CS280 Final Project
Art Through Transformation

Crystal Yan
crystalzyan@berkeley.edu

Dennis Lee
dennisl88@berkeley.edu

Alvin Zhang
alvinz@berkeley.edu

Abstract

Optical illusion art is an impactful form of art that takes advantage of the structural processes that formulate human vision to cause viewers to question their own perception. Through classical computer vision techniques, we explore ways of computerizing the creation of artistic illusions in the form of 3D perspective illustrations and wireframe sculptures. We reverse the typical inputs and outputs of a homography projection to calculate the original image on a plane or planes non-orthogonal to a viewer, such that the homography results in the desired perspective illusion. We also extend projection to the non-planar case, minimizing reprojection error to generate 3D point clouds that form different wireframe illustrations at specified viewer perspectives.

1. Motivation

Humans have long been fascinated by optical illusions [10, 6], perhaps motivated by curiosity about the gap between perception and reality. One particular facet of visual perception, perspective, has been known since the time of the Ancient Greeks [4, 6], and experienced a surge in popularity during the Renaissance, with significant study on forced perspective and creating illusions of depth [6].

A public, easily accessible means of artistic expression, sidewalk art emerged in the late 19th century, with hundreds of sidewalk artists active in London by 1890. However, the first known instance of forced perspective in sidewalk art was not until 1980, when ex-NASA scientist Kurt Wenner created the first large scale 3D street art [11]. Part of the reason for this long delay is the difficulty for the artist in creating 3D street art on such a scale.

The most common methods in use currently include overlaying a grid of the correct shape over the target image and approximating the shape in each cell, or projecting an image of the art at an angle with the ground [5]. Both of these methods are subject to error, and are difficult for a novice artist to perform. As a result, there are relatively few 3D street artists in the world.

Figure 1. A 3D illusion, by Leon Keer, being drawn on a sidewalk.

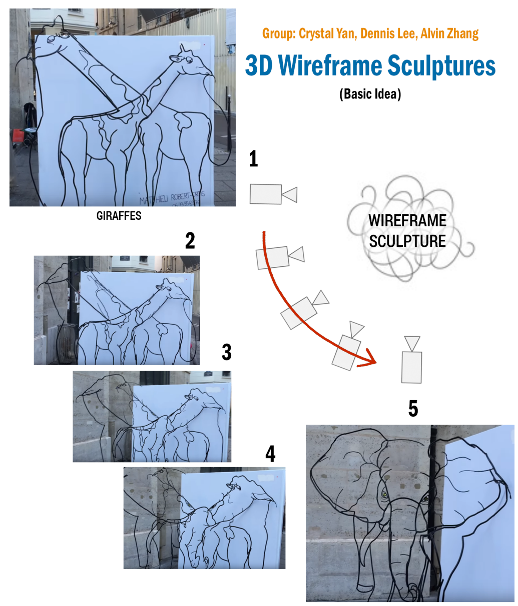

Figure 2. *Elphant-Girafes*, by Matthieu Robert-Ortis.

However, by using computers and linear algebra [7], it is simple to provide an artist with the exact image they need to draw, making it possible for novice artists and even electronic printers to create these 3D illusions. This method can be extended to objects with multiple surfaces at different angles, allowing for the creation of new types of art.

This idea has been further developed by the artist Matthieu Robert-Ortis [8], who has created 3D wireframe sculptures which appear to be different images when viewed from orthogonal angles (Figure 2). In our work, we extend this idea to arbitrary viewpoints and demonstrate our results on 3D-printed structures.

2. Methodology

Python code for our project can be found at: https://github.com/dennisl88/homography_art.

2.1. Street Art

To facilitate the creation of 3D street art, we perform the following steps:

First, we select the corners of the target area, where the image will be placed (Figure 4).

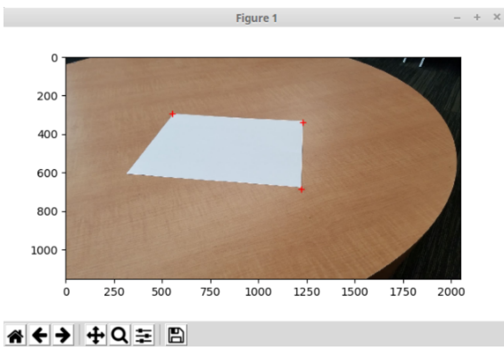

Figure 4. Corner selection for the target area.

Next, we input the actual dimensions of the target area, either as a length and width or corner by corner.

Then, we select the area of the source image which will be drawn over the target area, which will have the same shape (Figure 5).

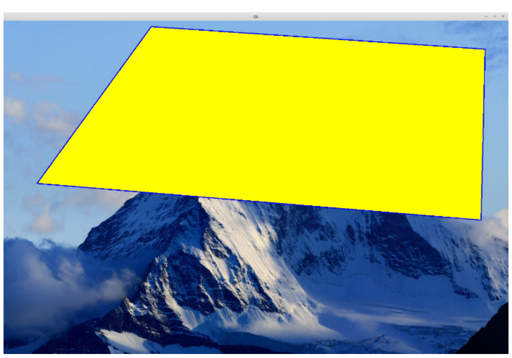

Figure 5. Selecting a region of the source image.

We next compute the homography matrix H from the coordinates in the target shape (x, y) to the coordinates in the source image (x', y'), using the following equation:

$$\begin{bmatrix} x' * z' \\ y' * z' \\ z' \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

This is equivalent to computing

$$\boldsymbol{PH} = \begin{bmatrix} -x_1 & -y_1 & -1 & 0 & 0 & 0 & x_1x_1' & y_1x_1' & x_1' \\ 0 & 0 & 0 & -x_1 & -y_1 & -1 & x_1y_1' & y_1y_1' & y_1' \\ -x_2 & -y_2 & -1 & 0 & 0 & 0 & x_2x_2' & y_2x_2' & x_2' \\ 0 & 0 & 0 & -x_2 & -y_2 & -1 & x_2y_2' & y_2y_2' & y_2' \\ -x_3 & -y_3 & -1 & 0 & 0 & 0 & x_3x_3' & y_3x_3' & x_3' \\ 0 & 0 & 0 & -x_3 & -y_3 & -1 & x_3y_3' & y_3y_3' & y_3' \\ -x_4 & -y_4 & -1 & 0 & 0 & 0 & x_4x_4' & y_4x_4' & x_4' \\ 0 & 0 & 0 & -x_4 & -y_4 & -1 & x_4y_4' & y_4y_4' & y_4' \end{bmatrix} \begin{bmatrix} h1 \\ h2 \\ h3 \\ h4 \\ h5 \\ h6 \\ h7 \\ h8 \\ h9 \end{bmatrix} = 0$$

This equation can be solved using SVD by computing $P = USV^\top$ and taking the eigenvector v corresponding to the smallest eigenvalue.

Finally, we iterate over the coordinates of the target shape, and for each coordinate (x, y), compute the corresponding (x', y') and interpolate within the source image to find the correct pixel. The resulting image (Figure 6) can be drawn by the artist or printed directly. The resulting image can be also be previewed, or compared to the final results. (Figure 7).

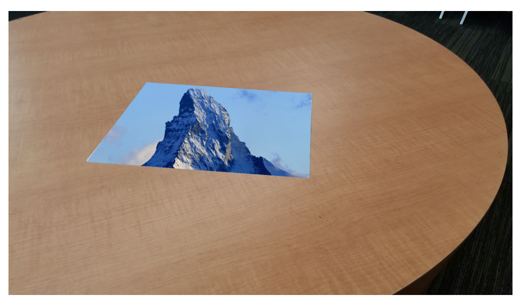

Figure 6. Preview of the applied homography.

Figure 7. Homography result physically printed and put in place

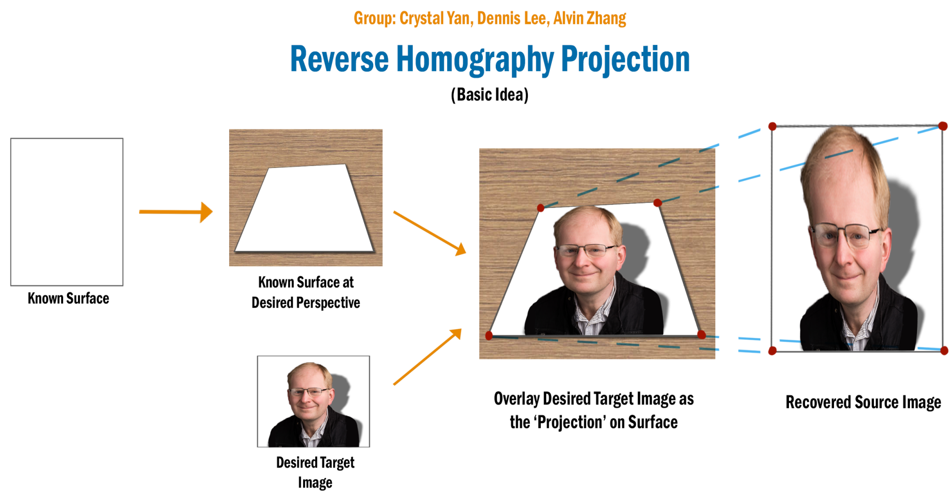

Figure 3. Graphical example of homography

2.2. Wireframes

If we no longer restrict ourselves to the planar case, then it is possible to create illusions which appear to be different images from different viewpoints [3, 9]. Figure 8 depicts the setup.

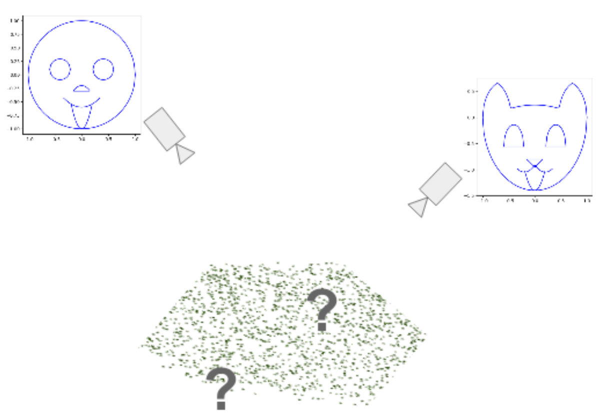

Figure 8. Schematic for the wireframe problem. In this case, we wish to find the point cloud such that the scene as viewed from the left camera appears to be a smiley face, while the scene as viewed from the right camera appears to be a cat.

To accomplish this, we utilize the camera matrices $P_1 = K_1 R_1 [I| - c_1]$ and $P_2 = K_2 R_2 [I| - c_2]$, where K_1 and K_2 are the camera intrinsics matrices, R_1 and R_2 describe the camera orientations in the world frame, and c_1 and c_2 are the camera positions in the world frame. For each pair of corresponding homogenous points $(p_1, p_2) = ((x_1, y_1), (x_2, y_2))$ in the desired images, we want to have $p_1 = P_1 N$ and $p_2 = P_2 N$, where $N = (X, Y, Z, W)$ describes the homogenous 3-D point which is projected to p_1 in camera 1 and to p_2 in camera 2.

Using the projection equations

$$x_1 = \frac{P_1[1,1]X + P_1[1,2]Y + P_1[1,3]Z + P_1[1,4]W}{P_1[3,1]X + P_1[3,2]Y + P_1[3,3]Z + P_1[3,4]W},$$

$$y_1 = \frac{P_1[2,1]X + P_1[2,2]Y + P_1[2,3]Z + P_1[2,4]W}{P_1[3,1]X + P_1[3,2]Y + P_1[3,3]Z + P_1[3,4]W},$$

$$x_2 = \frac{P_2[1,1]X + P_2[1,2]Y + P_2[1,3]Z + P_2[1,4]W}{P_2[3,1]X + P_2[3,2]Y + P_2[3,3]Z + P_2[3,4]W},$$

$$y_2 = \frac{P_2[2,1]X + P_2[2,2]Y + P_2[2,3]Z + P_2[2,4]W}{P_2[3,1]X + P_2[3,2]Y + P_2[3,3]Z + P_2[3,4]W},$$

we can solve for $N = (X, Y, Z, W)$ as the last right eigenvector of

$$M = \begin{bmatrix} x_1 P_1[3] - P_1[1] \\ y_1 P_1[3] - P_1[2] \\ x_2 P_2[3] - P_2[1] \\ y_2 P_2[3] - P_2[2] \end{bmatrix}$$

.

3. Results

By programming our approaches in Python, we have verified the correctness of our methodology by printing 2D and 3D artistic examples that successfully recreate the desired illusions.

3.1. Homography

To verify the correctness of our street art generation technique, we have created multiple examples of projecting illusions onto a known simple surface of an 8.5 x 11 inch sheet of paper. We took input photos of the paper from portrait, landscape, and slanted orientations relative to a flat table surface, and successfully projected artistic graphics of our own design onto each paper orientation (Figures 9, 10, 11). In homage to both computer vision and traditional art subjects, our examples depict the following: (1) a 3D shaded sphere that is often rendered by artists to learn lighting techniques, (2) a word rendered not in edges but in splotches akin to the famous hidden dalmatian illusion, and (3) a recognizable human figure popping out of an illusory hole in the original surface material.

Figure 9. Homography results of our custom-designed image of a 3D shaded sphere, projected at a portrait orientation.

Figure 10. Homography results of our custom-designed image of a word defined by splotches, projected at a landscape orientation.

Figure 11. Homography results of our custom-designed image of a hole rendered in the original surface, projected at a slanted orientation.

We expect the results to hold true to larger surfaces, such as sidewalks and human-sized surfaces. Interestingly, when using human observation on our 8.5 x 11 inch examples from short distances (less than 1 feet away), the illusory sense of perspective were best observed only with one eye open at a time. At such short distances, disparity between the two observer eyes was large enough to require different projection parameters for each eye. This effect disappeared when observing the illusion through a camera lens rather than through human eyesight. We expect this interfering effect to become negligible at the larger viewer distances typical in an outdoor street setting.

We also decided to extend our street art homography to multi-planar surfaces. As our homography technique is capable of projecting over arbitrary rectangular surfaces at arbitrary normal vector orientations, it is feasible for an artist to render a consistent illusion over multiple planes by specifying the 4 corners of every planar segment of the image. This has the implication that our technique can actually be applied consistently over complex 3D mesh or voxel surfaces, so long as the mesh consists of rectangular faces with known dimensions (Figure 12).

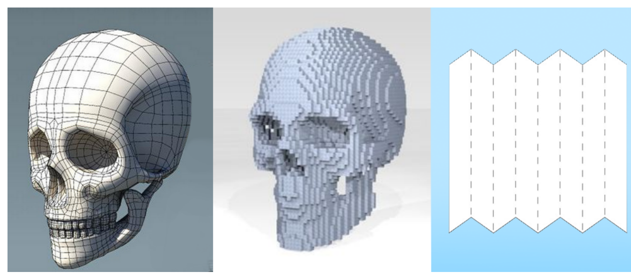

Figure 12. Complex multi-planar surfaces such as 3D meshes (left), voxels (middle), and lenticulars (right) are all candidates for homography projection.

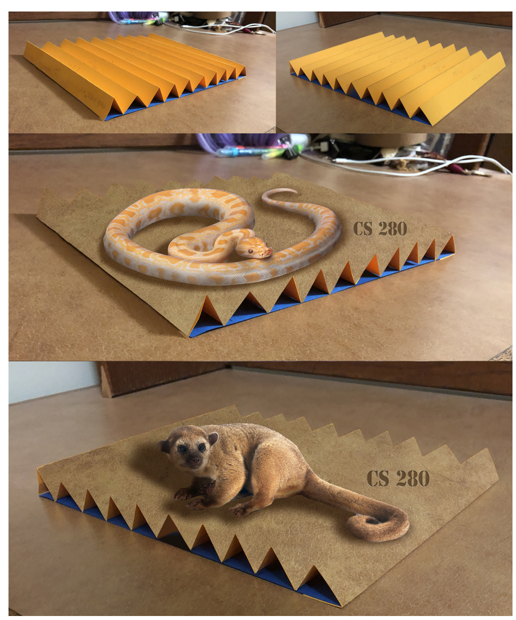

Figure 13. The original photo, and homography results, of our hand-crafted lenticular with custom-designed snake and mongoose images.

Figure 14. The interlaced projection image, and final hand-crafted lenticular, of our custom-designed snake and mongoose images.

In demonstration of this capability, we create two homography projection examples requiring multi-planar projection: a lenticular surface, and a non-continuous geometric object.

Our lenticular surface is not a flat image viewed through a ribbed lenticular lens, but an actual corrugated, accordion shape composed of twenty 8.5 x 1 inch rectangles at equal angles (Figure 13). Additionally, the lenticular surface is intended to be viewed not head-on, but on a flat horizontal surface to give an illusion of either a 3D snake or mongoose laying atop of the table.

At this relatively large a fold interval (1 inch), merely interleaving the two images would not achieve the desired illusion. Each co-normal plane is sufficiently offset along the orthogonal or depth direction that an unrectified interleaved image would not be perceived as continuous. This is evident in how the snake and mongoose planar segments still do not align in the final homography result, even when each is grouped back together into the same original image (Figure 14). Homography projection is thus still required in order to successfully generate this illusion.

Our second multi-planar example is of a geometric shape viewed at such an angle that the illusion image projects across a total of three different orthogonal normal directions, each normal surface being non-continuous. For simplicity's sake, the shape was constructed entirely out of 2 inch x 2 inch squares, more similar to a 3D voxel than a mesh. The illusion image was chosen such that when the shape was viewed while held in the palm of the hand, the viewer's hand would be seamlessly replaced by the illusion hand.

Figure 15. Original photo and homography results of a hand illustration projected over a geometric shape, held in the palm of the hand.

Noticeably, due to the homographic projection across three different normals, parts of the illusion that cross over different normals (such as the thumb) become greatly distorted as they cross the edge. Additionally, due to the orthogonal offset of the non-continuous planes sharing each normal direction, the same points in the image (such as the wrist) map to multiple points on the geometric shape when projected. Despite these non-linearities, the illusion is successfully achieved when constructed in real life.

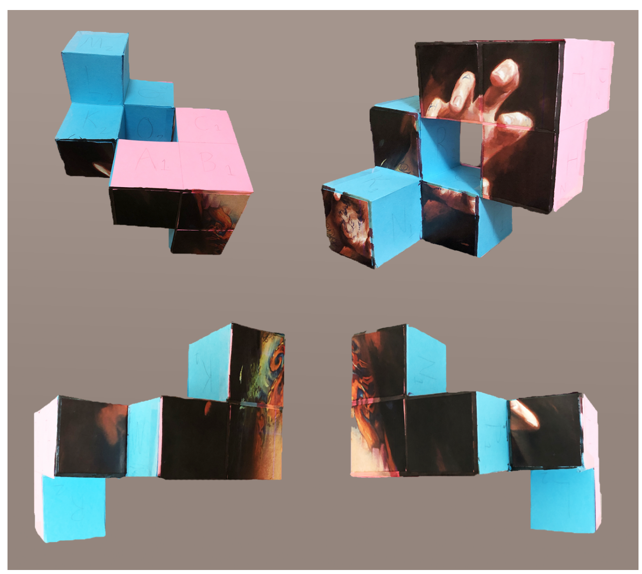

Figure 16. Different views of the final geometric shape illusion.

3.2. Wireframes

For all of our wireframe experiments, we choose $K = \text{diag}(35, 35, 1)$ for our camera intrinsics matrix. In order to facilitate visualization of a real-world 3D-print of our wireframes, we set the cameras' azimuths to $+/-90^o$, with an elevation of 30^o and a distance of 70 to the origin. We create illusions for two pairs of images: a smiley face/cat, and a demon/angel, as depicted in Figure 18. The results can be

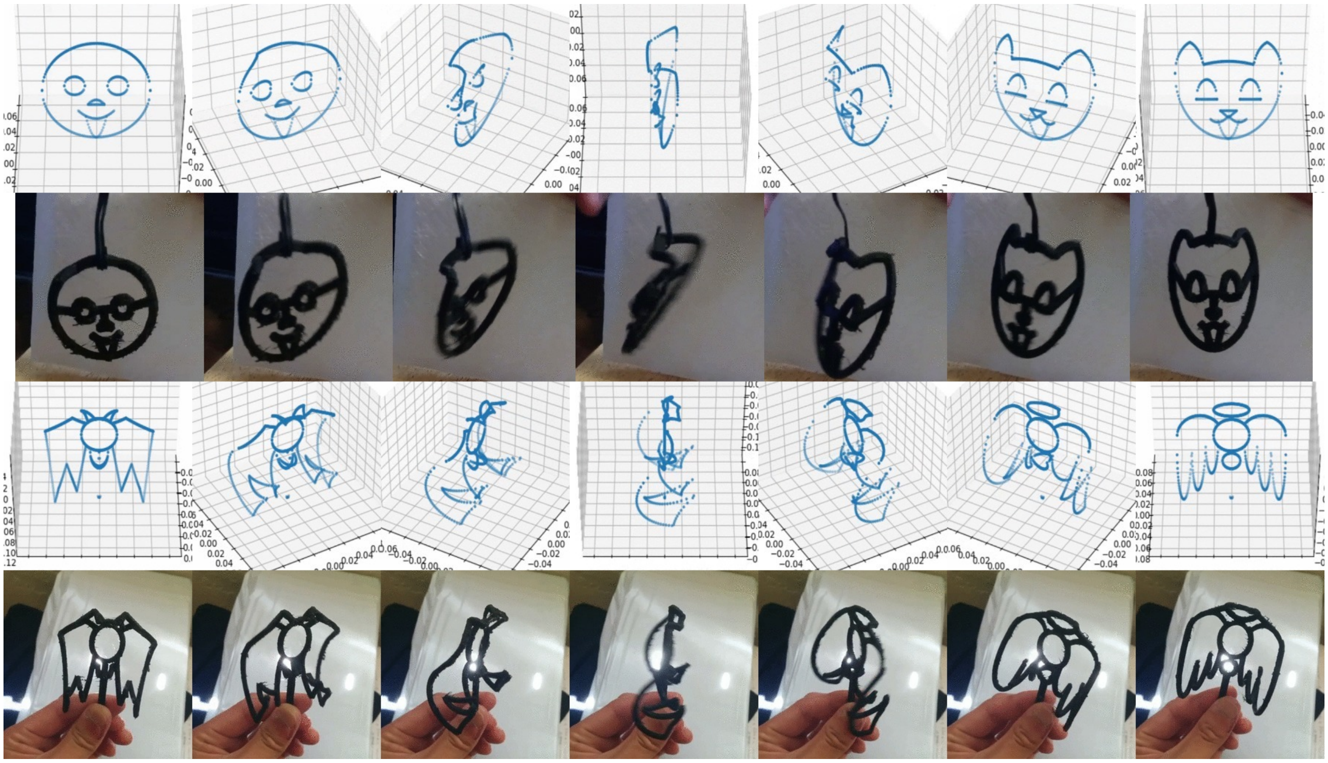

Figure 17. Top-to-bottom: Generated point cloud for the smiley face/cat wireframe illusion; 3D-printed result for the smiley face/cat wireframe illusion; Generated point cloud for the demon/angel wireframe illusion; 3D-printed result for the demon/angel wireframe illusion.

viewed in Figure 17.

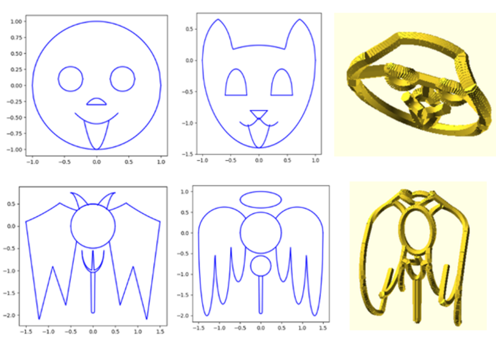

Figure 18. Left two columns: Target images for the smiley face/cat illusion and the demon/angel illusion. Rightmost column: Generated .stl files for 3D-printing the wireframes.

4. Future Work

In terms of artistic projects, one possible idea would be to approximate curved surfaces as the union of many planes. Utilizing the mesh/multiplanar approach exhibited in 3, we can apply our homography to each of these approximated planes [2]. For example, a beach ball could depict a game of chess when viewed from the proper angle.

Another artistically appealing approach would be to further experiment with wireframe designs and camera locations: it may be possible to have a wireframe which could be viewed as three or more different images, depending on the viewing angle. Figure 19 shows that the smiley face/cat illusion already appears as a bird when viewed from the correct angle, providing evidence for the feasibility of this idea.

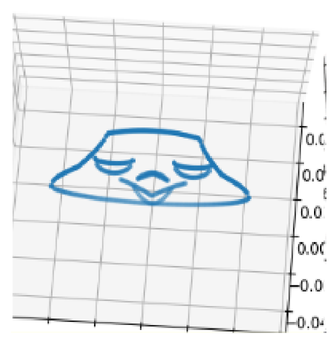

Figure 19. The smiley face/cat wireframe illusion appears as a bird when viwed from above.

But one disparate artistic technique that we did not pursue was the creation of uniformly lit illusions over non-uniformly lit and contoured surfaces (Figure 20). By assuming time-invariant illumination conditions and Lambertian reflectance over the projected surfaces, it would be possible to make 3D perspective illusions over multiplanar and/or multi-albedo surfaces that are consistent over illumination as well as perspective when seen by the viewer.

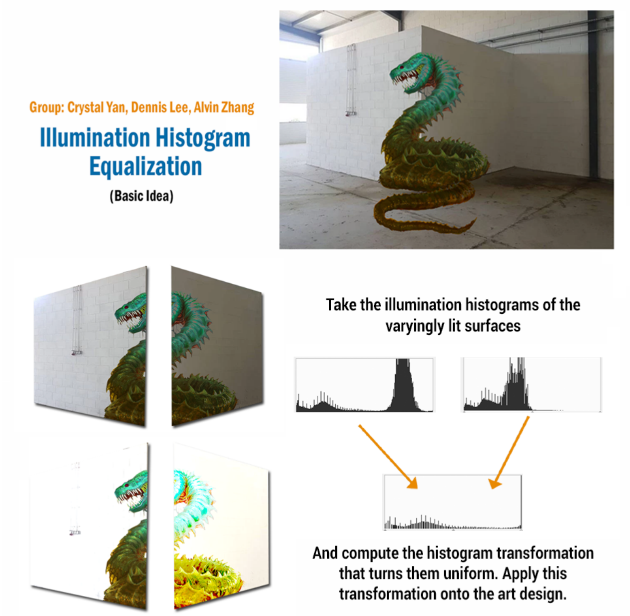

Figure 20. One artistic technique for future work is to generate consistently-illuminated illusion images over non-consistent illuminated surfaces.

The process to compute the necessary illusion is as follows. The artist first takes a photo of the blank surfaces under the expected illumination conditions, and provides 2 inputs: the desired perceived illusion image (i.e. the serpent graphic), and the photo (i.e. the blank walls). The artist would then have to specify the blank surfaces in the photo by selecting their outlines or corners. By computing the required image processing transformation that maps each blank surface under the original lighting conditions all to a uniform lighting condition, one can then apply this transformation to each segment of the illusion to compensate for the varying lighting on each surface when seen by the viewer.

There are two different methods for implementing the transformation for achieving uniform illumination in the perceived image: contrast stretching, and histogram equalization (Figure 21). Both methods, although intended for achieving high contrast in an image, achieve the desired effect of mapping the histogram distribution of illumination or intensity values in an image to a more uniform distribution [1].

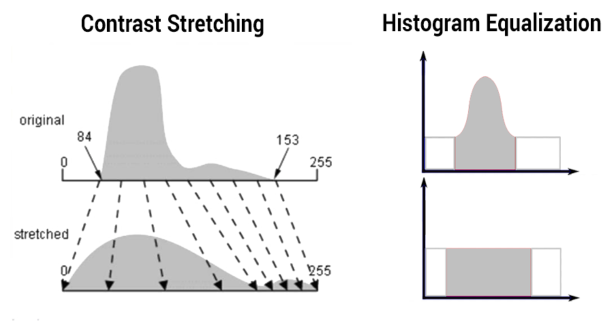

Figure 21. Although both contrast stretching and histogram equalization achieve uniform illumination, only contrast stretching preserves intensity histogram shape.

Contrast stretching is a linear process that takes the minimum and maximum intensity values in the image and maps them out to the ends of a full illumination range $[0, 1]$. The intermediary values are then stretched or spread evenly across this range. Histogram equalization, on the other hand, is a nonlinear process that attempts to normalize the probability distribution by flattening value ranges of high frequency, modifying the shape of the final intensity histogram. Both these techniques can be tested to see which technique provides superior results.

However, there are a few practical concerns to the implementation of this artistic idea. First, the assumption of time-invariant illumination conditions is rarely satisfied in the real world, especially in a street art setting, unless the artwork is to be displayed in a controlled setting environment such as a museum exhibition. Second, the application technique of the illusion itself, such as the paint or the printing technique used, is subject to its own unaccounted properties. While any glossy (and thus non-Lambertian) sheen in the final applied paint or ink may be corrected through an application of clear matte (Lambertian) finish, most application techniques, such as non-opaque paints and inkjet printing, are incapable of achieving lighter tones than the original color of the surface: they are only capable of dying the existing surface darker. This limitation would restrict the lightest value in the final illusion image to the darkest perceived value found across all surfaces.

Due to these limitations, it would be prudent to first test this idea in a controlled setting with a simple multi-planar shape, such as the lenticular shape used in Figures 13 and 14. In this setting, we can control the location and properties of the light source and therefore the theoretical illumination intensity distribution for each of the planes. This would allow us to determine what modifications, if any, should be made to our approach to facilitate real-world illusions.

References

[1] S. S. Al-amri, N. Kalyankar, and S. Khamitkar. Linear and non-linear contrast enhancement image. *International Journal of Computer Science and Network Security*, 10(2):139–143, Feb 2010.

[2] M. J. Brooks and B. K. Horn. Shape and source from shading. In *Proc. Int'l Joint Conf. Artificial Intelligence*, pages 932–936, Aug 1985.

[3] O. D. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Jan 1993.

[4] M. Hogenboom. How your eyes trick your mind. *BBC News*, Jan 2015.

[5] J. Jones. 3d street art: a question of perspective. *The Guardian*, Feb 2012.

[6] L. Lienau. A brief history of perspective. *THE CONSERVATORY FOR CLASSICAL ART*, Apr 2013.

[7] C. Loop and Z. Zhang. Computing rectifying homographies for stereo vision. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1999.

[8] M. Robert-Ortis. Matthieu robert-ortis.

[9] S. Seitz. Image-based transformation of viewpoint and scene appearance. Master's thesis, University of Wisconsin, 1997.

[10] N. J. Wade. *Early History of Illusions*. Oxford University Press, Dec 2017.

[11] K. Wenner. The origins of pavement art, Apr 2015.