Mathematics of Times Square

Douglas W. Dwyer

New York City, USA; douglas.dwyer@me.com

Abstract

In Times Square, NYC, a large screen wraps around a building. In late 2024, a perfume bottle would dance inside the screen and then pop out of it. In the National Gallery in London, there is a distorted painting of a Dutch interior across the inside of a box. When viewed through a peephole, one sees a distortion-free $3D$ image of the interior. We have designed an origami piece that has distorted silhouettes and a pattern of triangles and distorted rectangles. When folded and viewed from the correct vantage point an illusion of a chessboard on the floor of a cube and silhouettes on the faces emerges. All three of the objects draw on a specific application of projective geometry. A second application is an animation that draws an ice skater that then moves around within the $3D$ space of a photograph. In this paper, we will reveal how the math works and use vector algebra as an alternative to traditional tools like a ruler and compass, a projector, or specialized design software.

Introduction

When we learned to draw from life, we were taught to project what we saw onto a flat piece of paper, using an outstretched arm and a pencil to measure proportions. If done carefully, this method could produce a drawing comparable to what a conventional camera would see — an image built on rectilinear perspective. There are many other ways to make an image. This paper describes how to make an image on a multifaceted surface so that the image looks correct from a specific vantage point. This illusion goes back at least four centuries, and there are many tools available for an artist to achieve it. In this paper, we shall employ vector algebra.

Early examples of images made on multi-faceted surfaces are the perspective boxes of Samuel van Hoogstraten constructed in the 1600s. Figure 1(a) presents one example that is in the National Gallery in London. From the outside, the interior appears crooked. If viewed from the visible hole on the left side face of the box with one eye, a distortion-free painting of a Dutch interior emerges in $3D$. David Broomfield provides a detailed analysis of how the perspective works using, the language of geometry and speculates about how van Hoogstraten would have both designed it and constructed it [1]. The fact that many of the planes of the box are parallel to the corresponding planes of the *conceptualized Dutch interior* simplifies the problem somewhat. Remarkably, there is a second peephole on the right side of the box, and the interior looks correct from this one as well!

We made the photograph in Figure 1(b) from the downtown-westside corner of 45th Street and Seventh Avenue. The image shows a digital sign that goes around the corner of a building. From this vantage point the distortions in the image were minimal. The advertisement was animated: the perfume bottle danced around the inside of the box before popping outside of the box. Several trompe-l'œil techniques enhance the illusion of $3D$ as well. The box has a frame that is part of the screen but is easy to read as being a physical frame. Further, there is an image of a billboard that reads as if it were slightly in front of the box, enhancing the sense of space. Finally, when they animate the perfume bottle, it comes in front of the *virtual frame* of the box, but the designers have been careful to ensure that the corners of the perfume bottle are never cropped in the real screen as this would break the illusion of a real bottle floating in front of the box.

Figure 1(c) is an origami piece that we have designed that when folded and viewed from the correct vantage point becomes Figure 3(c). The mathematical idea behind each of these three pieces is the same—projecting an image onto a multi-faceted surface such that when viewed from a specific vantage point they become the desired image.

Figure 1: *Perspective boxes across the centuries: (a) Hoogstraten, (b) Times Square, (c) Our Origami Model*

Artists use many different tools for accomplishing such illusions including shadow techniques, rulers, compasses, projectors, graphic design software, and commercial solutions. In this paper, we demonstrate an approach that combines vector algebra with software used by data scientists. The approach can be done on a small scale and allows for animation as well as the mapping of an image to a complex multi-faceted surface.

This paper first lays out the mathematics of a generic solution to the problem and then shows an application of the approach. We follow this with a discussion of alternative approaches that use different techniques, and these alternatives can be seen as applications of the generic approach. We then show how the approach can be used to animate an anamorphism and finish off with concluding remarks and suggestions for future work. A Gallery Supplement accompanies the paper and provides links to additional animations.

The Mathematics

Suppose we have a drawing (e.g., a silhouette or the squares of a chessboard) defined in three-dimensional space by a set of points in a specific order. For the examples considered in this paper, these points are coplanar but they need not be. We also have a picture frame and another bounded region of a plane in $3D$ space defined by a set of coplanar vertices, which we will refer to as the polygon (e.g., the face of a box). Using mathematics, we can project the drawing onto both the picture frame and the polygon. In fact, we can solve for the intersections of the drawing's points with both the picture frame and the polygon. If we can solve for one polygon, we can solve for all the intersections with all visible polygons as well. If we know where the drawing intersects the polygon in the picture frame, we can solve for where it intersects the actual polygon and vice versa. Figure 2 demonstrates how the mathematics works: it is a side view of a cube with three figure skaters positioned on it. As it is a side view, the picture frame is a line segment. The cube has been rotated so that only two faces are visible. Figure 3(a) is a photograph which is the real world analog to Figure 2 with the sensor of the camera becoming the picture frame.

Consider a polygon in $3D$ space defined by at least three coplanar points that are not collinear (e.g., one of the faces of the glass cube). Let a drawing be defined by a set of points in a specific order in $3D$ space (e.g., one of the figure skaters in the picture).

For convenience, we will define the picture plane as the plane in which Z is equal to 1 and we will place the vantage point at the origin (in $3D$ space). It is convenient to define the picture frame by its four corners which in turn can be defined by the angle of view:

$$pictureFrame = \{(-v, -v, 1), (v, -v, 1), (v, v, 1), (-v, v, 1)\}$$

where $v = atan(\theta/2)$ and θ is the angle of view.

Using this setup, any point in $3D$ space can be mapped to a point on the picture plane by simply dividing its x and y coordinates by its z coordinate (provided z does not equal 0). One can determine if the projected point is inside the picture frame by testing whether or not the absolute value of both the x and the y coordinate are less than v.

We use linear algebra to solve for where a ray intersects a plane. We can represent any point on a plane defined by a starting point XYZ_0 and two vectors A and B as the staring point plus a linear combination of the two vectors :

$$XYZ_0 + AB \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$$

where XYZ_0 would be one point in the polygon and A and B would be the difference between XYZ_0 and two other points in the polygon. The three points cannot be collinear. The matrix AB is 3×2 and the first column is the vector A and the second is B. For example, the arrows A and B (when used with their intersection for XYZ_0) in Figure 2 represent the vectors that define the plane of the visible right-hand face of the cube in $3D$ space . Likewise, the vectors A' and B' in Figure 2 define the plane of the visible left-hand face in the cube.

We define the ray through a point on the picture plane and the origin as:

$$t_3 \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$$

where t_3 is a scaler. We wish to solve for t_1, t_2, t_3 such that the ray from the origin intersects with the plane defined by the polygon:

$$ABC \begin{bmatrix} -t_1 \\ -t_2 \\ t_3 \end{bmatrix} = XYZ_0$$

where ABC is the square matrix formed by combining the matrix AB with the vector $[x_i, y_i, 1]^T$. This system of three equations and three unknowns can be solved by left multiplying both sides of the equation by the inverse of the matrix ABC (provided the ray is not parallel to the plane in which case the matrix is not invertible).

Once one determines where the ray intersects the plane one needs to determine if the projection of this point onto the picture plane lies inside the projection of the polygon onto the picture plane. There are algorithms for determining whether or not a point lies inside a generic polygon. We find that the function *pointsInPolygon* from the *R* package *secr* is effective for this task.

One repeats this process for each of the polygons that are visible from the vantage point (e.g., the three visible faces of the cube). In practice, every point in the drawing need not map to a visible portion of the polygon in the picture frame. In our first application, we aimed to have every point in the set of drawings lie inside one of the visible faces, and we have adjusted both the position and the scale of the drawings as well as the angle of view to achieve this objective.

First Application

Suppose we had a glass cube. We could paint drawings on the different faces of the cube. We could hold the cube so that we would be able to see all the drawings on all six faces of the cube. In this application, we

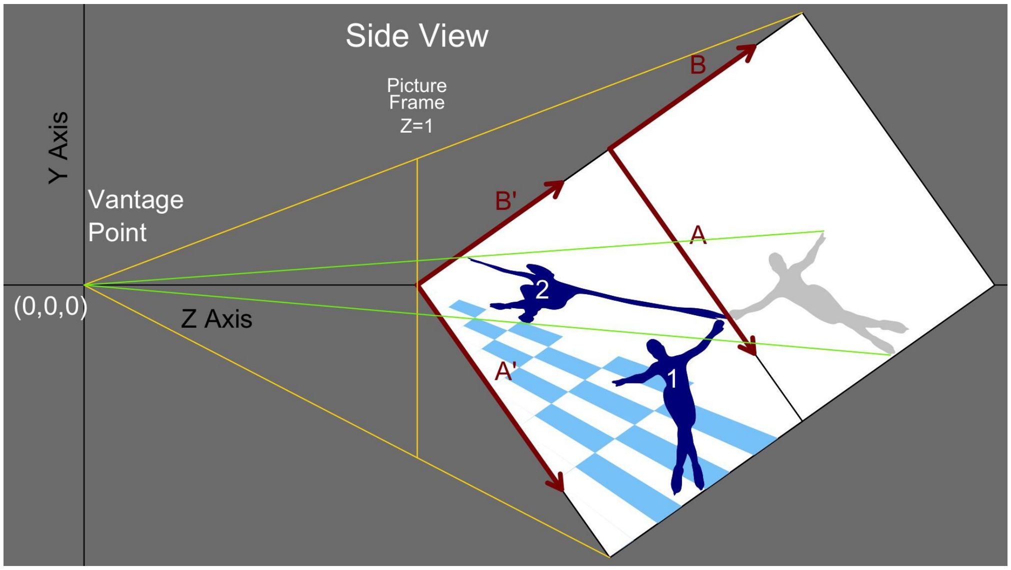

Figure 2: *Side view of an anamorphic origami box*

seek to create the illusion of figures drawn on six faces of a cube by folding a single square piece of opaque paper into a corner of a cube. This folded cube represents drawings on all six faces of the implied cube even though only three physical faces exist. In order to do this, we need to be able to project the drawings of the hidden faces onto the visible faces such that when projected onto the picture plane they correspond to what the camera would see if it were a glass cube.

The square piece of paper in Figure 1(c) , repeated in Figure 3(d), is a solution to this problem. When folded into a corner using two mountain folds and a valley fold, viewed from the proper vantage point it becomes Figure 3(c). The light-blue pattern of polygons on the left become a chessboard. There are six silhouettes of ice skaters. In the middle right of Figure 3(d), the one labeled 5 appears distortion free and has the *correct* orientation. This silhouette is distortion free in the sense that it is based on an accurate tracing of an actual photograph of an ice skater taken with a telephoto lens, using the methodology in [2]. There are three other distortion-free skaters (labeled 1,4, and 6) on the paper but with different orientations and there are two other distorted skaters (labeled 2 and 3). Distorted in the sense that the front arm is very extended and much longer than the skater's other arms and legs. When folded and viewed from the correct vantage point, one skater appears as if painted on each of the four sides of the cube as well as the two on top of the cube. The floor of the cube appears as a chessboard. This piece is constructed as follows.

The first step is to have a *conceptual glass cube* in *cube space*—a space in which each of the edges of the cube are parallel to a corresponding x, y or z axis. As we have a *connect the dots* representation of the skater and a chessboard, we can conceptually place them on the faces as desired in *(*cube space). We rotate the cube and project each shape onto the picture frame to produce Figure 3(b) – the conceptual camera view.

Once we have projections of each drawing on the picture frame, we associate each point with the visible face which when projected they lie inside. Once we have associated the points with a visible face, we can determine where they intersect with the corresponding plane.

We see the *back two* skaters, as well the chessboard, by looking through the transparent cube. Figure 2 shows the conceptual side view with skaters 1 and 2 labeled. The gray skater is the 'ghost skater,' representing where we would see the skater on a glass cube if we had such a cube. The green lines drawn between the origin and the gray skater's outreached arm and foot connect with the corresponding points of the distorted blue skater (labeled 2) seen in the visible face and then intersect with the picture frame, which appears as a

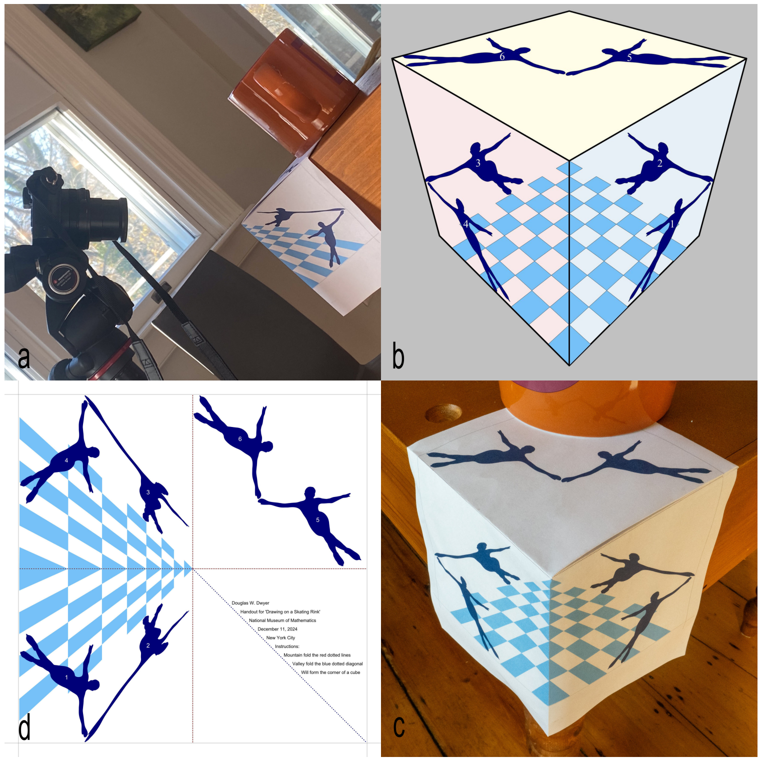

Figure 3: *An origami application: (a) photograph of side view, (b) conceptual camera view, (c) origami model folded, (d) origami model before folding (clockwise from upper left)*

line segment from this orientation. Figure 3(a) shows the corresponding side view taken of the actual piece and the camera used to take the photo in Figure 3(c). Our intent was to place the camera at the correct vantage point by aligning the lines of the chessboard when looking through the camera, but it appears we placed the camera a bit too far back and a bit too high. Nevertheless, the illusion is successful: the chessboard looks like a chessboard and the skaters are defining the other faces of the cube.

Once we have the drawings on each of the visible faces in *camera space*, we need to transform them onto the flat plane of the paper so that they can be folded as intended. The drawings can be converted back to cube space by rotating the cube such that the front face becomes orthogonal to the picture frame. Once in cube space, we need to map the drawings onto the actual piece of paper. Specifically, we map the top face to the upper-right square of the paper, the left side face to the upper-left square of the paper (rotated 90 degrees clockwise). Finally, we map the right face to the lower-left square of the paper (also rotated 90 degrees clockwise). As the lower-right square of the paper is between the folds, it disappears when folded and is a convenient place to write a logo and instructions.

For applications like this, it is convenient to have a matrix that facilitates going from cube space to camera space and vice-versa. We convert from cube space to camera space with a matrix that rotates around the vertical axis 45 degrees and then rotates around the horizontal axis by the arctan $\left(1/\sqrt{2}\right)$. Finally, we push forward each point so the near corner of the cube coincides with the center of the picture plane—the point $\{0, 0, 1\}$:

$$X_{camera} = X_{cube} Iso + \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

where X_{camera} and X_{cube} are matrices representing the points of the drawings in different spaces. Each row corresponds to a data point and the columns are the respective coordinates. Iso is a 3×3 matrix, where the columns are the vectors that transform the cube into camera space:

$$Iso = \begin{bmatrix} 0.707 & 0.408 & 0.577 \\ 0.000 & 0.816 & -0.577 \\ -0.707 & 0.408 & 0.577 \end{bmatrix}$$

After the matrix multiplication, we push the cube back so that its near corner sits on the picture plane, achieving the side-view orientation of the cube shown in Figure 2. Once we solve for the location of the projection of chessboard and the silhouettes on the visible faces of the cube in camera space, we transform them back into cube space by shifting the axes such that the near corner of the cube is at the origin, and left-multiplying by the inverse of the rotation matrix, Iso.

Alternative Approaches

There are tutorials available on YouTube that provide instruction on how to create anamorphic images (e.g., "Understanding Anamorphic Illusions" by Skill-Lync and "How to Create an Anamorphic Illusion" by DeepVizion). The simplest approach is perhaps to trace a shadow of an object and then draw the object within the outline of its shadow, which goes back to at least Leonardo. When viewed from the vantage point of the light that produced the shadow the illusion will emerge. Another approach involves placing a grid on a drawing and then making a so-called *perspective grid* on another sheet of paper and drawing the image on the perspective grid matching square by square. When viewed from the correct viewpoint the illusion emerges.

For anamorphisms that involve an image projected onto the corner of a room (or some other multi-faceted surface), one could project the image (or a grid) onto to a corner using a projector (or a shadow of a grid) and then draw the image using either the projection itself or drawing the image using the projected grid matching square by square. Alternatively, one could determine how to construct a different perspective grid for each face and then create a corresponding non distorted grid on the image itself and match square by square.

Another approach involves taking a picture of an object on top of a piece of paper such that in the picture the object lies within the boundaries of the paper. In this picture, the rectangle of the paper will be distorted. The next step is to construct a new image using software (e.g., Photoshop or Procreate) that is the size of the paper in the actual image and import the picture and transform the picture to align the edges of the paper with the edges of the new image. When printed and viewed from the same viewpoint as the camera the object will look correct, but when looked at head on the object will look distorted.

The problem of projecting a movie onto a curved surface using multiple projectors is well studied, and commercial solutions are available (see, for example, [5]). We believe that the core idea is the same: one starts with a "high-resolution flat movie" and for each pixel on the curved surface one forms a ray between the pixel and the vantage point and solves for the intersection with a *conceptual picture frame* and then map it to the closest corresponding pixel on the conceptual picture frame to determine what color to make the pixel on the curved screen for each frame of the movie.

A celebrated early anamorphism on a curved surface is the vaulted ceiling of the Church of San Ignazio Di Loyola in Rome painted by Fra Andrea Pozzo in the late 1600s. According to [6], this painting was constructed by suspending a *taut net* just beneath the ceiling. For each intersection of the net, a taut line was run from the ceiling through the intersection to a fixed point below on the floor of the church and marking the ceiling. They made a corresponding grid on the preliminary drawing of the painting and mapped the drawing of the painting to the ceiling square to distorted square.

Matt Pritchard made an animation that creates an illusion of a toy car going through a brick wall, and [4] describes the award-winning animation. The effect was achieved by filming an actual moving toy car and a fixed anamorphic drawing of a brick wall.

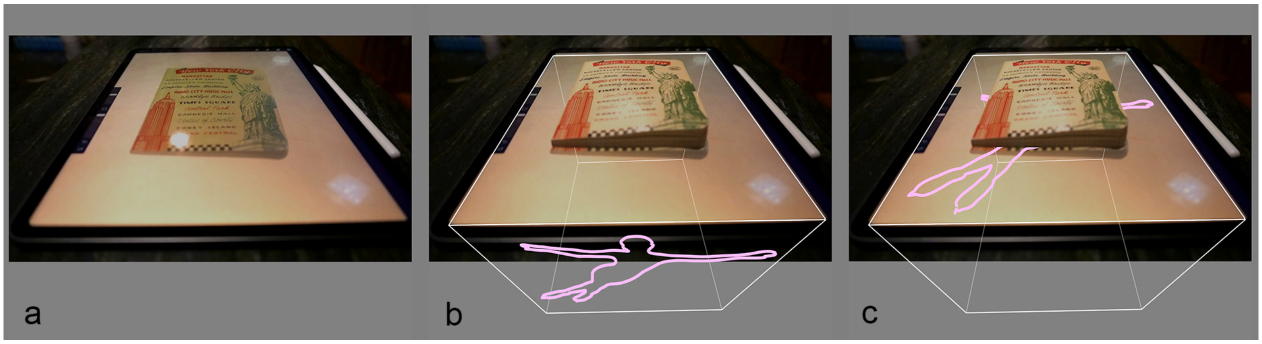

Figure 4: *Frames from an animation of an iPad perspective box: (a) beginning, (b) middle, and (c) end*

The mathematics used by all these approaches are in a sense that same, but the toolboxes are different. In this paper we are using vector algebra, a computer, and software used by data scientists. The other *do it yourself* approaches use rulers, compasses and projectors or software used by graphic designers. For our taste, using vector algebra is potentially time saving if one wants to animate an anamorphism, work directly with line drawings and project onto surfaces with many faces.

Second Application

Figure 4 presents three stills from one of our animations that applies a combination of techniques. The animation relies on three photographs taken from a fixed vantage point of an iPad that is also in a fixed position. The first photograph is a picture of an iPad with a notebook on top of it. The notebook is removed and the photograph is transferred to the iPad and then distorted so that the edges of the iPad in the photograph coincide with the edges of the iPad itself. Without moving either the camera or the iPad, another photograph is taken (Figure 4(a)). This approach is comparable to the YouTube technique for creating an anamorphism described in the last section. The key difference is rather than printing out the distorted image, we are viewing it on the iPad. In Figures 4(b) and 4(c), the notebook has been returned to the iPad and positioned so that the image of the notebook on the iPad reads as the reflection of the real notebook above it. In the animation, we draw a skater on the surface of the iPad before the top notebook has appeared and then draw a box below the iPad. The skater then moves across the faces of the box and returns to her starting point—by which time the top notebook has appeared, and she appears to slide between the two notebooks. The real notebook fades away and the drawings are drawn in reverse to disappear and return to the Figure 4(a), after which the animation starts over in an endless loop (the gallery supplement provides a link to the actual animation).

We determine *iPad space* using an adaptation of the methodology described in [3]. We locate the 4 corners of the iPad using the locate function in *R* and then solve for the vanishing point above the iPad. From our knowledge of the camera used to take the picture, we calculate the angle of view of to be 72 degrees (it was taken with a zoom lens at a 24mm setting using the 35mm equivalent standard). The angle of view combined with the vertical coordinate of the vanishing point implies that the slope of the iPad relative to the picture plane was 0.943. This allows us to compute *left-right*, *up-down* and *back-front* vectors. For convenience, we choose to place the intersection of the ray from the vantage point through the center of the picture frame with the iPad at the point $\{0, 0, 1\}$ in camera space. These vectors, along with the starting point, facilitate placing the figures in each of the faces of the iPad box and projecting them onto the picture frame.

The lines of the box and the skater are drawn on-top of Figures 4(b) and 4(c) in *R* using the functions *plot, lines* and *rasterImage*. The second notebook is placed on top of the resulting image with use of the *Magick* package once again in *R*. We create the illusion of the skater being positioned between the two notebooks in the Figure 4(c) as follows. We generate an image that is either pure black or pure white and is black where we want the second notebook to appear. We think of this image as *the mask*. We make a composite image of the image without the notebook and the mask using the *image composite* function with a darken operator so that now the pixels for where we want the second notebook to appear are all black and the other pixels are unchanged. We also take the image with the second notebook and mask-out all but the second notebook by combining the negative of the mask with the second photograph using the darkening operator again. We then combine the two images with a lightening operator (take the lighter of the two pixels) to get the desired effect. We can also blend between the images with the *image morph* function. For each second of animation, we output 24 JPEGs with a resolution of 2000×2000 from *R* and then use *Python* to make an *MP4* video with the *cv2 package*. Many tools are available to animate and manipulate photographs along these lines. Our approach is economic because it relies on open-source software, and it facilitates automation as it is code driven.

Conclusion

In this paper, we show how the same basic mathematics have been employed across centuries to create a pleasing illusion in different contexts and different scales. Different artists use different mathematical tools to create such illusions. Here we have shown how to achieve them with vector algebra, which supports both animations and working on irregular, multi-faceted surfaces. A gallery supplement provides additional examples with links to animations.

A future challenge would be to make a modern version of a perspective box. While we have seen many modern examples of anamorphisms, we have not seen something where you look inside a modestly sized box from two different peepholes and everything looks correct from both. Another challenge would be to make an *anamorphic home movie* by projecting a film onto a multi-faceted physical surface–the corner of one's living room, for example. Another possibility would be an anamorphic animation utilizing multiple screens showing different sections of the same image that read correctly from a single viewpoint: one could create the illusion of a ghost moving about a room that is only visible through the screens in the room. Finally, one can extend the mathematics to incorporate both shadows and reflections into the imagery.

Acknowledgments

We thank the National Museum of Mathematics in NYC for inviting us to give a *Math Encounters* talk in December of 2024. The ideas in this paper emerged out of preparation for that talk. Additionally, we thank the anonymous reviewers for the helpful and detailed comments.

References

[1] D. Bomford. "Perspective and Peepshow Construction", *National Gallery Technical Bulletin*, vol 11, 1987.

[2] D. Dwyer. "Drawing with Statistics", *Bridges Conference Proceedings*, Halifax, Nova Scotia, Canada, July 27-31, 2023, pp. 157–165. http://archive.bridgesmathart.org/2023/bridges2023-157.html.

[3] D. Dwyer. "Drawing on a Skating Rink", *Bridges Conference Proceedings*, Richmond, Virginia, USA, August 1-5 2024, pp. 91-98. https://archive.bridgesmathart.org/2024/bridges2024-91.html#gsc.tab=0.

[4] M. Parker. *Love Triangle: How Trigonometry Shapes the World*, Riverhead Books, 2024.

[5] R. Raskar, J. van Baar, S. Rao, and T. Willwacher, "Multi-Projector Imagery on Curved Surfaces", Mitsubishi Electric Research Labs, 2001.

[6] E. Renner. *Pinhole Photography: Rediscovering a Historic Technique*, Focal Press, 2nd edition, 1999.