

# Occlusion-Free Animation of Driving Routes for Car Navigation Systems

Shigeo Takahashi, *Member, IEEE*, Kenichi Yoshida,  
Kenji Shimada, *Member, IEEE*, and Tomoyuki Nishita, *Member, IEEE*

**Abstract**— This paper presents a method for occlusion-free animation of geographical landmarks, and its application to a new type of car navigation system in which driving routes of interest are always visible. This is achieved by animating a nonperspective image where geographical landmarks such as mountain tops and roads are rendered as if they are seen from different viewpoints. The technical contribution of this paper lies in formulating the nonperspective terrain navigation as an inverse problem of continuously deforming a 3D terrain surface from the 2D screen arrangement of its associated geographical landmarks. The present approach provides a perceptually reasonable compromise between the navigation clarity and visual realism where the corresponding nonperspective view is fully augmented by assigning appropriate textures and shading effects to the terrain surface according to its geometry. An eye tracking experiment is conducted to prove that the present approach actually exhibits visually-pleasing navigation frames while users can clearly recognize the shape of the driving route without occlusion, together with the spatial configuration of geographical landmarks in its neighborhood.

**Index Terms**—car navigation systems, nonperspective projection, occlusion-free animation, visual perception, temporal coherence

---

◆

## 1 INTRODUCTION

Car navigation systems guide a car driver through a complicated route in road networks, while displaying geographical information along the route. Today, commercially available car navigation systems can provide a 3D perspective view of a road network as well as its corresponding bird's-eye-view, as shown in Figure 2. Especially for displaying the 3D perspective view, the system generates consecutive navigation frames using *perspective* projection to simulate realistic scenery from a car window. Furthermore, as shown in Figure 1(a), previewing a 3D terrain surface around the route provides a sense of fun for both the car driver and fellow passengers.

Perspective projection is a useful tool to provide animation frames in car navigation systems because it can supply precise perspective to the frames where driving route directions are displayed. However, in our daily experience, such precise perspective is not necessarily the best in conveying intuitive visual information of a 3D scene because it often fails to illuminate the relative positions of geographical landmarks when their corresponding 2D positions overlap on the screen. For example, Figure 1(a) shows a perspective snapshot where the driving route shown in red is occluded by a mountain on the right, which prevents prediction of future routes on the 2D screen. Commercially available systems solve this problem by providing the corresponding bird's-eye-view simultaneously, while this imposes psychological stress on the users because they have to move their eyes more often between the two different subwindows as shown in Figure 2. Although translucent representation of the occluding mountains may resolve this problem, it still cannot handle the case when the route is occluded by bumpy terrain regions multiple times (See Figure 12 for example.). This difficulty justifies the need for combining the 3D perspective view and bird's-eye-view into one coherent image to yield occlusion-free representations of features. Such projection has been realized by simulating human skill in creating artistic pictures and hand-drawn illustrations, where an ordinary perspective image is intentionally distorted so that each feature is projected as if it is seen from a different viewpoint. In this paper, such projection will be re-

ferred to as *nonperspective projection*. Modeling such nonperspective projection is an important issue in computer graphics in general, because it enriches the use of 2D projections, the common visual media for conveying information about 3D scenes.

While a relatively small amount of work has been done on this subject, recent study has made progress in the techniques for distorting perspective projections. In particular, the latest methods have extended the expressive power of such projections – not by distorting 2D perspectives or bending sight rays directly – but by deforming the target 3D objects instead. However, these methods limit the degrees of freedom in the deformation of the target objects because they require us to design the associated distortion in 2D projection indirectly by deforming their 3D shapes. This leads to a tedious trial-and-error process. Moreover, animating such projections with temporal coherence introduces further difficulties to the methods.

This paper presents a method of animating nonperspective projections. The application is a new type of car navigation system in which a driving route remains visible without being occluded by surrounding mountains and valleys. The technical contribution of this paper lies in formulating the nonperspective animation as an inverse problem of finding a deformed 3D terrain surface from the 2D screen arrangement of its associated geographical landmarks. Furthermore, the present approach fully augments the visual realism of a 3D scene in the locally distorted projection by assigning appropriate textures and shading effects on the terrain surface. The method thus offers a perceptually reasonable compromise between the perfect perspective and visual clarity in the 2D projection. Figure 1(b) displays such an example in which the route around the current position is more visible using our nonperspective approach.

Conventional non-perspective projections are assumed to include *rectangular* objects such as buildings, which help us perceive partial perspective in the scenes. However, in that case, we can assign a different viewpoint to each rectangular object, and then camouflage the associated inconsistency between their perspectives on the flat ground. Refer to [1] for an example. On the other hand, this study rather pursues seamless change in perspective over the 2D projection, and thus focuses on depicting *smooth* terrain undulations such as mountain and valleys, rather than city areas with rectangular buildings. Several route navigations in mountain areas are also conducted to demonstrate the effectiveness of the present approach.

The rest of this paper is organized as follows: Section 2 refers to previous work related to this method. Section 3 provides an overview of the car navigation system implemented in this study. Sections 4 to

- 
- S. Takahashi, K. Yoshida, and T. Nishita are with the University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa, Chiba, 227-8561, Japan. E-mail: takahashis@acm.org, kenyoshi@visual.k.u-tokyo.ac.jp, nis@is.s.u-tokyo.ac.jp.
  - K. Shimada is with Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, U.S.A., E-mail: shimada@cmu.edu.

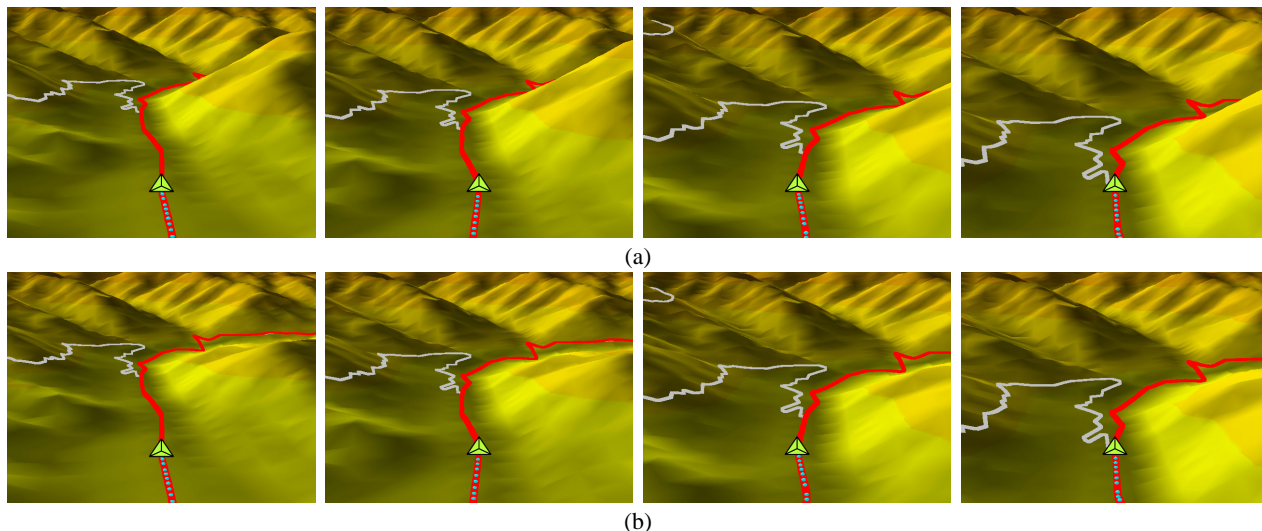


Fig. 1. Animation snapshots in navigating the Takeshi village, Nagano, with (a) ordinary perspective projection and (b) temporary coherent nonperspective projection. The route (in red) is occluded by a mountain on the right in (a) while it is visible in (b).



Fig. 2. An example of a commercially available car navigation system. A 3D perspective view of a road network (left) and its corresponding bird's-eye-view (right).

7 correspond to steps of generating nonperspective animation frames. Section 4 describes a method for extracting geographical landmarks that guide the distortion of the 2D perspective image. Section 5 formulates a heuristic algorithm that calculates an optimal arrangement of the extracted landmarks on the screen in order to avoid route occlusions. Section 6 explains how to deform a 3D terrain surface to satisfy the precomputed optimal arrangement of the landmarks on the 2D screen. Section 7 describes a techniques for preserving temporal coherence in animating nonperspective images. After presenting several route navigation results together with an eye tracking experiment in Section 8, Section 9 concludes this paper and refers to future work.

## 2 RELATED WORK

Compared with photorealistic representations that pursue the physical realism of optical properties, nonphotorealistic representations have been studied rather by taking account of human visual perception and understanding. These representations give rise to a new approach called *nonphotorealistic rendering (NPR)* [6, 18]. However, most non-photorealistic representations are limited to the rendering stage in the graphics pipeline, and a relatively small number of such representations have been applied to the projection stage.

While several 2D image-based methods have been proposed to warp ordinary perspective images, their ability to incorporate different viewpoints is limited. Seitz et al. [15] introduced a method called *view morphing*, which simulates the motion of a virtual camera to interpolate between photographs taken from different viewpoints. Zorin et al. [24] presented approaches for correcting distortions that may exist

in ordinary perspective images such as photographs and pictures.

We can also enrich the expressive power of perspective images by modifying the projection mechanism. *Multiperspective panoramas* by Wood et al. [23] allow us to merge local perspective images seamlessly along a camera path for creating background images for cel animation. Although this is the first to address multiple viewpoints assigned to local features, it still cannot preserve the smoothness of a 3D scene everywhere in the final 2D images. Agrawala et al. [1] presented a method called *artistic multiprojection rendering*, where each 3D object is rendered as seen from its own vista point individually. Nonetheless, the method is not suitable for our purpose because the 3D objects must be disconnected.

Bending sight rays with various types of lenses enables magnification of the specific features in the 2D projection [11]. Bier et al. [2] introduced a see-through interface equipped with a magnification lens paradigm called *magic lenses*. Such magnification effects are very useful especially when we want to highlight some specific features in the context of information visualization [7, 16]. For volume rendering, on the other hand, Cignoni et al. [3] introduced the *magicsphere* paradigm that can apply different visualization modalities to the target datasets, and Kurzion et al. [8] simulated 3D object deformations by bending the sight rays together with hardware-assisted 3D texturing. Moreover, LaMar et al. [9] and Wang et al. [21] refined the effects of such magnification lenses and accelerated the associated computation with the help of contemporary hardware environments. However, these approaches aim at simulating rather optimal properties of magnification lenses, and thus have relatively little freedom in distorting 2D perspectives.

Recently, several models have been presented that distort 2D projections by deforming the associated 3D objects. Rademacher [14] introduced a concept of *view-dependent geometry* that encodes view dependency during the phase of 3D object modeling, which leads to several recent methods for controlling 3D shapes according to the camera position. Martin et al. [13] proposed *observer dependent deformations*, utilizing user-defined non-linear functions relating the transformation of a 3D object with its orientation and distance from the camera. Singh et al. [17] presented a *fresh perspective* approach to generate the mixture of perspective, parallel, and other projections by attaching camera constraints that impose perspective locally in 2D projection. This has been extended to a framework called *RYAN* [4], which accommodates temporary coherent animations. They have also developed an interactive interface that directly manipulates the 2D positions of features on the screen for designing nonperspective projec-

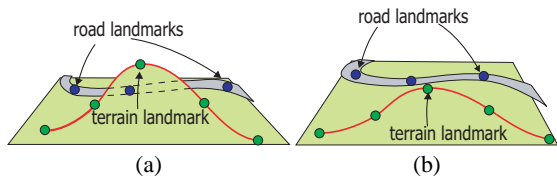


Fig. 3. Resolving occlusions with geographical landmarks : (a) the route occluded by the mountain, and (b) the route that avoids the mountain.

tions [5]. Their framework, however, aims at rather artistic representation of the target scene, and assumes that the 2D positions of feature constraints are provided manually by users. Takahashi et al. [19] introduced a model of *surperspective projection* to simulate hand-drawn illustrations such as mountain guide maps. While the method accomplishes more freedom in the deformation of the target object, it is still limited to static images and cannot provide enough degrees of freedom for our purpose because it tries to find optimal arrangements of a small number of clustered mountain and valley regions in the scene, rather than local characteristic landmarks such as mountain skylines and driving routes.

### 3 SYSTEM OVERVIEW

This section presents an overview of our car navigation system.

#### 3.1 Fundamental settings in the system

Our car navigation system stores elevation data on a regular grid at 50 meter intervals together with road networks in the database. The terrain surface on a grid is triangulated in advance to create a mesh representation. The system then searches for the shortest route between the given starting and end point. Next, the system simulates a drive on the route around the current position by animating nonperspective views of the real terrain surface from a car window.

In our implementation, the tilt angle of the camera is set to 30 degrees and the distance of the camera from the current car position is fixed to be constant. While the tilt angle can usually range from 20 to 70 degrees in commercially available navigation systems, the 30 degrees was chosen in this paper because it is the best to show the effects of occlusion elimination in the system. As shown in Figure 1, the current position is represented by a green wedge-like object and the route of interest is represented by red thick lines; other roads are painted in gray. The car position is fixed at the lower center of the 2D screen so that the forthcoming route can be viewed as clearly as possible in the upper part of the screen, while the previous positions are also tracked as light blue points. See also navigational snapshots of the system in Figures 11 and 12.

#### 3.2 Calculating animated frames

The overall process of calculating animated frames in the navigation system consists of the following four steps:

1. Extracting geographical landmarks such as mountain tops and road features (Section 4).
2. Calculating an optimal arrangement of the geographical landmarks on the 2D screen (Section 5).
3. Deforming the 3D terrain surface so that it satisfies the precomputed arrangement of the landmarks (Section 6).
4. Animating the resulting nonperspective image by retaining its temporal coherence (Section 7).

Each step will be described in Sections 4 to 7, respectively.

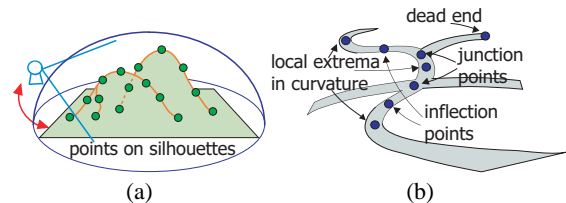


Fig. 4. Geographical landmarks: (a) terrain landmarks and (b) road landmarks.

## 4 EXTRACTING GEOGRAPHICAL LANDMARKS

This section describes a method of extracting geographical landmarks from terrain surfaces and road networks. The geographical landmarks indicate positions on which 2D screen constraints are imposed, in order to deform the target 3D terrain surface later.

### 4.1 Classification of geographical landmarks

Since the aim of distorting perspective images is to avoid occlusions of driving routes, geographical landmarks should be extracted from regions that may cause such route occlusions. For example, Figure 3(a) shows an ordinary perspective image in which a mountain hides a road. This implies that once we can extract geographical landmarks such as mountain skylines and road segments, we can resolve the route occlusion by changing the relative positions of these landmarks together with an appropriate deformation of the terrain surface, as shown in Figure 3(b).

In our approach, we classify the geographical landmarks into two groups: terrain landmarks and road landmarks. The following two subsections describe how to extract these two types of landmarks.

#### 4.2 Terrain landmarks

*Terrain landmarks* are defined as points on the silhouettes of mountains and valleys. The silhouettes correspond to border edges between visible and invisible faces in the triangulated terrain surface [12], and may cause the occlusion of driving routes.

While such silhouette edges are necessary to resolve the route occlusion problems, they still cause some problems in retaining temporal coherence in animation because they are view-dependent features. This suggests the idea that edges on the triangulated terrain surface should be extracted if they are on the silhouettes when seen from one of all possible viewpoints, so that we can track the landmarks consistently with any viewpoints. In practice, we sample viewpoints on the viewing hemisphere that covers the terrain surface, in order to detect such terrain landmarks when seen from each viewpoint sample as a preprocessing step as shown in Figure 4(a). We then store the results in the system so that we can retrieve the extracted terrain landmarks immediately. Note that we apply a Gaussian-like filter [20] in this preprocessing step for finding globally smooth silhouette lines when projecting the terrain surface from viewpoint samples. Figure 5(a) shows terrain landmarks (in green) extracted in our system.

#### 4.3 Road landmarks

As shown in Figure 4(b), *road landmarks* are defined as points of extrema in curvature and inflection points where the signs of curvatures change on the road, as well as its dead ends and junction points. Moreover, the method also extracts the connectivity of the road landmarks as edges to keep the topology of the original road networks. This is because distortion around sharp curves on the road as well as the change in road connectivity can be easily perceived by the human eye. Since these landmarks are inherently view-independent features, they can also be extracted beforehand along with their network connectivity in the system. Figure 5(b) exhibits road landmarks (in blue) extracted in the system.



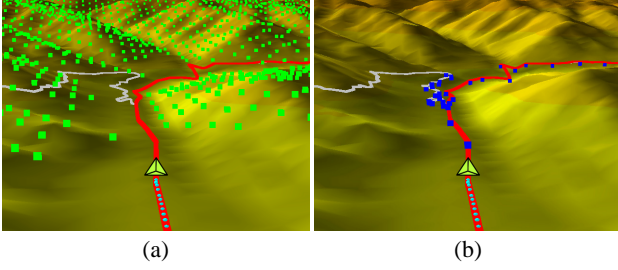


Fig. 5. Landmarks extracted from the terrain surface and road networks: (a) terrain landmarks (in green) and (b) road landmarks (in blue).

#### 4.4 Restricting the area for landmark extraction

Since our attention is limited to the neighborhood of the current car position, the system retrieves terrain landmarks only for local terrain area within the view frustum and some specified radius of the current position. Here, we consider the view frustum of the double-sized screen to cope with drastic changes of view orientations along the driving route. As for the road landmarks, attention is further restricted to landmarks on the driving route of interest. However, in this case, landmarks on other roads around the junction points of the route were also collected in order to maintain the original road shapes in the vicinity of the route for later navigation phases. Figure 5(b) shows that this method also extracts road landmarks on other road (in gray) in addition to those on the route (in red), around the junction point at the center of the screen.

### 5 OPTIMIZING 2D ARRANGEMENT OF LANDMARKS

The extracted landmarks serve as point-position constraints in deforming the 3D terrain surface to generate nonperspective projections. However, in the present approach, we calculate the 3D positions of the landmarks not directly, but indirectly by first calculating the optimal arrangement of the landmarks on the 2D screen and then finding the associated deformation of the terrain surface. This strategy is our major technical contribution because it can fully respect the 2D arrangement of geographical landmarks and thus makes it possible to exclude unexpected occlusion of the driving route while smoothly interpolating the associated perspective over the 2D screen. This section describes an algorithm for automatically finding such optimal arrangement of the extracted landmarks.

#### 5.1 Conditions for occlusion-free arrangements

First, we examine conditions for the optimal arrangement of landmarks on the 2D screen. In order to achieve occlusion-free navigation of driving routes, we employ the following conditions: (1) The arrangement maintains the relative positions of landmarks obtained by projecting them on a horizontal plane. (2) Each landmark lies as near as possible to its corresponding screen position in an ordinary perspective projection.

The first condition is important because the landmarks projected on the horizontal plane form an arrangement that definitely avoids route occlusions on the screen. This is because, in our framework, the terrain surface has a single-valued function representation. Figure 6(c) shows such an arrangement where  $q_i$  ( $i = 1, \dots, N$ ) represents the 2D screen coordinates of the  $i$ -th landmark on the horizontal plane. In our implementation, the horizontal plane is set to be at the same height of the current position, in order to arrange the landmarks around the center on the screen. The second condition maintains the laws of perspective in the 2D projection so that the distortion of the associated ordinary perspective image is minimized. Figure 6(a) shows such an ordinary perspective image where  $p_i$  ( $i = 1, \dots, N$ ) represents the 2D screen coordinates of the  $i$ -th landmark.

These considerations lead to a reasonable compromise between the arrangements of landmarks in Figures 6(a) and (c). Figure 6(b) illustrates such a compromise where  $r_i$  ( $i = 1, \dots, N$ ) represents the optimal 2D screen coordinates of the  $i$ -th landmark. This arrangement avoids

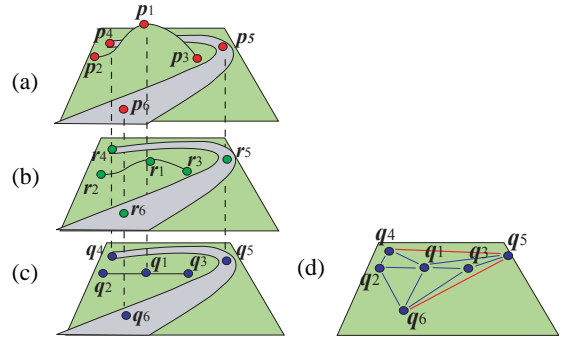


Fig. 6. Arrangement of geographical landmarks: (a) on the terrain surface, (b) in an intermediate state, (c) on the horizontal plane at the height of the current position, and (d) its associated triangulation where red edges are fixed to represent the road network.

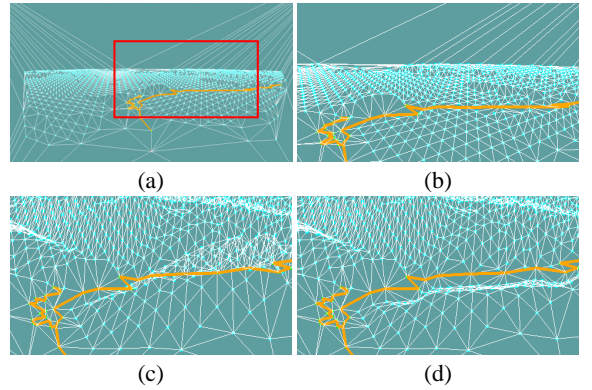


Fig. 7. (a) 2D screen arrangement of landmarks based on constrained Delaunay triangulation, (b) its enlarged images around the route (in orange), (c) an arrangement after the road landmarks is fixed, and (d) the final arrangement of the landmarks. Note that none of the terrain landmarks go beyond the road in the final arrangement.

route occlusions while keeping the laws of perspective in the scene as much as possible. We developed a heuristic algorithm that finds the optimal arrangements of the landmarks  $r_i$ .

#### 5.2 Two-step algorithm for finding landmark arrangements

Our algorithm finds the optimal arrangement of landmarks in two steps. First it locates the optimal positions of the road landmarks by taking account of the influence of its neighboring terrain landmarks, and then calculates the positions of the terrain landmarks that avoid route occlusion. Here, the 2D coordinates  $r_i$  are assumed to move on the line passing through  $q_i$  and  $p_i$ , which implies that  $r_i = (1 - t_i)q_i + t_i p_i$  ( $i = 1, \dots, N$ ). Here, the internal ratio  $t_i$  varies over  $[(0 - \delta), (1 + \delta)]$  in our framework where  $\delta$  indicates some pre-defined margin. (We use  $\delta = 0.2$  in our experiments.) Actually, the algorithm finds an optimal internal ratio for each landmark so that it can offer the best arrangement that suffices the aforementioned conditions.

Before calculating the optimal positions of the landmarks, the algorithm performs the constrained Delaunay triangulation of the 2D screen space with the coordinates  $q_i$  ( $i = 1, \dots, N$ ) where the road landmarks are connected with edges (in orange) beforehand to retain their associated road shape. For example, Figure 7(a) shows the extracted landmarks for the first snapshot of Figure 1, and Figure 7(b) represents its enlarged image.

The algorithm then updates the positions of  $r_i$  by iteratively apply-

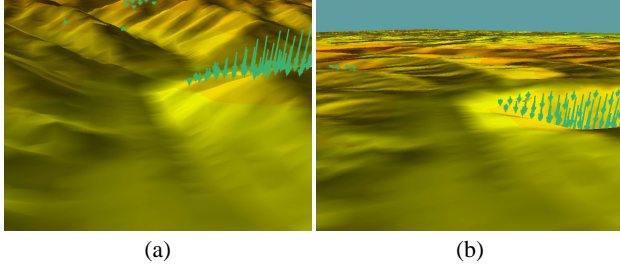


Fig. 8. (a) The shape of the transformed terrain surface and (b) its corresponding displacement from the level surface. Each arrow in green shows the displacement at a extracted terrain landmark.

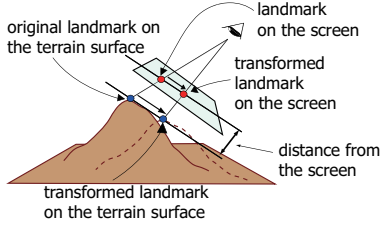


Fig. 9. Relationships between 2D constraints on the screen and 3D constraints on the terrain surface. The distance between the screen and each landmark on the terrain is kept unchanged.

ing forces, to find the equilibrium positions of the road landmarks as shown in Figure 7(c). The forces is formulated as follows:

$$k_a \left( \sum_j \frac{|r_i - r_j| - |q_i - q_j|}{|r_i - r_j|} (r_i - r_j) \right) + k_b (p_i - r_i) + k_c \left( \sum_j \frac{(r_j - r_i)}{n} \right).$$

Here, the first term represents a spring force that retains the distance between  $r_i$  and its neighbors in the original triangulation as shown in Figures 7(a) and (b), the second term a spring force that pulls  $r_i$  toward its perspective position  $p_i$ , and the third term a spring force that moves  $r_i$  toward the average position of its adjacent landmarks. Note that  $j$  represents an index of a neighboring landmark of the  $i$ -th landmark, and  $k_a$ ,  $k_b$ , and  $k_c$  correspond to the coefficients of each force and are set to 0.1, 0.5, and 0.1 in our experiment, respectively. Remember that the first and second forces respect the conditions described in Section 5.1, respectively, while the third one is introduced to avoid undesirable folding of the triangulation.

After fixing the positions of the road landmarks as shown in Figure 7(c), we eliminate the undesired occlusion of the driving route by investigating the relative position of a terrain landmark  $r_i$  (in light blue) and every road edge  $e$  (in orange). This is achieved by examining whether the current terrain landmark  $r_i$  still remains on the same side (left or right) of the road edge as in the initial triangulation. If this is not satisfied, we find an internal ratio  $t_i$  that satisfies the condition by sampling the range  $[0 - \delta, 1 + \delta]$ . The algorithm then applies the same forces to the terrain landmarks as in the first step while fixing the road landmarks. The final equilibrium configuration of the terrain landmarks is thus obtained as shown in Figure 7(d). This configuration enables us to maintain the advantage of perspective projection while occlusions are eliminated, as shown in Figure 8.

## 6 3D DEFORMATION USING DISPLACEMENT SURFACES

To find the 3D terrain surface that enables the obtained landmark arrangement, we must formulate an inverse problem of finding the optimal deformation of the 3D terrain surface from the 2D screen constraints. For this purpose, we introduce a displacement surface that

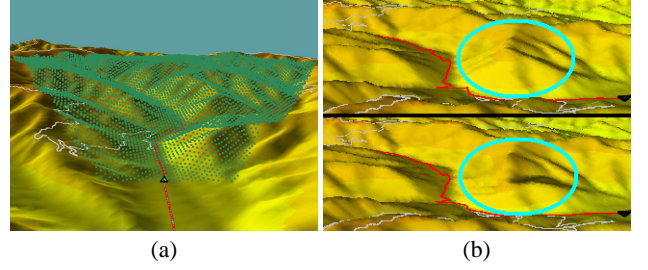


Fig. 10. (a) Restricted area for deformation (in green), and (b) deformation effects from the side view before (upper) and after (lower) the deformation.

represents the difference between the original terrain shape and the deformed terrain shape. The resultant terrain surface should be smoothly deformed while its geographical details are preserved.

### 6.1 Deformation using displacement surfaces

Although the optimal positions of the landmarks on the 2D screen serve as constraints for the 3D terrain surface, they still have one degree of freedom in their positions along the depth axis in the screen coordinate system. Here, a simple scheme is employed that equalizes the depth of each landmark with that of its original position as shown in Figure 9. This simple scheme also keeps the projected size of each feature area unchanged because its distance from the screen is unchanged.

Having obtained the 3D constraints imposed on the target terrain surface, the next step is to formulate its 3D deformation that satisfies the given constraints. Here, the terrain surface is assumed to be a single-valued function  $z = f(x, y)$ , where  $s = (x, y, z)$  represents the 3D coordinates of a point on the surface. This is equivalent to expressing the point using another set of parameters  $(u, v)$  as  $s(u, v) = (x(u, v), y(u, v), z(u, v)) = (x, y, f(u, v))$ .

Now we assume that the point coordinates on the terrain surface  $s(u, v)$  are transformed to new coordinates  $s(u, v) + \bar{s}(u, v)$  through the deformation process. Here,  $\bar{s}(u, v)$  represents a 3D displacement surface to be calculated in our system. Thus, the  $i$ -th 3D position constraint  $w_i$  ( $i = 1, \dots, N$ ) results in the equation  $\bar{s}(u_i, v_i) = w_i - s(u_i, v_i)$ , where  $(u_i, v_i)$  is the original  $(x, y)$ -coordinates of the corresponding  $i$ -th landmark on the terrain surface. This representation has an important advantage because it decomposes the original surface shape into a low-frequency component for the displacement and a high-frequency component for the geographical details.

The actual computation of the displacement surface has been realized by fitting the *multilevel B-spline* surface to the given constraints. Lee et al. [10] introduced an elegant algorithm for accelerating the computation of the multilevel B-splines by taking advantage of a coarse-to-fine hierarchy of control points. In addition, the global smoothness of the resultant B-spline surface can be controlled by selecting the finest level of the approximation hierarchy. Actually, our approach selects the appropriate maximum level of such B-spline hierarchy to guarantee the global smoothness of the perspective interpolated over the 2D image.

### 6.2 Restricting the area for deformation

Recall that when extracting landmarks we restrict our interest to a small region of the terrain surface that is visible from the current viewpoint. This gives us an idea of also restricting the deformation area to a small surface region similarly to the case where the geographical landmarks are extracted, to accelerate the computation of the surface deformation. Figure 10(a) shows such a restricted area, which is colored in green. Actually, this area is in the view frustum of the double-sized screen and some specific radius of the current position, which allows us to avoid unexpected artifacts even when rapidly panning and tilting the view direction. Figure 10(b) shows side views of the terrain

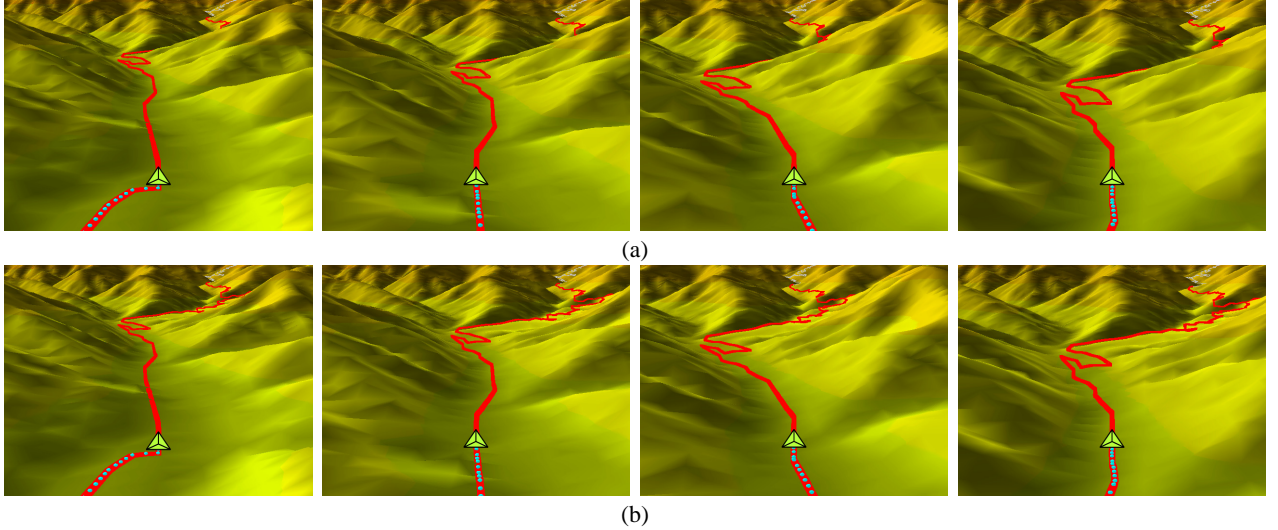


Fig. 11. Animation snapshots in navigating the Hida Highway, Nagano, with (a) ordinary perspective projection and (b) temporary coherent nonperspective projection. The route (in red) is partially occluded by the foot of a mountain on the right in (a) while it is clearly seen up to the destination in (b).

surface before and after the deformation process, where the circles in light blue indicate the difference. These figures exhibit that the deformation area is effectively restricted while its boundary is satisfactory far away from the viewpoint so that we cannot perceive the influence of incoming and outgoing landmarks in animation.

## 7 ANIMATION WITH TEMPORAL COHERENCE

So far the focus has been on how to generate static nonperspective images by deforming the 3D terrain surface using 2D screen constraints. However, simply collecting the consecutive nonperspective frames yields an animation with uncomfortable artifacts such as wavy movements of terrain surfaces and road networks. It is thus necessary to preserve the temporal coherence in animating these nonperspective frames. The following subsections describe ways to preserve such coherence in this approach.

### 7.1 Tracking landmarks over frames

The first idea is to consistently track the geographical landmarks in the view frustum over animated frames. Since the terrain and road landmarks are independent of the viewpoint position, systematic tracking of landmarks is achieved by carefully monitoring outgoing and incoming landmarks at each frame and suppressing their influences on the animation. This is reasonable because in this situation the view orientation remains almost unchanged and therefore the number of outgoing and incoming landmarks is small. Furthermore, the system can also control the number of outgoing and incoming landmarks intentionally to avoid unexpected deformation of the terrain surface even with rapid panning and/or tilting of the view orientation.

### 7.2 Interpolating 3D terrain shapes over frames

In general, temporal coherence in 2D animation is often considered with a spatiotemporal volume that interpolates between successive 2D images. However, in this case, more meaningful coherence can be attempted by interpolating between target 3D terrain surfaces, rather than their 2D projections, because the distortions in 2D projections are driven by the deformation of the target 3D terrain surface. The present system calculates the weighted sum of the terrain shapes at several past and future frames for this purpose. Note that we can take advantage of the terrain shapes at future frames because the driving route to the destination is usually obtained beforehand in navigation systems. Suppose terrain surfaces  $s_i(u, v)$  ( $i = -M, \dots, M$ ) at  $(2M + 1)$  successive frames where  $s_0(u, v)$  represents the terrain

shape at the current frame. We replace the current shape  $s_0(u, v)$  with the weighted sum of these terrain surfaces in our animation, such as  $\bar{s}_0(u, v) = \sum_{i=-M}^M w_i s_i(u, v)$  ( $\sum_{i=-M}^M w_i = 1$ ), where  $w_i$  represents a collection of Gaussian-like weight values and  $M = 4$  in our implementation. This strategy also successfully reduces the influence of incoming and outgoing landmarks on the deformation of the terrain surface.

## 8 RESULTS AND DISCUSSION

### 8.1 Navigation frames

Figure 1 shows animated frames of the driving route navigation in the Takeshi village, Nagano, where upper frames exhibit ordinary perspective images and lower frames nonperspective images with temporal coherence generated in our car navigation system. In this example, the route of interest disappears behind a mountain in ordinary perspective frames, while the nonperspective images successfully display the route by deforming the mountain area on the right. Note that, even in the nonperspective frames, the scene outside the region of interest excludes terrain deformation while preserving the globally smooth change in perspective over the frame.

Figure 11 exhibits navigation displays when the car approaches the mountain pass along the Hida Highway, Nagano, the area famous for its steep mountain regions. Lower nonperspective frames successfully reveal the overall shape of the route while in the upper perspective frames the route is partially hidden by the foot of the mountain. Figure 12 shows animated frames of the route navigation in the Kirigamine area, Nagano. Here, the route is on a famous scenic driving course in Japan. Although the ordinary perspective frames cannot avoid road occlusions due to the bumpy mountain surface, the nonperspective frames can clearly show the route to the mountain top in the vicinity of the current position. This cannot be achieved by simply rendering the mountains and valleys translucently because the route is occluded by the terrain surface multiple times. These results demonstrate that the present car navigation system exhibits the route of interest effectively by deforming the terrain surface around the route while keeping perspective in the scene as much as possible. See also the accompanying video for the full animation sequences.

### 8.2 Evaluation

As shown in Figures 1, 11, and 12, our system successfully avoids the route occlusion while it augments the reality of the navigation frames by assigning appropriate texturing and shading effects to the terrain surface. Here, the multiple tone color texturing with respect to the



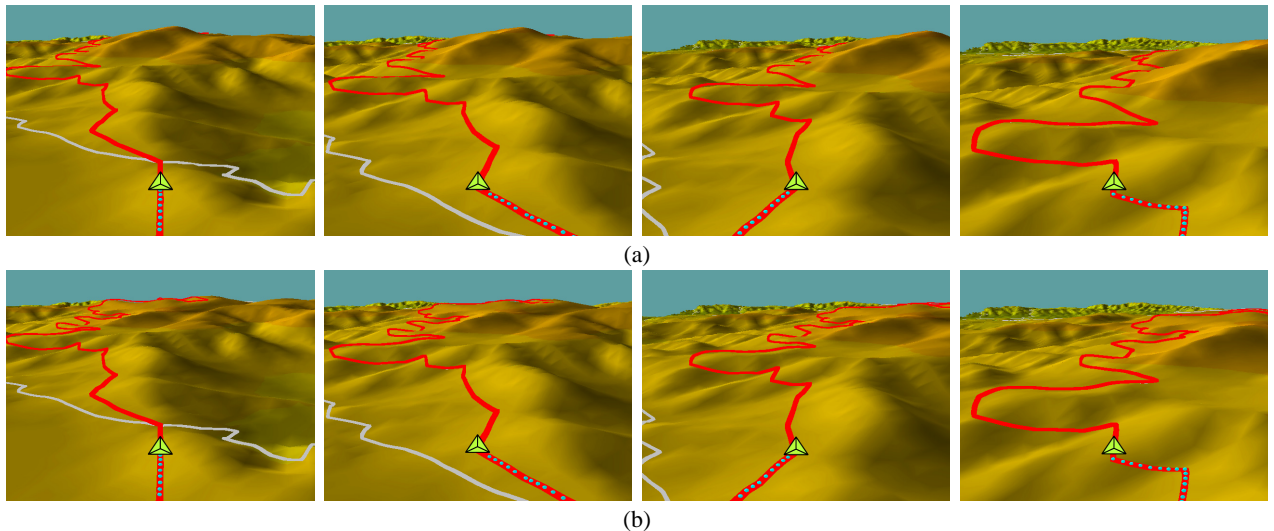


Fig. 12. Animation snapshots in navigating the Kirigamine area, Nagano, with (a) ordinary perspective projection and (b) temporary coherent nonperspective projection. The part of the route (in red) is occluded multiple times by a bumpy terrain surface in (a) while the overall route is visible in (b).

height is applied together with the shading effects that take account of the original geometry of the terrain surface. In order to confirm these effects, we asked six participants (See Table 1) to extract information on driving routes through the navigating frames generated by our system, and tracked their eye movements as shown in Figure 13(a). Here, the participants are requested to see perspective animation clips that guide the Takeshi village and Hida highway first, and then the corresponding nonperspective clips without being informed of the difference between two animation clips. The movements of their eyes are recorded using the eyemark recorder EMR-8B, which is courtesy of NAC Image Technology, Inc.

From this experiment, we obtained the following results:

- Participants B and D almost kept their eye gaze around the current position of the car. This is possibly because they do not drive cars.
- All the participants could not notice the distortion in the nonperspective frames even after they saw the ordinary perspective frames. This implies that our system successfully augmented the reality of the navigation frames by assigning the same texturing and shading effects as those for ordinary perspectives to the terrain surface.
- The eye movement was influenced by the occlusion of the features. Participants A, C, and F moved their eyes along the driving route, while they stopped such eye movement where the route was occluded in ordinary perspective frames as shown in Figures 13(b) and (d). On the other hand, in nonperspective frames, their eyes freely tracked the whole route because its overall shape was always visible as shown in Figures 13(c) and (e).

In addition, we separately asked twenty-five graduate students to find perceivable distortions in the nonperspective frames, however only less than 20% of the students could identify the distortions even after they were informed of the possible existence of such distortions. These results also prove the effectiveness of the present approach to route guidance in car navigation systems.

### 8.3 Limitations

The system generates occlusion-free animation frames with a resolution of  $640 \times 480$  at 4-5 fps on a 3.0GHz Pentium 4 PC with 2GB of RAM. The performance should be improved even for the 1-2 Hz refresh rate of the Global Positioning System (GPS) receivers, while

Table 1. Participants of the eye tracking experiment

	Sex	Drive a car?		Sex	Drive a car?
A	female	Yes	D	male	No
B	female	No	E	male	Yes
C	male	Yes	F	male	Yes

the CPU designed for car navigation systems is still making rapid progress. For example, the latest CPU for in-car use has 3D graphics controller with video memory and now provides up to 600 MHz high-speed processing. Furthermore, the performance of our algorithm can be further improved by using a smaller number of geographical landmarks and sparse samples on the driving route. While this may incur minor artifacts such as small route occlusions and wavy movements of terrain surfaces, it will be still satisfactory especially for in-car use where the route guidance is displayed through a rather small monitor.

The current implementation of our system is still limited to route guidance in mountain areas. Extending our system to the use in city areas provides an interesting and yet a challenging task. Our preliminary experiment shows that directly applying our framework successfully eliminates route occlusions while more attention is required for the arrangement of rectangular objects such as buildings in order to naturally interpolate perspectives over the 2D image.

### 9 CONCLUSION

This paper has presented a method for animating occlusion-free nonperspective projections, and its application to a new type of car navigation system in which driving routes of interest remain always visible in the vicinity of the present position. The system takes as input the discrete elevation model of a terrain surface along with the road networks, and then displays the route that automatically avoids overlaps with surrounding mountains and valleys on the 2D screen. Our technical contribution lies in formulating the nonperspective animation as an inverse problem of deforming 3D terrain surface with temporal coherence while satisfying the 2D arrangement of geographical landmarks. Occlusion-free navigation examples are demonstrated so that we can confirm the visibility of the driving route near the present position even in steep mountain areas.

Extending the present framework to route guidance in city areas is an interesting theme for future research. Exploring new perceptual effects such as image intensities and colors [22] to enhance our vi-

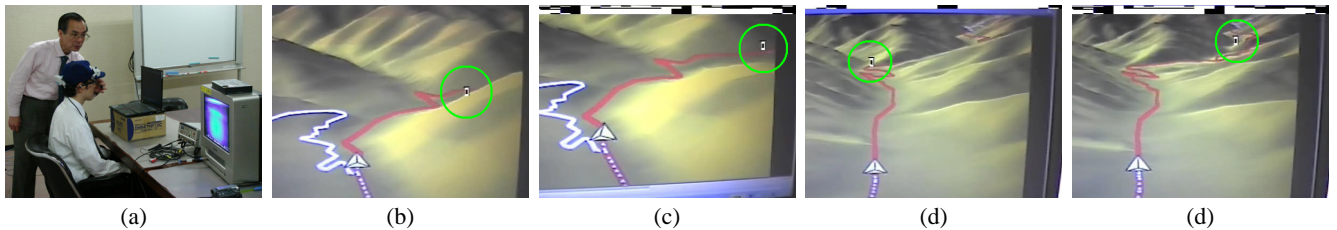


Fig. 13. (a) The eyemark recorder EMR-8B courtesy of NAC Image Technology, Inc. The eye movements in navigating the Takeshi village with (b) ordinary perspective projection and (c) nonperspective projection, and in navigating the Hida highway with (d) ordinary perspective projection and (e) nonperspective projection. Note that the eye movement is stopped by the occluding mountain in perspective frames while it freely tracked the whole route in nonperspective frames.

sual cognition is also a challenging problem. Another future direction might include exploration of new spatial cognition problems where some specific configuration of features on the screen is required.

**Acknowledgments** The eyemark recorder EMR-8B is courtesy of NAC Image Technology, Inc. We thank Kazuyo Kojima for assistance with the accompanying video. This work has been partially supported by Japan Society of the Promotion of Science under Grants-in-Aid for Young Scientists (B) No. 17700092 and Scientific Research (B) No. 18300026.

## REFERENCES

- [1] M. Agrawala, D. Zorin, and T. Munzner. Artistic multiprojection rendering. In *Eurographics Rendering Workshop 2000*, pages 125–136, 2000.
- [2] E. A. Bier, M. C. Sone, K. Pier, W. Buxton, and T. D. DeRose. Tool-glass and magic lenses: The see-through interface. In *In Proc. of ACM Siggraph '83*, pages 73–80, 1993.
- [3] P. Cignoni, C. Montani, and R. Scopigno. Magicsphere: An insight tool for 3d data visualization. In *Proc. of EUROGRAPHICS '94*, pages 317–328, 1994.
- [4] P. Coleman and K. Singh. RYAN: Rendering your animation nonlinearly projected. In *Proc. of the 3rd International Symposium on Non-photorealistic Animation and Rendering (NPAR2004)*, pages 129–156, 2004.
- [5] P. Coleman, K. Singh, L. Barrett, N. Sudarsanam, and C. Grimm. 3d scene-space widgets for non-linear projection. In *Proc. of ACM GRAPHITE 2005*, pages 221–228, 2005.
- [6] B. Gooch and A. Gooch. *Non-Photorealistic Rendering*. A. K. Peters, 2001.
- [7] T. A. Keahey and E. L. Robertson. Nonlinear magnification fields. In *Proc. of Information Visualization '97*, pages 51–58, 1997.
- [8] Y. Kurzion and R. Yagel. Interactive space deformation with hardware-assisted rendering. *IEEE Computer Graphics & Applications*, 17(5):66–77, 1997.
- [9] E. LaMar, B. Hamann, and K. I. Joy. A magnification lens for interactive volume visualization. In *Proc. of Pacific Graphics 2001*, pages 223–232, 2001.
- [10] S. Lee, G. Wolberg, and S. Y. Shin. Scattered data interpolation with multilevel b-splines. *IEEE Transactions on Visualization and Computer Graphics*, 3(3):228–244, 1997.
- [11] Y. K. Leung and M. D. Apperley. A review and taxonomy of distortion-oriented presentation techniques. *ACM Transactions on Computer-Human Interaction*, 1(2):126–160, 1994.
- [12] L. Markosian, M. A. Kowalski, S. J. Trychin, L. D. Bourdev, D. Goldstein, and J. F. Hughes. Real-time nonphotorealistic rendering. In *In Proc. of ACM Siggraph '97*, pages 415–420, 1997.
- [13] D. Martín, S. García, and J. C. Torres. Observer dependent deformations in illustration. In *NPAR 2000: First International Symposium on Non-Photorealistic Animation and Rendering*, pages 75–82, 2000.
- [14] P. Rademacher. View-dependent geometry. In *In Proc. of ACM Siggraph '99*, pages 439–446, 1999.
- [15] S. M. Seitz and C. R. Dyer. View morphing. In *In Proc. of ACM Siggraph '96*, pages 21–30, 1996.
- [16] M. Sheelagh and T. Carpendale. Extending distortion viewing from 2D and 3D. *IEEE Computer Graphics and Applications*, 17(4):42–51, 1997.
- [17] K. Singh. A fresh perspective. In *Proc. of Graphics Interface 2002*, pages 17–24, 2002.
- [18] T. Strothotte and S. Schlechtweg. *Non-Photorealistic Computer Graphics: Modeling, Rendering, and Animation*. Morgan Kaufman, 2002.
- [19] S. Takahashi, N. Ohta, H. Nakamura, Y. Takeshima, and I. Fujishiro. Modeling surperspective projection of landscapes for geographical guide-map generation. *Computer Graphics Forum*, 21(3):259–268, 2002.
- [20] G. Taubin. A signal processing approach to fair surface design. In *Proc. of ACM Siggraph '95*, pages 351–358, 1995.
- [21] L. Wang, Y. Zhao, K. Mueller, and A. Kaufman. The magic volume lens: An interactive focus+context technique for volume rendering. In *Proc. of IEEE Visualization 2005*, pages 367–374, 2005.
- [22] C. D. Wickens, A. L. Alexander, M. S. Ambinder, and M. Martens. The role of highlighting in visual search through maps. *Spatial Vision*, 17(4-5):373–388, 2004.
- [23] D. N. Wood, A. Finkelstein, J. F. Hughes, S. E. Thayer, and D. H. Salesin. Multiperspective panoramas for cel animation. In *Proc. of ACM Siggraph '97*, pages 243–250, 1997.
- [24] D. Zorin and A. H. Barr. Correction of geometric perceptual distortions in pictures. In *Proc. of ACM Siggraph '95*, pages 257–264, 1995.