A Fresh Perspective

Karan Singh

Department of Computer Science
University of Toronto

Abstract

"Painting is an activity, and the artist will therefore tend to see what he paints rather than to paint what he sees."
E.H. Gombrich.

While general trends in computer graphics continue to drive towards more *photorealistic* imagery, increasing attention is also being devoted to painterly renderings of computer generated scenes. Whereas artists using traditional media almost always deviate from the confines of a precise linear perspective view, digital artists struggle to transcend the standard pin-hole camera model in generating an envisioned image of a three dimensional scene. More specifically, a key limitation of existing camera models is that they inhibit the artistic exploration and understanding of a subject, which is essential for expressing it successfully. Past experiments with non-linear perspectives have primarily focused on abstract mathematical camera models for raytracing, which are both non-interactive and provide the artist with little control over seeing what he wants to see. We address this limitation with a cohesive, interactive approach for exploring non-linear perspective projections. The approach consists of a new camera model and a toolbox of interactive local and global controls for a number of properties, including regions of interest, distortion, and spatial relationship. Furthermore, the approach is incremental, allowing non-linear perspective views of a scene to be built gradually by blending and compositing multiple linear perspectives. In addition to artistic non-photorealistic rendering, our approach has interesting applications in conceptual design and scientific visualization.

Key words: Non-Photorealistic rendering, Multiprojection, Non-linear Perspective, Camera model.

1 Introduction

Before a user can express a digital 3D scene using a 2D projection, he must obtain a good perception of the scene himself. 2D projections, however, are the only common way a user can currently explore a digital 3D scene. While a linear perspective view certainly constitutes a robust and well understood medium for exploring and visualizing localized regions of an object, the model can be restrictive for the visualization of complex shapes. The user could then simply use many such disjoint exploratory images to express a 3D scene. It is, however, neither spatially efficient nor aesthetically desirable to expect a viewer to percieve detail from multiple images: it takes time and skill to understand such a scene since the eye has to jump between the images and mentally stitch them together. Industrial designers often have to take recourse to turntable animations to display or draw attention to parts of their models, which could have been captured effectively in a single non-linear perspective view.

The goal of this work is to allow artists to explore, understand, and subsequently express 3D shapes in 2D imagery. To motivate this objective, we will consider three pictures utilizing non-linear perspective projections.

Figure 1: Femme nue accroupie (P. Picasso,1959), ©2002 Estate of Pablo Picasso / Artists Rights Society (ARS), New York.

Figure 1 shows a painting by Pablo Picasso, which typifies his style of compositing different views of different parts of a scene into a single projection. It is a perfect example of a scene that can be visually thought of as broken into disjoint parts that are viewed from different linear perspectives and then patched back together. The choice of view directions is entirely at the artist's mercy. Note the region around the hands that provides a continuous transition of perspective from one arm to the other. In contrast, the line down the middle of the face is a discontinuous transition between the two sides of the face.

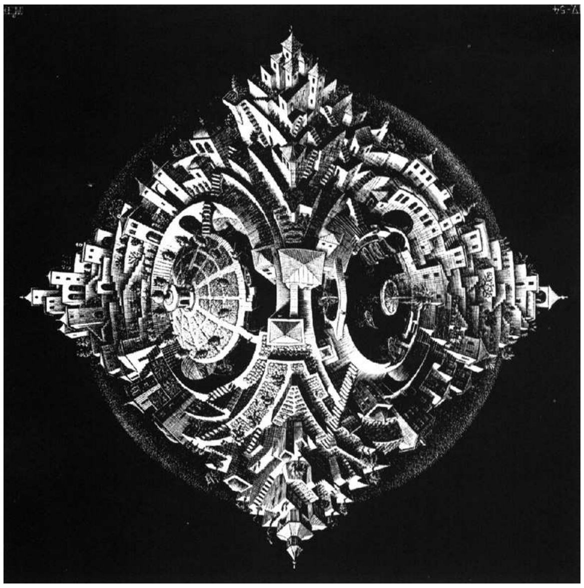


Figure 2: Tetrahedral Planetoid (M.C. Escher), ©2002 Cordon Art B.V. - Baarn - Holland. All rights reserved.

Figure 3: Pearblossom Hwy. No. 2 (D. Hockney 1986), ©1986, D. Hockney, The J. Paul Getty Museum, Los Angeles.

In keeping with the mathematical leanings of M.C. Escher's art, Figure 2 shows a continuous change in perspective across the scene. In contrast to the Picasso painting, the transition between the different projections are gradual and continous, with localized regions of the image adhering to an almost linear perspective.

Figure 3 shows a photomosaic by David Hockney, where many linear perspective views capturing parts of a scene have been patched together with intentional stylistic discontinuity. Hockney has even described Pearblossom Hwy. as "a panoramic assault on Renaissance one-point perspective." Hockney suggests that our actual experience of looking is better expressed by a collage of photographs than by a single image: "If you put six pictures together, you look at them six times. This is more what it's like to look at [something]."

Most visualizations of a scene, scientific or artistic, are compositions of regions that locally have linear perspective[1]. While the three pictures represent very different projective techniques, they all showcase the significance of non-linear perspective projections as an expressive artistic tool. They bring to light the importance of linear perspective as a powerful way to explore and present parts of a scene, as well as the inability of a single linear perspective to visually capture all aspects of a complex scene. Using these images as our guideline, we propose the usage of multiple linear perspective views to explore a scene, as well as the construction of nonlinear perspective visualizations of the scene from these views.

1.1 Related work

Non-linear perspective projections have been applied in computer generated imagery for a variety of purposes, that can be divided into the following main categories: image warping, 3D projections, and multi-perspective panoramas.

Image warping [3, 7, 22] is a popular technique for manipulating digital images. Since this approach is inherently 2D, however, it limits the ability to explore different viewpoints and the spatial relationship between objects. View morphing as presented by Seitz and Dyer [19] attempts to automate the interpolation of a viewpoint in images to provide more natural morphs that have a compelling 3D look to them. The problem of compensating for the distortion introduced by a perspective projection onto a curved object (such as an OMNIMAX screen) [14] or planar screens for off-axis viewing [5] has been researched. Zorin and Barr [24] have developed an approach to correct the perceived geometric distortion seen in many photographic images by reprojection. Research in the area of nonlinear magnification for the purpose of visualization is well documented by Carpendale [4].

As an alternative, 3D deformations [9, 18] are widely used for manipulating 3D geometry. For some applications, however, it is preferable to modify the camera

[1] There are exceptions to this, such as historical Japanese prints done using an inverse linear perspective.

transformation rather than to change the 3D shape of the object being depicted. Barr [2] used non-linear ray tracing to render deformed objects; an idea also applied by Gröller [10]. View dependent distortions to scene geometry for animation and illustration have also been explored [13, 17]. The distortion associated with 3D projections has been utilized for artistic purposes by Inakage [11] and Levene [12]. Inakage developed a library of esoteric projection tools, and this work was extended by Levene with an interactive system for editing a non-linear 3D projection surface and the warped space around it. Further, Levene explored the spatial relationship in a 3D scene by allowing different projections to be associated with individual objects. The main limitations of these approaches, however, are that they rely on a single center of projection, which can lead to severely distorted control lattices (for example if the artist wishes to direct the viewer's attention towards a region of an object far from the center of interest), and that the results of associating different projections with different objects can be unpredictable.

Recently, non-linear projections have also used in conjunction with *multi-perspective* panoramas. Inspired by the compelling illusion of depth in classic Disney animations, Wood et al. [21] generated 2D panoramas for predescribed 3D camera paths, achieving the effect of 3D perspective as the camera panned across the panorama. In a related technique, albeit with a very different motivation, Rademacher et al. [16] generated *multiple-center-of-projection images* by moving a camera along a 3D path, sampling a 1-dimensional slice of a 2D panorama at each point on its way. The aim of this work was to provide a more flexible and efficient representation for image based data sets. Peleg et al. [15] have generalized the creation of mosaic panoramas from camera motion by allowing them to be mapped on to an adaptively changing 2D manifold. Panoramas are an effective way of visualizing landscapes with a wide angle of view, or for unfolding the detail of an object. These approaches are catered to capturing imagery using real cameras. They are also, unfortunately, not well suited to interactive manipulation.

The research of most relevance to this paper is work on abstract camera models by Wyvill and McNaughton [23] and the approach to multiprojection rendering by Agrawala et al. [1].

Wyvill and McNaughton define an abstract camera model using a surface in 3D from which rays emanate in arbitrary directions. While the general model is very powerful and conceptually simple, it is hard to specify and control intuitively by a user and is difficult to implement interactively.

Agrawala et al. [1] use multiple linear perspectives to define a scene. In their approach each object in the scene is assigned to some camera in the scene and rendered based on the linear perspective of that camera. The rendering of all cameras in the scene are composited to generate the final image. A visibility ordering is created for the objects using a master camera and this is used during the compositing stage. Conflicts in a clear visibility ordering of objects are resolved at the pixel level by simple depth comparison. The use of linear perspective to both explore the scene and to construct the nonlinear image makes their system both interactive and easy to use. While the approach works well for scenes with disjoint objects with different linear perspectives, it would be difficult to construct scene visualizations of the kind shown in Figures 2 and 3. The main reasons are the continuous transitions of linear perspective seen in Figure 2 and collage like compositing of camera images in Figure 3. Figure 1 could be generated with the technique if the woman in the scene were carefully segmented into distinct objects that could then be viewed from different linear perspectives and composed to construct the image. Our approach will also use multiple linear perspective cameras but objects in our framework are potentially influenced by all cameras, yielding an interpolated virtual camera whose parameters often vary across different points on the object.

In summary, non-linear projections have been used in a variety of contexts for visualizing 3D shapes, but existing techniques are limited with respect to our goal of allowing artists and scientists alike to intuitively explore, understand, and subsequently express or visualize 3D models in 2D images.

1.2 Overview

The rest of this paper is organized as follows. Section 2 presents our model for the construction of a non-linear perspective projection of objects in a scene using a number of linear perspective cameras. Section 3 describes the implementation details and the user interface framework that is crucial to the success of this model. Section 4 concludes with a discussion of the results obtained and provides directions for future research.

2 Model for non-linear perspective

Let C_i represent the camera parameters[2] [6], for exploratory view $i \in 1,..,n$. Let M_i represent the perspective projection matrix built from the parameters C_i. $< x, y, z >= PM_i$ represents the linear projection of P into canonical space $x \in [-1, 1], y \in [-1, 1], z \in [0, 1]$.

Given a viewport specification represented by matrix V_i the resulting point in two dimensional screen space

[2]Example parameters that specify a linear perspective camera are eye position, center of interest, up-vector, hither/yon clip planes and focal length.

$< x_s, y_s >$ is $< x_s, y_s, z_s >= PM_iV_i$. Usually, $z_s = z$ is the depth value of the point P, unchanged by V_i. We extend the viewport transformations V_i so that the canonical depth of a point $z \in [0, 1]$ is linearly mapped to z_s in an arbitrary, user specified range. While the relative depth values are preserved with respect to a single perspective view, this allows the powerful visual capability of intuitively altering the relative depths of points in a scene as one transitions between the mutiple linear perspectives. It is through the depth mapping that visibility in the nonlinear projection of the scene is controlled, both automatically [1] and with user interaction. This will be illustrated in Section 3.3 (Case Study II).

Now suppose that a normalized weight vector $< w_{1P}, w_{2P}, .., w_{nP} >$ is specified for any point P in the scene. We define the projection of P to be PM_PV_P, where M_P is the perspective projection of a virtual linear perspective camera C_P. The parameters of C_P are obtained as an average of the parameters of cameras $C_1..C_n$, weighted by $w_{1P}..w_{nP}$. Similarly V_P is generated by weight averaging the affine components of viewport transformations $V_1..V_n$ with the weights $w_{1P}..w_{nP}$.

The rationale for generating an interpolated camera and an interpolated viewport independently, rather than simply weight averaging the projected points resulting from applying each linear perspective camera projection to P, is twofold. First and foremost, a number of camera parameters are angular and are best interpolated individually using quaternions. Secondly, the camera parameters have intuitive physical manifestations and their interpolation can be better understood and controlled by a user.

We now have an abstract conceptual model for the generation of a non-linear perspective projection from multiple linear perspective views. One can observe that the multiprojection setup of Agrawala et al. [1] is a special case of this model where:

- The $< x, y >$ viewport transformations of all cameras are the same.
- The $< z >$ mapping for all viewport transformations can be set based on the visibility ordering of a master camera.
- $w_{kP} = 1$, for all points P on an object, where the object is being visualized by the kth camera, C_k.

In the next section we look at techniques and controls that provide a compelling and interactive user interface to the model described above. The generation of non-linear perspective visualizations of a scene from multiple linear exploratory views is thus made fast and easy to control.

3 Implementation

This section describes an implementation written as a plug-in to the animation system *Maya 1.5*. The exploratory linear perpective cameras are rendered using OpenGL. These are composited into a non-linear perspective projection that is calculated and then displayed in an orthographic GL view along with multiple translucent red boxes that represent the multiple viewports in the scene. The scene is lit in perspective space with the surface normal at a point P transformed using the projection matrices computed at P.

3.1 User Interface

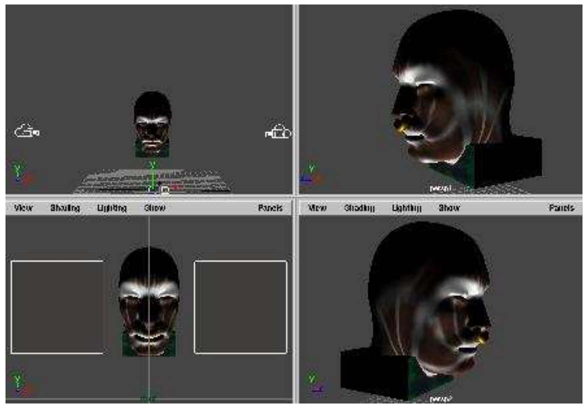


Figure 4: UI Framework (equal camera weight)

The basic user interface framework can be seen in Figure 4. A global linear perspective view (*top-left* panel) shows the object in the scene as well as two exploratory cameras. The views through the exploratory cameras can be seen in the *top-right* and *bottom-right* panels. The *bottom-left* panel shows two square viewports for the two exploratory cameras and the composite non-linear perspective view. The weights are set to be equal for all points in the scene and thus the composite view is a linear perspective projection as viewed through a virtual camera placed halfway between the two exploratory cameras and mapped to a viewport that is right in the middle of the viewports of the exploratory cameras.

Exploratory cameras can be added and deleted at will. Their parameters can be accessed by selecting the camera from its icon in the global perspective view or directly from its own view panel. The camera parameters can be interactively manipulated directly using various camera tools. The viewports are represented by boxes in the composite view (*bottom-left* panel of Figure 4). These can be interactively selected and transformed and the depth of the boxes defines the range to which the cannonical z value of points is mapped.

We now calculate the weight vector $< w_{1P}, .., w_{nP} >$ for a point P in the scene.

3.2 Computing the relative influence of cameras

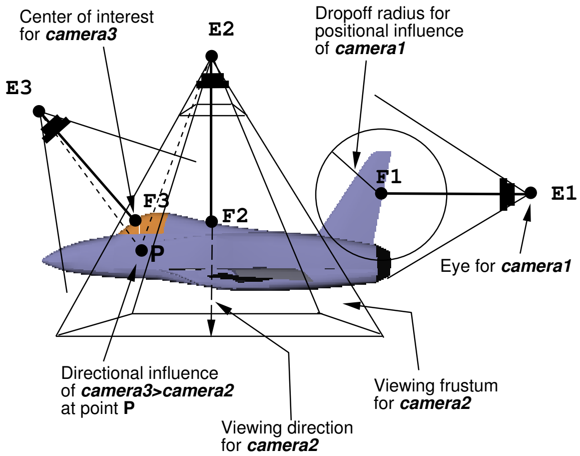


Figure 5: Camera parameters

We use two heuristics that result in smoothly varying weight vectors for points in the scene.

- **Positional influence :** is based on the observation that a local region around the center of interest or focus of an exploratory camera is likely to be visualized using that camera. This is computed as a radial dropoff function around the center of interest; the intensity, radius of influence and decay rate of which are under user control (See Figure 5).

- **Directional influence :** similarly notes that points in the scene that are along the viewing direction of an exploratory camera are more likely to be visualized by that camera than points on the fringes of the camera's viewing frustum [6]. We compute this as a prismatic dropoff function that decays from the view direction axis to the extent of the trapeziodal frustum that is obtained from the viewing parameters of the camera [6] (See Figure 5). Once again the intensity, radius of influence and decay rate of the dropoff function are under user control.

Directional influence provides a number of useful scientific projections like panoramas. Figure 6 shows the same exploratory camera configuration as Figure 4 with directional influence activated. The resulting non-linear projection is a smooth panoramic transition from the view of the *bottom-right* panel to the *top-right* panel as the viewport moves from left to right. Fish eye views can also be obtained using directional influence for a planar

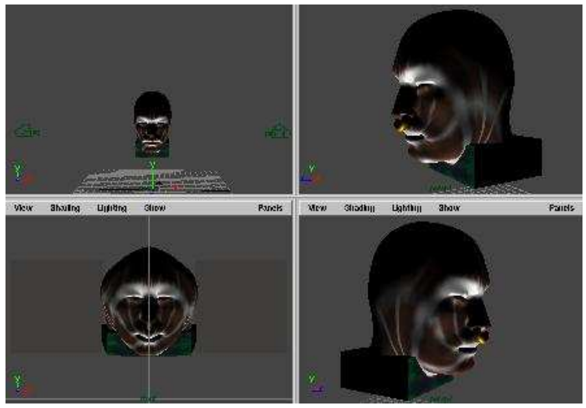
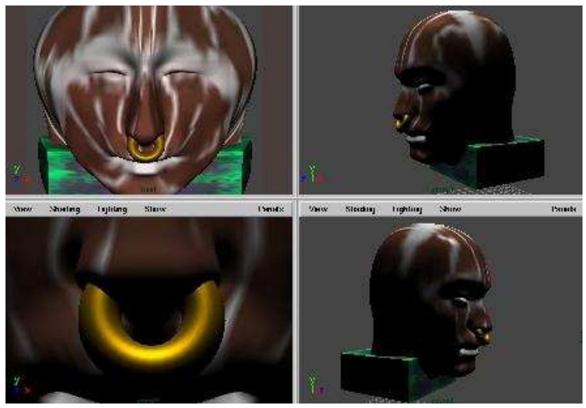
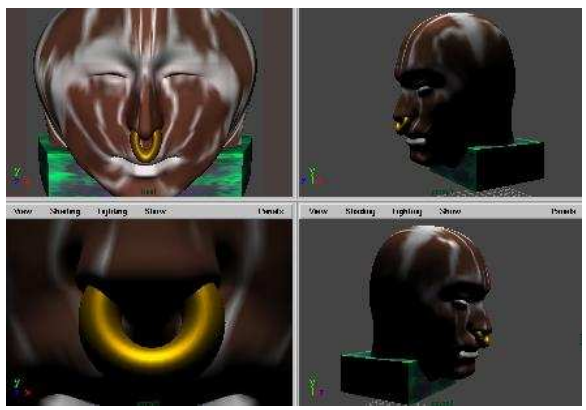


Figure 6: UI Framework (directional influence)



(a) Nose camera active



(b) Nose camera inactive

Figure 7: Refining the visualization (positional influence)

grid of exploratory cameras with parallel viewing directions (to simulate a real lens whose area is the size of the grid). The distorted view of Figure 2 is a strong example of perspective changing from principal exploratory views based on positional and directional influence.

Positional influence provides strong local control and is ideal for incremental local refinements of a non-linear perspective using additional cameras. This is well illustrated in Figure 7, where adding a camera trained on the nose corrects an otherwise flattened nose-ring.

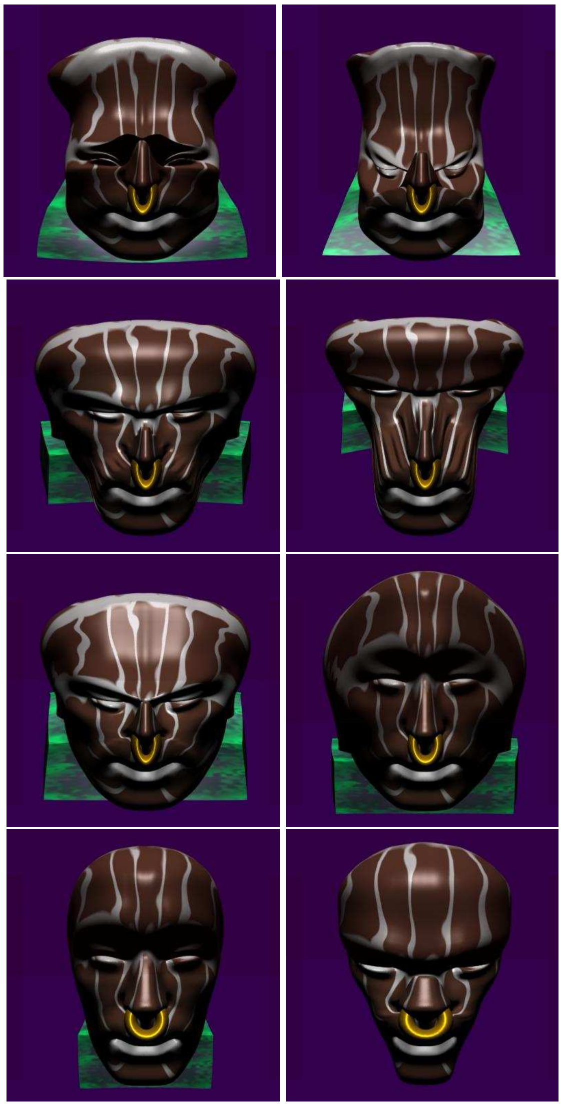


Figure 8: Non-Linear perspective animation stills

In general, however, it is difficult to computationally anticipate an artists intention for more stylistic visualizations (See Figure 1). The user is therefore able to paint weight values for the various exploratory cameras on the objects in the scene directly, using interactive brush tools such as *Maya-Artisan*. User painted weights and other heuristic influences are simply blended together and normalized to generate the weight vector for any given point. Figure 8 has examples of user painted weights applied using a stripe texture to the cheeks that causes the creases above the mouth in the fourth image of the sequence.

For interactivity of the system we only project the control vertices of the objects. While this is an analytically precise operation for linear perspectives and polygon based objects it is clearly a discrete approximation for a non-linear perspective projection. The accuracy of the projection can be simply improved by using subdivided geometry that has a denser sampling of control points. The implementation shows interactive update rates while animating the camera parameters for objects with thousands of polygons.

We now look at two example case studies that illustrate some of the controls described for the generation of a non-linear perspective view.

3.3 Case Studies

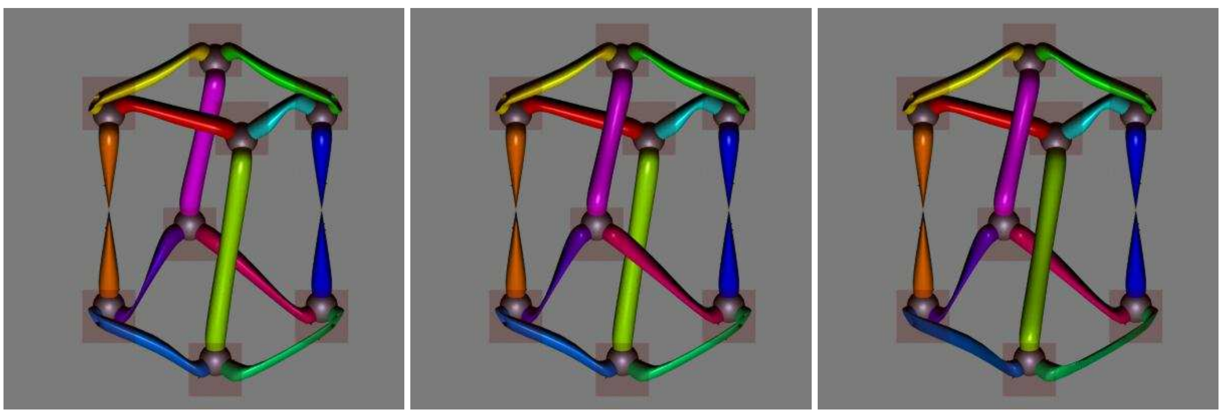
Case Study I : We begin with a two camera panorama of the head in Figure 9a, using a setup like the one shown in Figure 6. The panorama is then stretched in Figure 9b as a result of interactively moving the viewports of the two cameras further apart. In Figure 9c a third camera focused on the nose is added and its viewport stretched horizontally. In Figure 9d this viewport is shrunk horizontally and translated down. Note how the upper lip pouts over the lower lip by reducing the z-depth of the camera on the nose relative to the other two cameras. This can be seen even better in Figure 9e where the viewport for the nose camera is pulled out further and off-center. Figure 9f shows the diversity of visualization that non-linear perspectives bring. A fourth camera is added above the head looking down with a strong directional influence and weak positional influence. This makes most of the head shown as viewed straight down the forehead, except the eyes. Each eye is visualized by one of the side cameras that focuses on them with a strong positional influence. The two ear like projections are regions of the head on the fringes of the viewing frustum of the overhead camera. The side cameras thus have more of an influence on them. The nose is picked up by the camera trained on the nose and the projection of the mouth is a more equal contribution of all 4 cameras. Finally Figure 9g shows the same camera configuration with a lowered intensity of the various influences used.

Case Study II : Figure 10 shows a cube with 8 cameras located at its origin and aimed at the corners of the cube. Figure 11a shows a non-linear perspective projection with the positional influence turned up. The translu-

(a) (b) (c) (d)

(e) (f) (g)

Figure 9: Case Study I



(a) (b) (c)

Figure 11: Case Study II

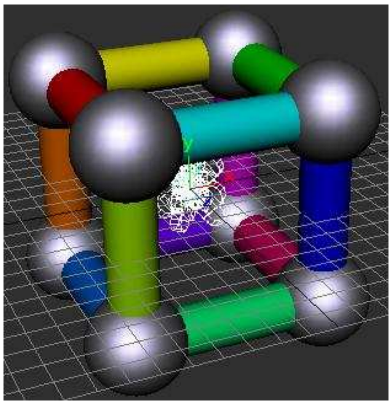


Figure 10: Case Study II: Camera configuration

cent squares are the 8 viewports. Some of them are horizontally scaled by -1 to unravel the object better but are also responsible for the twisted behavior of the orange and royal blue edges. The relative depth of the *purple-magenta-pink* corner's viewport is changed to make the cube projection appear turned inside out in Figure 11b. Note that the silhouette of the edges remains unchanged. Changing the depth on the *green-limegreen-blue* corner's viewport in Figure 11c results in a 2D projection of a 3D projection of a tesseract (a common Escher motif, held by the jester in his work titled Belvedere).

4 Conclusion

We have presented a new interactive approach for exploring and rendering 3D objects. Our chief contribution is an intuitive way for artists to experiment with a 3D subject and subsequently convey it expressively in a 2D rendering. Aside from its applicability to non-photorealistic and painterly rendering, the model has wider applications in scientific visualization, where the limitations of the traditional linear perspective are well recognized [20]. Interaction of illumination models with such non-linear projections is fertile area for future research. In conclusion, the approach presented in this paper marks a step towards overcoming the limited expressive potential of existing projection models. We hope that this work will motivate further discussion and open the door to an interesting new type of computer generated imagery.

References

[1] M. Agrawala, D. Zorin and T. Munzner. Artistic Multiprojection Rendering. *Eurographics Rendering Workshop*, 125–136, 2000.

[2] A. Barr. Ray Tracing Deformed Surfaces. *Computer Graphics*, 20(4):287–296, 1986.

[3] T. Beier and S. Neely. Feature based image metamorphosis. *Computer Graphics*, 26(2):35–42, 1992.

[4] S. Carpendale. A framework for elastic presentation space. *PhD. Dissert.*, *Simon Fraser University*, 1999.

[5] J. Dorsey, F. Sillion and D. Greenberg. Design and Simulation of Opera Lighting and Projection Effects. *Computer Graphics*, 25(4):41–50, 1991.

[6] J. Foley, A. van Dam, S. Feiner, J. Hughes. Computer Graphics, Principles and practice. *Addison-Wesley*, Chapter 6:229–284 ,1990.

[7] C. Fu, T. Wong, P. Heng. Warping panorama correctly with triangles. *Eurographics Rendering Workshop*,1999.

[8] E. H. Gombrich. Art and Illusion. Phaidon, 1960.

[9] J. Gomez, L. Darca, B. Costa and L. Velho. Warping and Morphing of Graphical Objects . *Morgan Kaufmann, San Francisco, California*, 1998.

[10] E. Gröller. Nonlinear ray tracing: visualizing strange worlds. *The Visual Computer*, 11(5):263–276, 1995.

[11] M. Inakage. Non-Linear Perspective Projections. *Proceedings of the IFIP WG 5.10)*, 203–215, 1991.

[12] J. Levene. A Framework for Non-Realistic Projections. *Master of Engineering Thesis, MIT*, 1991.

[13] D. Martin, S. Garcia and J. C. Torres. Observer dependent deformations in illustration. *Non-Photorealistic Animation and Rendering 2000* , Annecy, France, June 5-7, 2000.

[14] N. Max. Computer Graphics Distortion for IMAX and OMNIMAX Projection. *Nicograph '83 Proceedings* 137–159.

[15] S. Peleg, B. Rousso, A. Rav-Acha and A. Zomet. Mosaicing on Adaptive Manifolds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1144–1154, 2000.

[16] P. Rademacher and G. Bishop. Multiple-Center-of-Projection Images. *SIGGRAPH*, 199–206, 1998.

[17] P. Rademacher. View-Dependent Geometry. *Computer Graphics*, 439–446, 1999.

[18] T. Sederberg and S. Parry. Free-form deformation of solid geometric models. *Computer Graphics*, 20:151–160, 1986.

[19] S. Seitz and C. Dyer. View Morphing: Synthesizing 3D Metamorphoses Using Image Transforms. *Computer Graphics*, 21–30, 1996.

[20] Edward Tufte. Envisioning Information. *Graphics Press*, 1990.

[21] D. Wood, A. Finkelstein, J. Hughes, C. Thayer and D. Salesin. Multiperspective Panoramas for Cel Animation. *Computer Graphics*, 243–250, 1997.

[22] George Wolberg. Digital Image Warping. *IEEE Computer Society Press*, 1992.

[23] G. Wyvill and C. McNaughton. Optical models. *Proceedings CGI*, 1990.

[24] D. Zorin and A. Barr. Correction of Geometric Perceptual Distortion in Pictures. *Computer Graphics*, 257–264, 1995.