more total internal reflections, travels back into the water body, and eventually heads towards the left wall of the Cornell box. In each of Figures 9(d) through 9(f), we show the result of merging multiple screenshots into a single composite between boundary angles where the rays undergo a serious change in the sequence of refractions and reflections; ray bundles experiencing the same series of reflection and refraction events share the same color. Using these composites, we are able to identify the wall that each bundle of rays ultimately ends up striking, and thus we can explain the appearances of each region seen in Figure 8(d). This visualization program confirms the explanations given in Section 4.4 for the behavior of individual rays within each distinct region.

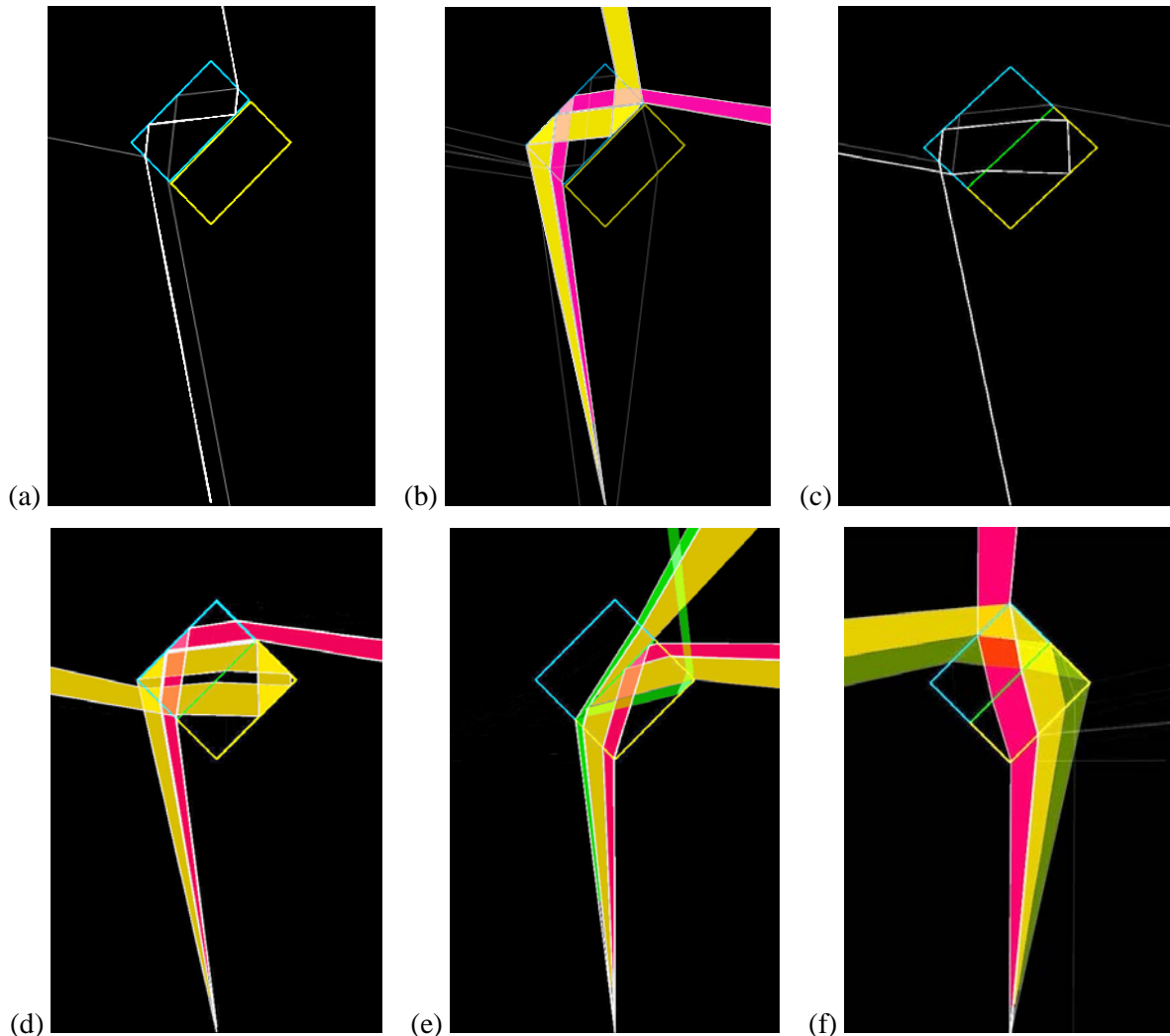


Figure 9: (a,b) Tracing light rays through two glass and water half-cubes with an air-gap: rays are totally reflected at the internal interface. (c-f) Using a properly merged interface: (c) a single ray experiencing 4 refractions and 3 reflections; (d) ray bundles entering through the left water face, (e) through the left glass face, and (f) through the right front glass face

4.6 Tinting Due to the Passage Through Water

Most advanced renderers allow rays that travel through some medium to take on gradually the color of that medium. They do that by exponentially decreasing the difference between the original color of the photons and the color of the medium as a function of the distance traveled in this medium. In Maya the two parameters that specify this rate of tinting are “Max Distance” and “Color at Max Distance.” For more details, refer to the Mental Ray [14] documentation, section 2.6.2.5. We use this mechanism in the final renderings of the RCO. The color of the glass plates will be set to the average surface color of the RCO as displayed in Figure 1a. For the body of water we use a somewhat exaggerated bluish color to make the presence of this medium quite apparent.

5. Rendering the Complete RCO in the Cornell Box

Once we have gained confidence that the chosen modeling technique and the adopted rendering environment produce physically justifiable results, we can apply the same methods to the assembled RCO as described in Figure 3. Special care had to be taken to create the properly merged interfaces where the body of water was in contact with the glass plates forming the container. For eight of the 26 glass plates, their inner surfaces had to be split into two parts – one interfacing with water, and the other interfacing with air. Overall we have four different types of interfaces (Figure 3b): One instance of water/air (W); 26 + 17 instances of glass/air (G), 17 instances of glass/water (I), plus all the opaque surfaces (O) associated with the filler framework located around the edges of the glass plates. Any ray that hits the latter type of surface is terminated and returns the illumination value found at this location. The first three types of surfaces are modeled as smooth surfaces that permit refraction as well as reflection processes to occur; in general they will split an incoming ray of generation n into two sub-rays of generation $n+1$; the relative strengths of the two sub-rays are determined by the Fresnel equations [8].

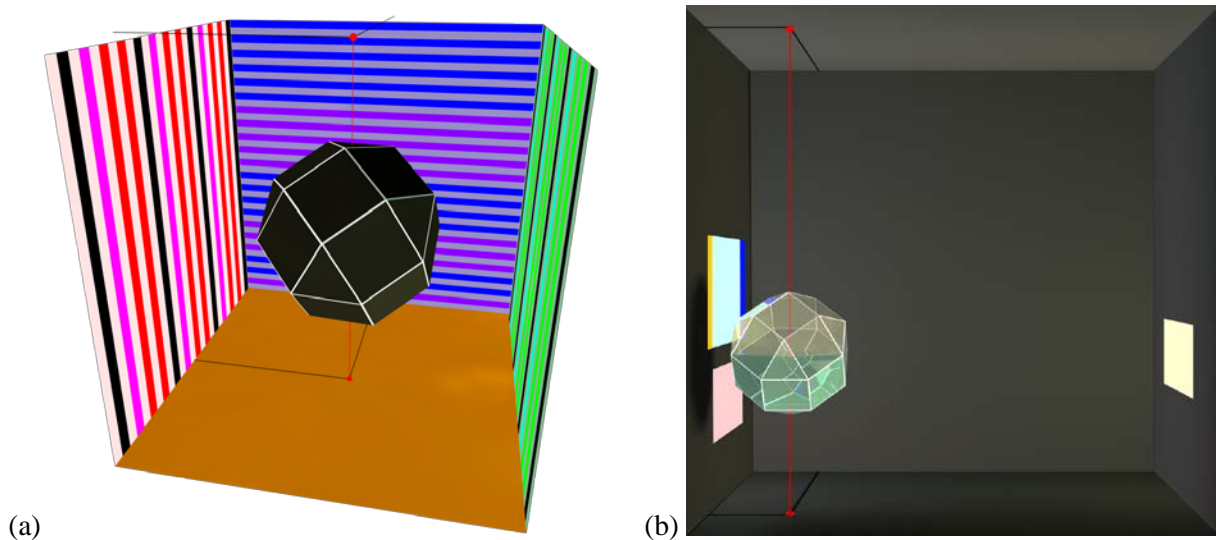


Figure 10: Placement of the RCO inside virtual Cornell Boxes: (a) for detailed studies of refractions and internal reflections, and (b) for final studies of external window reflections.

This RCO model is first placed in the center of a virtual Cornell Box with distinctly striped walls (Figure 10a), so that the final stopping point of any ray being reflected on or refracted through the RCO can be identified more easily. The RCO is viewed from the front of the Cornell box along its central axis. In Figures 11(a-f) we focus on the reflections and refractions in the RCO. In this setting we now use ray-traced renderings with a controlled depth of the refraction/reflection event tree.

All walls of the Cornell box have purely diffuse coloring. The RCO has a mix of diffuse, reflective, and refractive surfaces: The filler frame is diffuse, but the glass plates and waterbody are modeled using diffuse, reflective, and refractive surfaces. The strength of the diffuse component has been set to about $2/3$ of the combined strength of the refractive plus reflective components, which seems to give the proper amount of dustiness on the final RCO rendering.

In the following experiments we have introduced the following lights into the scene: a weak ambient illumination, a directional light shining from above onto the top/left of the RCO that is six times stronger, and another directional light shining from below onto the bottom right/front of the RCO that is another factor of two stronger. These lights were chosen to approximate the light balance on the painted RCO for the final renderings.

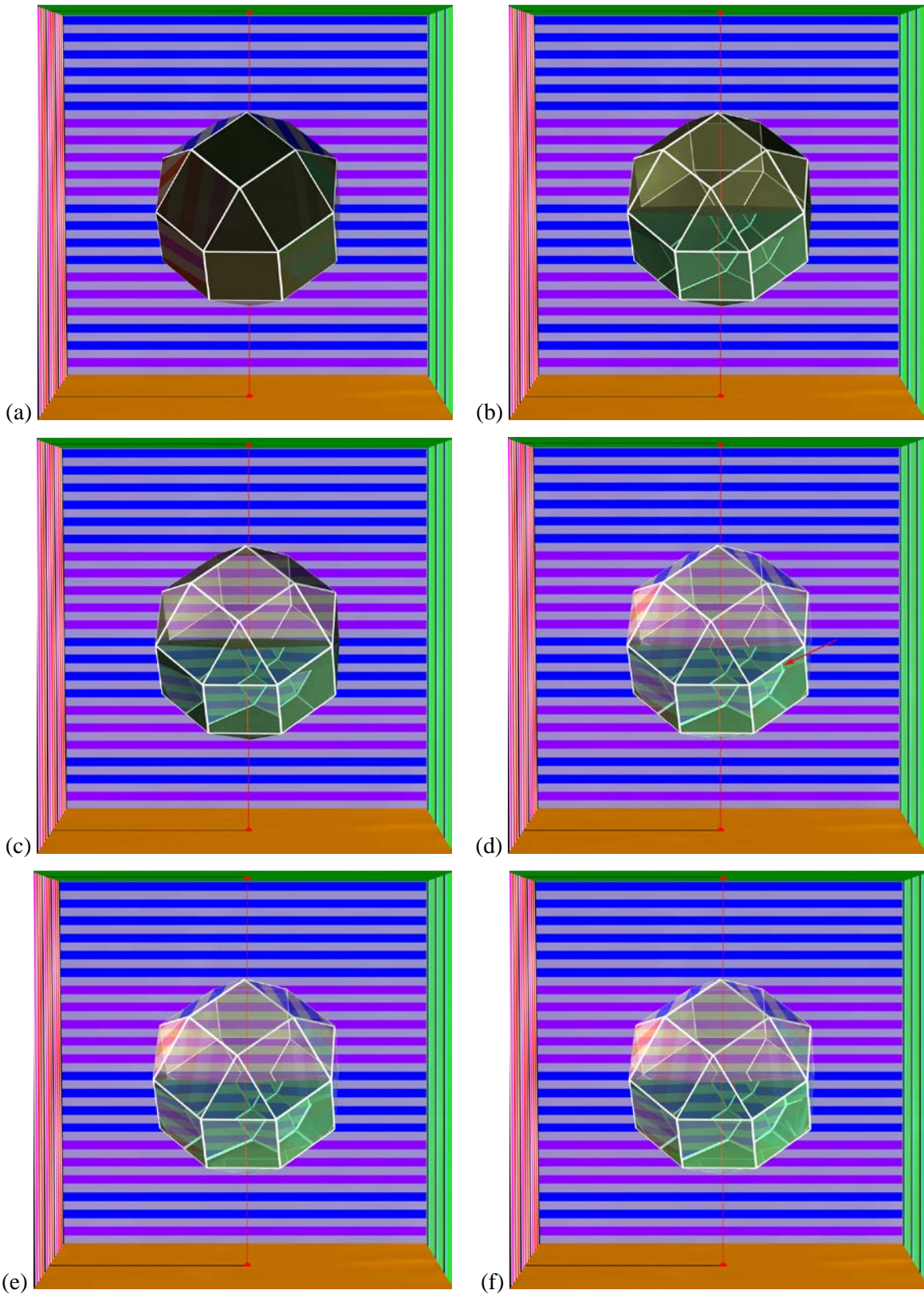


Figure 11: Ray-tree limits: (a) $M=1$; (b) $R=3$; (c) $R=4$; (d) $M=1, R=4$; (e) $M=1, R=6$; (f) $M=R=10$.

Figure 11(a) shows the result when we allow only a single reflection ($M=1$). Reflections of the walls can be seen most clearly in those glass panels that have a shallow grazing angle of incidence, e.g., the top two panels.

Figure 11(b) allows up to three refractions. Three refractive crossings of material interfaces are necessary to allow the filler framework at the back to be seen through the water (Air / Glass / Water / Glass \rightarrow Filler). No internal or external reflections are allowed. Note the displacement of the red string inside the water, compared with the location above water – a similar effect is observed with the filler framework.

With four refractions (Figure 11c) things get interesting, since rays can now exit from the back glass plate and thus produce images of the walls of the Cornell Box. Most prominently we now see the horizontal stripes on the back wall in both the upper and the lower halves of the RCO. Some of the filler framework from the bottom of the RCO begins to appear on the water surface (not visible in the printed version, but in the online color display at 300% magnification); they result from rays that are refracted through the top surface of the water and then reach filler material in the bottom half of the RCO.

With four refractions and one reflection allowed (Figure 11d), we see again the surface reflections that were already visible in Figure 11(a). The bottom-most triangular facet and right-most quad facet have become much brighter because of the strong additional reflective component produced by the glancing angle. Also, the water surface, which has remained mostly dark so far, now reflects portions of the filler frame from the upper back half of the RCO. Moreover, we can identify an instance of total internal reflection: Looking at the top right corner of the bottom/right facet, new features have appeared, for instance, an additional filler strut (red arrow). As this feature has not been previously seen in Figure 11(a), we conclude that the rays responsible for producing it must have undergone an additional internal reflection in the RCO.

In Figure 11(e) we have added two more levels of refraction and see some additional complex effects in the bottom facets of the RCO – which all contribute to the final image in Figure 11(f). The water surface also changes color due to the refracted components of rays hitting this top surface, which can now travel all the way through the bottom glass plates of the RCO to the orange floor.

As we increase M and R beyond 6 there is very little change in the displayed image. Some small differences between Figures 11(e) and 11(f) are visible in the bottom facets: A few more reflected details of the filler framework can now be seen. These features result from some lengthy chains of reflections and refractions, which are impossible to track by a human mind without computer assistance. Thus, with $M=R=10$, which is the maximum number of refractions and reflections that Maya calculates, we have essentially established “ground truth” for any visual effects resulting from refraction and/or reflection on this RCO model.

6. Rendering the RCO in an Environment Similar to the Pacioli Painting

Having thoroughly tested our virtual RCO model and our rendering environment, we can now rely on it to give us a depiction that is close to physical truth with respect to the placement of the various geometrical elements resulting from reflection and refraction. There are still many parameters that could be fine-tuned: the refractive index of glass; the amount of tinting imparted by the glass plates and the water body; and the amount of diffuse scattering on the outer surfaces – perhaps due to the presence of dust. We chose some of these parameters to make our rendering look somewhat similar to the depiction in the Pacioli painting, as depicted in Figure 1b. However, our main focus was on showing the geometrical issues related to the placement of the reflected and refracted geometrical features, because these effects are most relevant to the question whether the artist was actually observing an RCO half-filled with water.

To produce the final renderings (Figure 12), we have suspended the RCO in a darkened room with three bright, colored “windows” (Figure 10b); these are really just bright, flat “paintings” on the walls. The RCO was placed close enough to the large cyan “window” on the left wall, so that portions of it

appear in both the upper left panels of the RCO, where the original painting shows the primary window reflections. This “window” was enhanced with blue and yellow vertical edges to make it easier to understand what is being reflected in those two facets of the RCO. In the upper panel, we can see the blue back-edge of that window; in the lower facet, the yellow front-edge appears. This disparity in reflected window portions illustrates the physical impossibility of a single continuous image of the window appearing across both faces, as depicted in the Pacioli painting.

There are other features on the surface of the painted RCO that can be interpreted as depicting reflections. The lower right of the RCO has some pronounced greenish tint that can be seen as a reflection of the green table cloth, and a fuzzy dark feature might represent one of the bodies standing behind the table. More distinctly drawn, a small black reflection in the front triangular face of the RCO (Figure 1b) was probably meant to represent the artist who drew the image of the RCO; however, it is unrealistically small. A human observer standing some distance d in front of a plane mirror will see his reflection as being located at that same distance d behind the mirror, and the perceived “image” at the reflecting surface will be half the size of the observer. Thus the black figure outline is much too small to be a reflection of the artist; to a first approximation, it should be about the same size as Pacioli or his companion. In our computer rendering, the small black reflection shown was produced by a 6 inches tall puppet cut-out, located in the plane of the camera. Moreover, since the triangular front face of the RCO is not perpendicular to the line of sight, the puppet had to be displaced about ten body lengths downwards and to the left from the position of the camera lens – supposedly the eye of the artist.

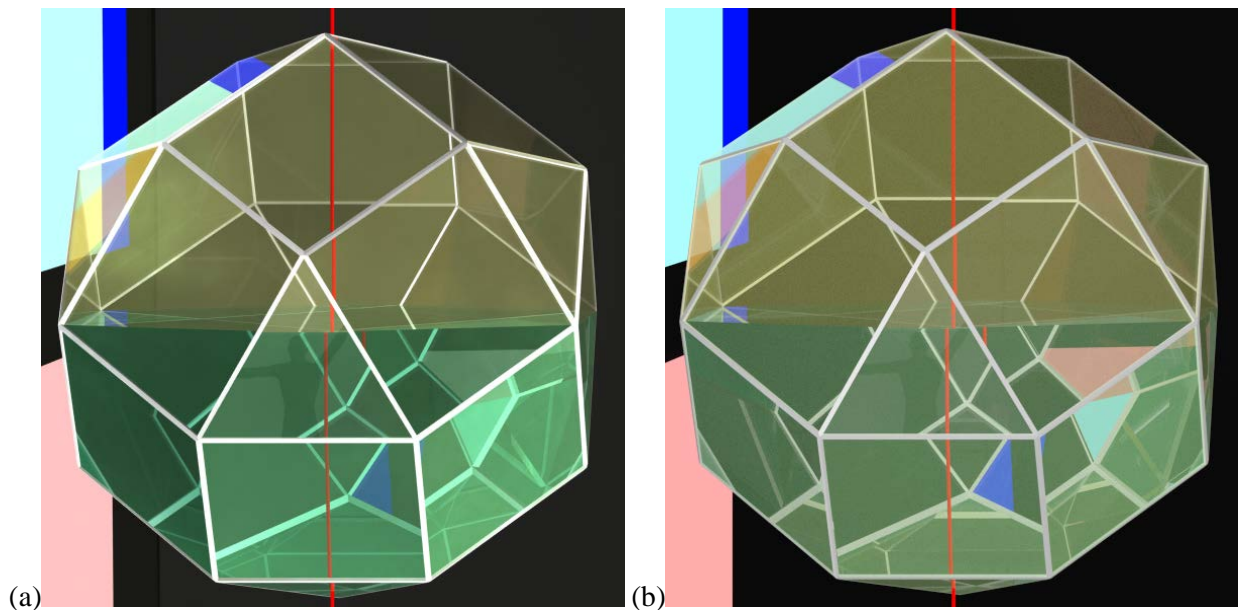


Figure 12: *RCO in a dark room with windows: (a) rendered with Maya to a recursion depth of 10; (c) rendered with Blender to a depth of 40.*

The computer renderings (Figure 12) also show internal reflections in quite different locations compared with the painted RCO. For instance, we see portions of the light blue window and its dark blue back edge in two facets in the bottom right quadrant of the ray-traced RCO, but none of these have a similar appearance in the painted RCO, even accounting for the murkiness of the painted water. Conversely, the painted RCO shows some internal reflections that are not seen anywhere in the ray-traced RCO: in the back/right center quad facet above the water surface, and along the bottom/right of the RCO. In attempting to replicate these extra internal reflections seen in the painted RCO, we expanded the windows to fill the entire left wall, but still we could not obtain the desired internal reflection in the upper half of the RCO.

As mentioned before, the refracted locations of the string and the filler framework also appear in inappropriate locations. In the painting, the string appears as a continuous line above and below the water surface, whereas in the computer rendering the string is discontinuous at the water surface. The displacement is caused by the different indices of refraction of air and water. The same applies to the appearance of the filler framework; our ray-traced RCO renders the refracted framework struts in very similar locations to the physical model shown in Figure 2(a) – in contrast to the painted RCO.

In order to check the reproducibility of our rendering results and the robustness of the step-by-step validation approach recommended above, we also engaged a different rendering program: Blender [4] is an open source program readily available to everybody, and may thus be a candidate that many JMA readers might use for some quick experiments. Once our model data had been adjusted to be compatible with the Blender environment, the rendered geometrical features looked the same as with the Maya renderer. We could also confirm that increasing the depth of the recursive ray-tree from 10 to 40 did not result in any meaningful differences; any changes that might be attributable to increased ray-tree depth are small compared to the noticeable variations in the displayed intensities of reflected and refracted components between the two rendering programs. The major differences between the two renderings occurred with respect to hue changes and brightness variations. Different rendering programs use different ways of letting the user specify the relative importance of the specularly reflected, refracted, and diffusely reflected photons, resulting in different relative intensities of some of the struts viewed via several instances of reflection and refraction.

More troublesome for our particular rendering comparison was the fact that Blender permits volume-tinting only for domains surrounded by a closed manifold surface, but not for volumes that have some individually merged interfaces as part of their boundary representation. Thus the tinting of the glass and water components had to be achieved entirely via some approximate changes in surface parameters. This explains why the reflections of the pink window are much more vivid in the Blender rendering – the rays experience less tinting while traveling a substantial distance through water.

This emphasizes yet again the need to carefully evaluate the capabilities and idiosyncrasies of any rendering environment employed, by using an incremental approach that starts with some simple, well-understood test geometries.

7. Discussion and Conclusions

The RCO plays an important role in the Capodimonte panel. It depicts an important accomplishment of the famous mathematician Fra' Luca Pacioli and was clearly meant to impress viewers of this painting. Baldasso [3] even argues that the RCO plays the role of a third figure in this composition.

Our analysis indicates that it is highly unlikely that the artist who painted the rhombicuboctahedron in the Pacioli painting was looking at a physical glass RCO partly filled with water of the size implied in this painting. One of the anonymous reviewers [1] contributed the following valuable information:

In 1495 it was possible to make a small glass model of a polyhedron, and several are mentioned in two sixteenth century inventories of the Ducal palace at Urbino where Pacioli had worked; for technical reasons to do with the available glass it is not possible that these models were as large as the Pacioli RCO appears, nor could they have been strong enough to hold several liters of water.

However, the artist probably had access to an empty RCO model made of triangular and square glass plates. It also seems quite likely that a depiction of the RCO was first sketched or painted separately from such a model, because the depicted geometry of the RCO edges is in excellent agreement (including perspective distortion) with the computer generated rendering. When this initial depiction was later copied into the Pacioli painting, the discrepancies in the RCO's scale and its perspective projections, discussed at the beginning of this article, were introduced.

The primary window reflection across the top left facets is so far off from any possible reality that it cannot have been drawn based on an actual observation. It may, however, be possible, that the painter observed the general nature of some window reflections on various polyhedral glass models and decided that reality was much too confusing for most observers to yield a pleasing painting. The artist may then have made a conscious decision to render a window reflection that would be more plausible and would help to identify the location of the space portrayed. A contiguous picture of the ‘Duomo in Urbino’ can readily be understood by most viewers, whereas physically realistic reflections that break this image into disconnected pieces would be puzzling to most observers.

Moreover, to emphasize the transparency and 3-dimensionality of the RCO, two other internal reflections of the same window were added in two facets on the right hand side, one just above the water surface, and one on the bottom right in the water. The first one occurs where one would assume to see a secondary image, if this were the result of an intersection of a parallel beam of light coming through the assumed window. Furthermore, the contrast and saturation of these imagined reflections appears unnaturally strong compared to the other depicted features in the RCO. In addition, the apparent luminosity of the three reflections has been enhanced by the artist, by surrounding them with a slightly darkened halo.

Though we have focused in this article on the defects with respect to a physically realistic rendering of the RCO, we should remember that physical realism was not the primary objective of the painter. A more likely goal was to present this intriguing mathematical object in the best possible way that would bring on a sense of wonder and awe in most viewers. We would like to agree with many art historians that this goal has been achieved extremely well despite the discussed rendering flaws.

Acknowledgements

We would like to thank the anonymous reviewers for their many constructive comments and for some valuable insights into the context in which the discussed painting has been created. This work is supported in parts by the Undergraduate Research Apprentice Program (URAP) at U.C. Berkeley.

References

- [1] Anonymous referee, communication during the review process.
- [2] Autodesk Maya. -- <http://www.autodesk.com/products/autodesk-maya/overview> (Dec. 2014)
- [3] R. Baldasso, *Portrait of Luca Pacioli and Disciple: A New, Mathematical Look*. The Art Bulletin, (March-June 2010)
- [4] Blender Foundation, *Blender*. -- <http://www.blender.org/> (Dec. 2014)
- [5] C. Boehring, *Physical model of the RCO*. Private communication. -- claude.boehring@mac.com (Dec. 2014)
- [6] A. Ciocci, *Il Doppio Ritratto del Poliedrico Luca Pacioli*. De computis. Revista Española de Historia de la Contabilidad 15 (2011), pp 107-130.
- [7] Cornell University, *The Cornell Box*. -- <http://www.graphics.cornell.edu/online/box/> (Dec. 2014)
- [8] *Fresnel's Equations*. -- <http://hyperphysics.phy-astr.gsu.edu/hbase/phyopt/freseq.html> (Dec. 2014)
- [9] E. Gamba, L'Umanesimo matematico a Urbino. A. Marchi and M. R. Valazzi, eds., *La città ideale: l'utopia del Rinascimento a Urbino tra Piero della Francesca e Raffaello*. (Milan: Electa, 2012), pp 233-247.
- [10] A. S. Glassner, *An Introduction to Ray Tracing*. Morgan Kaufmann, 1989.
- [11] G. Hart, *Luca Pacioli's Polyhedra*. -- <http://www.georgehart.com/virtual-polyhedra/pacioli.html> (Dec. 2014)
- [12] J. Logan, *An Analysis of Certain Mathematical Cyphers in the Portrait of Fra Luca Pacioli*. Unpublished draft; private communication, Jan. 19, 2015.
- [13] N. Mackinnon, *The portrait of Fra Luca Pacioli*. The Mathematical Gazette, 77 (1993) pp 130 - 219.

- [14] Mental Ray documentation. -- <http://docs.autodesk.com/MENTALRAY/2012/CHS/mental%20ray%203.9%20Help/files/tutorials/architectural-library.pdf> (Dec. 2014)
- [15] Museo e Gallerie di Capodimonte, *Ritratto di Fra' Luca Pacioli*. -- <http://cir.campania.beniculturali.it/museodicapodimonte/itinerari-tematici/galleria-di-immagini/OA900154> and specifically: -- http://www.polomusealenapoli.beniculturali.it/museo_cp/cp_scheda.asp?ID=16 (Dec. 2014)
- [16] G. S. Owen, *Ray Tracing*. -- <https://www.siggraph.org/education/materials/HyperGraph/raytrace/rtrace0.htm> (Dec. 2014)
- [17] J. Rekveld, *ghosts of luca pacioli*. umwelt, observations (2007)
- [18] J. Rekveld, *light matters: ghosts of luca pacioli*. -- <http://www.joostrekveld.net/?p=615> (Dec. 2014)
- [19] J. Sander, ed., *Albrecht Dürer: His Art in Context*. (Munich: Prestel, 2013), pp 190-191.
- [20] C. H. Séquin, *Misinterpretations and Mistakes in Pacioli's Rhombicuboctahedron*. -- http://www.cs.berkeley.edu/~sequin/X/Leonardo/pacioli_rco.html (May 2015)
- [21] H. Serras, *Mathematics on the Ritratto di Fra' Luca Pacioli*. -- <http://cage.ugent.be/~hs/pacioli/pacioli.html> (Dec. 2014)

Appendix A: Modeling the Glass/Water Interface

Since it is important to obtain the proper physical behavior at these glass/water interfaces, let's briefly review one example of how a proper model can be generated in Maya [2]. A suitable process is called *Merge Vertices*. It joins non-manifold edges by matching coinciding vertices and then merging these vertices. For instance, our test “sandwich” (Figures 4d, 8b), with one shared interface between the two half-cubes, can be modeled as follows:

1. Create two half-cubes of equal size and stack them behind each other.
2. Delete the back face of the front half-cube.
3. Combine the two half-cubes into a single mesh, by merging four coinciding vertex pairs.
4. Double check that all face normal vectors of the resulting mesh are oriented properly.

Maya then allows us to set the reflective, refractive, and diffusive coefficients separately for each material. It also provides a utility to use the Fresnel equations to determine the relative strengths of reflection vs. refraction depending on the incident angle of a ray, and it automatically ensures that light energy is conserved. In order to preserve proper physical behavior as expected from glass and water, we set the reflective and refractive coefficients equal for all materials used: glass-air interface; glass-water interface; and water-air interface. This is to prevent the water or glass surfaces from looking either super-reflective or not reflective enough, since the appearances of both, in actuality, are governed by the Fresnel equations. The Maya documentation shows how to create a glass material, suggesting precisely this relationship between reflective and refractive coefficients. Additionally, we tune the diffuse coefficient, which controls the fraction of light energy not devoted to reflection and refraction effects, so as to best match the slightly “dusty” look of the painted RCO. For a more detailed explanation, please refer to the Mental Ray [14] documentation, section 2.6.2.6.

Appendix B: Two-Dimensional Ray-Path Visualizer

In order to understand the complicated ways in which light rays may move through even simple transparent objects and thereby create a bewildering richness of depicted features, and also to further verify the correctness of our environment and material settings in Maya, we independently built a 2-D

visualizer application to show the paths of reflected and refracted rays, as they interact with some of our transparent test objects. In particular, to understand the rendering results from our test “sandwich” (Figures 4d, 8b), we model rays proceeding in a horizontal plane through the middle of the cube. For the sample traces shown in Figure 9 we chose a camera location and body geometry to be proportional to the environment depicted in Figures 8(a-d).

The visualizer application launches rays from a single location corresponding to the lens of the camera and sweeps them through an angular range that covers the whole width of the target. Blue lines denote a water/air boundary; green lines, a water/glass boundary; and yellow lines, a glass/air boundary. White/grey lines correspond to traced rays, and their brightness indicates the amount of light energy carried by that particular ray. We assume no light energy is lost by diffuse scattering, so that the rays’ light energies are split between reflection and refraction purely based on the Fresnel equations. To reduce uninformative clutter, we set a minimum light energy threshold (1%) as a stopping condition for any refraction or reflection branch, after which we no longer trace or display that particular ray or any of its children rays. The main loop of the program follows the model of a generic ray-tracer:

- 1.) If the recursion depth has been exceeded or the ray’s light energy is below the minimum threshold (1%), stop.
- 2.) Recursively cast a ray from the “eye” (camera lens) and find the next collision point with some interface.
- 3.) With brightness proportional to the amount of light energy carried by the ray, draw a line segment starting at the ray’s origin and ending at the collision point. If there is no collision, choose the end point of the line segment to be the boundary of the simulator domain and stop.
- 4.) Use the Fresnel equations to calculate the percentage of light energy split between reflection and refraction at the collision point with an interface.
- 5.) Initialize two new rays with origin at the collision point, representing the reflected and refracted components of the original ray after interacting with the interface. For each ray, propagate forward its light energy (calculated in step 4), returning to step 1.

Some of the rays cast through our test object will hit an edge, and rays slightly to the left and to the right of such boundary rays may undergo an entirely different scenario of subsequent refractions and reflections. Beams of rays with the same basic behavior between two such “boundary rays” are shown with the same color in Figure 9. This makes it easier to understand how completely different patterns may appear in one contiguous front-facing facet of our test object.