

Focus+Context Route Zooming and Information Overlay in 3D Urban Environments

Huamin Qu, *Member, IEEE*, Haomian Wang, Weiwei Cui, Yingcai Wu, and Ming-Yuen Chan

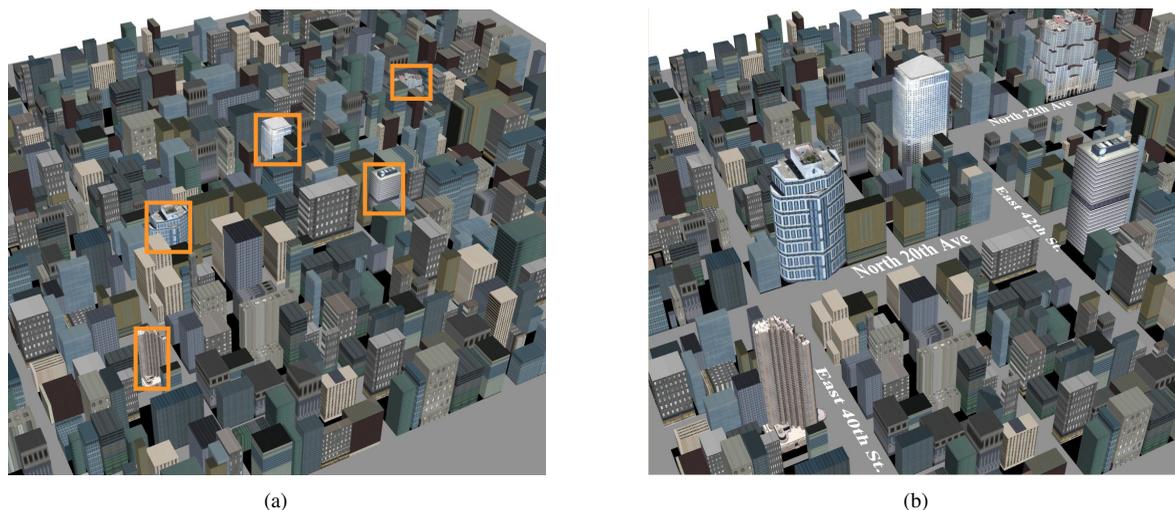


Fig. 1: Focus+Context route zooming and information overlay: (a) A route and its associated five landmarks in a 3D virtual city are occluded by other context buildings; (b) The route and landmarks are enlarged with minimum distortion after applying our route zooming technique. Other useful information such as road names can be conveniently overlaid on the road.

Abstract—In this paper we present a novel focus+context zooming technique, which allows users to zoom into a route and its associated landmarks in a 3D urban environment from a 45-degree bird’s-eye view. Through the creative utilization of the empty space in an urban environment, our technique can informatively reveal the focus region and minimize distortions to the context buildings. We first create more empty space in the 2D map by broadening the road with an adapted seam carving algorithm. A grid-based zooming technique is then used to enlarge the landmarks to reclaim the created empty space and thus reduce distortions to the other parts. Finally, an occlusion-free route visualization scheme adaptively scales the buildings occluding the route to make the route always visible to users. Our method can be conveniently integrated into Google Earth and Virtual Earth to provide seamless route zooming and help users better explore a city and plan their tours. It can also be used in other applications such as information overlay to a virtual city.

Index Terms—focus+context visualization, zooming, 3D virtual environment, seam carving.

1 INTRODUCTION

With the rapid development of 3D modeling and rendering technologies, it is now possible to model a whole city and then show it to the users via Google Earth or Virtual Earth. This opens doors to many applications, especially for tourists, to virtually explore a city and plan their tours. A very common task users often perform in a 3D urban environment is to find a route from one building to another. For example, users may stay in one hotel and need to go to another hotel for a conference, or they may want to know how to go to a tourist attraction from a landmark building. Some systems such as Google Earth or Virtual Earth allow users to input a start and an end location, and a route is then automatically computed. To visualize the whole route and its associated landmarks, users can use the map view, street view, or map+street view. The map view only shows the route and building

names, but the realism of the 3D environment is lost. The street view can show the views along the route perfectly. However, other parts of the city are often occluded by the tall buildings along the street, and users will only have a sense of a small area of the city.

Compared with these views, a 45-degree bird’s-eye view can provide users with a realistic 3D overview of a large area of the city while at the same time keeping the whole route as well as landmarks in sight, making it very useful. However, traditional bird’s-eye views suffer from several drawbacks such as occlusion and cluttered display. Modern cities are often over-crowded with very tall buildings that occlude roads and lower buildings; thus, the route and landmarks users are interested in may be totally occluded. In addition, users are often overwhelmed by the rich but cluttered details. Some landmarks may appear too small and even indiscernible from a bird’s-eye view. It is very difficult for users to zoom into a route and examine some landmarks using traditional zooming techniques. Some critical context may be lost, and users may have to zoom in and out many times to get a clear picture of the route they plan to travel. Some severe distortion may also be introduced.

Given the usefulness of the bird’s-eye view and the tasks users often need to perform, it will be very desirable to develop some interaction techniques that allow users to quickly zoom into a route without losing

• All authors are with the Hong Kong University of Science and Technology, E-mail: {huamin|whaomian|weiwei|wuyc|pazuchan}@cse.ust.hk.

Manuscript received 31 March 2009; accepted 27 July 2009; posted online 11 October 2009; mailed on 5 October 2009.

For information on obtaining reprints of this article, please send email to: tvcg@computer.org.

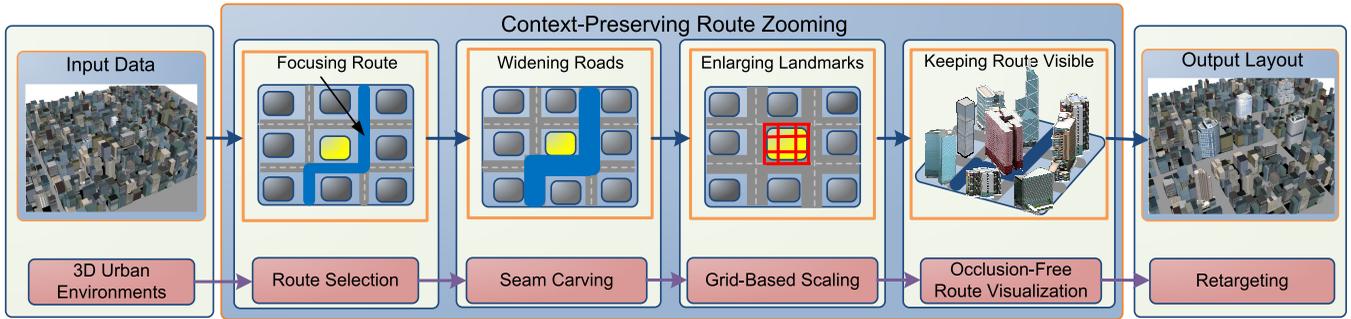


Fig. 2: An overview of our focus+context route zooming process.

the context. In this paper, we develop a novel focus+context zooming technique, which guarantees the visibility of the route and landmarks, and their size can be gradually enlarged with other contexts being preserved. The whole zooming process is very intuitive and can be conducted effortlessly by pressing and moving mouse buttons. The distortions caused by the zooming are minimized by taking advantage of some recent computer graphics techniques, especially seam carving [2] and grid-based focus+context zooming [19]. An extended seam carving algorithm is developed to widen road segments and thus create more empty space around the route and landmarks. Then the grid-based zooming technique [19] is exploited to enlarge landmarks with minimum distortion to reclaim the created empty space. Our method can also facilitate information overlay in 3D urban environments because we can now conveniently overlay some useful information in the newly created empty space.

2 PREVIOUS WORK

Route and Landmark Visualization Agrawala and Stolte [1] presented a set of cartographic generalization techniques to make effective route maps. Chen *et al.* [4] proposed a framework for modeling large street networks by tensor fields. Grabler *et al.* [8] developed a system for the automatic generation of tourist maps, which highlights the most salient objects such as streets and landmarks by multi-perspective rendering and cartographic generalization techniques. Degener *et al.* [5] proposed a method for informatively visualizing short routes in a 3D environment through a single warped image of the routes. Möser *et al.* [12] used multi-perspective and importance scaling techniques to make routes in 3D urban environments more visible. Takahashi *et al.* [16] proposed a car navigation system with occlusion-free driving routes by distorting landmarks such as mountains so that routes will always be visible. In this paper, we focus on focus+context visualization of routes and associated 3D landmarks from 45-degree bird's-eye views, which is quite different from the previous approaches.

Effective landmarks act as important reference points in virtual environments [14]. Birgit and Volker [6] presented a design concept for the visualization of building landmarks and explained how they can be effectively visualized. Tezuka and Tanaka [17] proposed a new approach for extracting landmarks from digital documents using web mining techniques. Grabler *et al.* [8] obtained landmarks through web-based information extraction as well as low-level analysis of the city geometry, textures, and ground plane images. In this paper, we assume that landmarks for a route are already available for our system.

Focus+Context Visualization Focus+context or detail-in-context techniques have been widely used in visualization, computer graphics, and image processing to emphasize focus regions while keeping context regions. One of the widely used focus+context techniques, fisheye view, dates back to a paper published in 1982 by Spence and Apperley [15]. Since then, many papers have been published [7, 9, 10, 11, 13]. Some works used focus+context visualization techniques in 3D urban environments. Möser *et al.* [12] and Carpendale *et al.* [3] introduced the deformation technique into 3D environments to magnify the region of interest (ROI). However, distortions introduced by these methods

are always a major concern. Trapp *et al.* [18] made use of generalization lenses to gain focus+context visualization in virtual urban environments. However much context will be missing by generalization. Recently, a grid-based focus+context zooming with minimum distortion technique [19] has been proposed for 3D geometric objects. In this paper, we use their algorithm to handle the zooming of blocks and landmarks in a 3D urban environment with minimum distortion.

Seam Carving [2] is a very effective technique to support content-aware image resizing. Based on the energy function formulated according to the content of an image, a seam which is an optimal 8-connected path of pixels on a single image from top to bottom, or left to right, can be repeatedly carved out or inserted to an image that best protects the content of the image. In this paper, we use an adapted seam carving algorithm to repeatedly add empty space to a map without causing much distortion to the content of the map.

3 ROUTE ZOOMING

An overview of our system is shown in Fig. 2. To simplify the presentation, we define the following terms: *building* refers to some 3D geometry; *landmark* refers to some buildings users are interested in and want to examine in more detail (marked in yellow in Fig. 2); *road* refers to the surface transportation network represented by a graph; *route* is a path on a road and is what users want to zoom into (marked in blue in Fig. 2); *block* encloses a set of buildings on the ground and is represented by a 2D polygon. Usually a block is surrounded by roads; *empty space* refers to all the spaces not occupied by buildings, including space between buildings in a block and space between blocks; and *3D urban environment* consists of buildings, blocks, and roads. Our model is a simplified representation of a real city environment. The input to our system is a 3D urban environment and a route and its associated landmarks. Such a 3D urban environment can be constructed from a vector 2D map and 3D building models. Some current systems like Google Earth have vector 2D maps and 3D models which can be specified using the KML file format, and thus can generate the input to our system. The landmarks can be manually input or automatically computed. We want develop a system that allows users to zoom into the route and enlarge the landmarks and route in the 3D urban environment while preserving the context with minimum distortion. Our system consists of three major components. First, based on the viewpoint and zoom factor, we select the route and some other road segments and broaden them in the 2D map without introducing much distortion to the other parts. Second, for each landmark, we use a grid-based zooming technique to expand the blocks and the landmarks into empty space. Third, for the buildings still blocking the route, we use an occlusion-free visualization technique to reveal the route to users.

3.1 Road Broadening by Seam Carving

The goal of road broadening is to make the route chosen by users more visible in the final display and create some empty space or "buffer" around the landmarks, which can later be used to absorb the distortion caused by the nonlinear enlargement of the landmarks. Broadening a road without introducing much distortion to other parts is not a trivial problem. We develop a road broadening algorithm based on seam

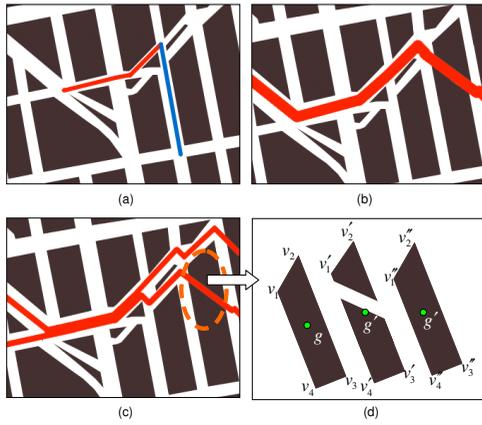


Fig. 3: Road broadening by seam carving: (a) The route is divided into two seam segments shown in red and blue; (b) Seam carving without using the resistance value; (c) Seam carving by dynamically computing the resistance value; and (d) Computing the translation parameters for each block. g and g' represent the centroid of the block before and after the seam carving respectively. The translation for this block is simply $(g' - g)$.

carving [2]. Our algorithm consists of the following five major steps.

Distance Field Calculation In preprocessing, we first discretize a vector digital map and then compute the minimum distance of each pixel to the skeleton of the road, which is represented as a graph. We store the distance along with the road segment ID for each pixel. The reason for discretizing a vector digital map is to better use the seam carving algorithm and handle various special cases. After the road broadening, the discrete digital map is restored to the vector format with updated block positions.

Seam Segment Selection We first divide a route into a set of segments, and each segment serves as part of a seam line for seam carving. As each seam line can only add exactly one pixel for one row or column, the pixels in each segment must have an ascending order of x coordinates or y coordinates. For a selected route R whose skeleton is represented with pixels $P = \{p_1, p_2, \dots, p_n\}$, our route segment division algorithm is shown in Alg. 1.

Algorithm 1: Route segments division algorithm

Input: A route skeleton with pixels $P = \{p_1, p_2, \dots, p_n\}$.

Output: A route segment set Seg , in which each segment has an ascending order of x or y coordinates.

```

1 foreach  $p_i$  do
2   Add  $p_i$  into current segment  $sg_j$ ;
3   if  $sg_j$  has an ascending order of  $x$  coordinates or  $y$ 
   coordinates then
4     Keep  $p_i$  in the current segment  $sg_j$ ;
5   else
6     Remove  $p_i$  out from  $sg_j$ ; output  $sg_j$  to  $Seg$ ;
7     Create a new segment  $sg_{j+1}$  and add  $p_i$  into  $sg_{j+1}$ ;
8     Set  $sg_{j+1}$  as the current segment;
9   end
10 end
    
```

Besides the route, some roads surrounding the landmarks also need to be broadened to create empty space for the future expansion of the landmarks. We call these segments “seam” segments. Then, we can use the seam segments as seeds and repeatedly add seams to the map until the width of each chosen road segment reaches a threshold value computed based on a zoom factor chosen by users or computed by the system. Fig. 3 (a) shows a route that can be divided into two seam segments.

Importance Value Computing Our method is based on seam carving, which needs an importance value for each pixel. The seam carving algorithm adds a seam next to the pixels with the low importance values. To compute the importance value for each pixel, we need to consider the following factors. First, in most scenarios, the added seam should follow the road. Adding space to the road will cause less distortion to the map and will not affect the map’s structure. Thus, the road should have lower importance than the buildings and blocks. Second, because we want to broaden a specific route, the route should have lower importance than other ordinary roads. Third, the empty space between the buildings should have lower importance than the buildings themselves. To summarize, a building is more important than the empty space in its associated block, which has a higher importance value than the road. The ordinary road will have more importance than the route or the road segments to be broadened. Based on these observations, we divide the pixels into different categories, which have different importance values. Another issue we need to consider is that after broadening the route and area around the landmarks, we want the caused distortions to be absorbed globally. Thus, we do not want the over-expansion of a subregion. To model this, we introduce a resistance value to indicate the distortion level around a pixel. Taking these factors into consideration, we design the following importance function to compute the importance value for pixel (i, j) :

$$I_{ij} = Dist_{ij} + Cat_{ij} + Res_{ij}, \quad (1)$$

where I is the importance value; $Dist$ is the minimum distance of a pixel to the road; Cat is a category value; and Res represents a resistance value used to prevent the over-expansion of a subregion. The category function assigns different importance values to different categories of pixels. We divide all pixels into four categories in our system: pixels in a route segment to be broadened, pixels in other road segments, pixels in other empty spaces (*i.e.*, space between buildings in a block), and pixels representing buildings. The rationale behind using a category function is to force the seams to be first added to a category with low importance. The importance value for each category can be configured in our system. By adjusting the category importance, we can control where to create extra empty space. The minimum distance term is used to lead the seam to the road. The resistance value is computed dynamically after each seam carving iteration.

Dynamic Seam Carving Once we have the importance value for each pixel, we can run the seam carving algorithm for each seam segment. In the seam carving process, at every step, a seam representing a pixel-wide empty space line that is added into the road map. We define the cost of each seam as $C(s) = \sum_{i=1}^n I(p_i)$, where $I(p_i)$ is the importance value of pixel p_i in a n -pixel seam. For each seam carving step, we compute an optimal seam s^* that minimizes the seam cost $C(s)$. We can dynamically adjust the resistance value for each pixel to disperse the distortions globally and prevent an area from being over-broadened. If we give a high resistance value to an area, seams will less likely be added into this area. For example, we can assign the resistance value for an area to be the times in which the seams have been added to this area. We can also assign a very high resistance value for pixels in a road segment to totally shut off that branch (*i.e.*, no seams will be added to the branch). This can avoid the over-expansion of the road branch, thus reducing distortion. Fig. 3 compares the seam carving results with and without the resistance value. Note that the seam segment has been dramatically enlarged, while the distortions to other parts are dispersed in Fig. 3 (c).

Translation of Blocks After seam carving, more space will be created around the route area, and the blocks will change their positions in the world coordinate. We need to record the translation values for each block, and then let the graphics hardware handle the real transformation for all 3D models in the block. Each block is represented as a polygon in the 2D map with vertices $V = \{v_1, v_2, \dots, v_n\}$. We only need to record the positions of these vertices before and after the seam carving. We can then compute the centroid of a block before and after the seam carving. We will use the position difference of the centroid to compute the translation parameters for the whole block

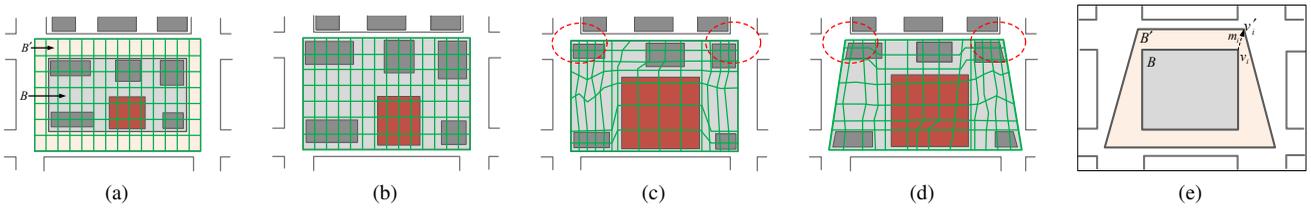


Fig. 4: Grid-based scaling for a block: (a) Original block layout B , with the red square as the landmark; the bounding space B' is B plus the adjacent empty space, and is partitioned by regular grid space; (b) Linear scaling; (c) Grid-based scaling to keep the shape of the original buildings; the scaling changes of the two corners (encircled in red) are abrupt; (d) The bounding space is adapted to the perspective view; the scaling changes are smoother compared with (c); and (e) Initialize the bounding space B' to the area of B plus the adjacent empty space, and then set the shape of the bounding space to be consistent with the perspective view by adjusting the value of m_i .

(see Fig. 3 (d)). Note that our algorithm allows a seam line passing through a block, which may cause damage to the block boundary in the 2D image. But this will not cause any problem because we only use the seam carving result to compute the translation parameters for each block. After the translations of these blocks in the world coordinate, some extra space around the route will be created, and the block boundaries and buildings are not distorted.

Compared with the original seam carving algorithm, our method introduces the following enhancements. First, we develop a new algorithm to compute the importance value, which is specially designed for 2D maps and our applications. Second, the importance value can be dynamically changed to dissipate the distortions caused by broadening the road. The added seams are clustered along the road segments we intentionally want to broaden but are dispersed elsewhere. As the distance field can be pre-computed, the adapted seam carving algorithm can be run efficiently.

3.2 Grid-based Building Scaling

After seam carving, more empty space is reclaimed, resulting in wider roads. The landmarks can be enlarged for clarity in the view by utilizing the empty space. A straightforward solution is to uniformly scale the building size. However, this can only fill up limited adjacent empty space. Various Focus+Context techniques have been proposed to suppress the context for revealing the details in focal regions, like fisheye. The major concern on these methods is the undesired distortion to viewers' perception. For example, the shape of the model may be severely deformed.

The topology and the appearance of the map must be maintained. A model-preserving distortion method [19] was proposed to efficiently magnify the focal region by squeezing the empty space. A similar technique was also applied on feature-preserving image rescaling [20]. In this paper, we apply the grid-based scaling [19, 20] to 3D urban environments, and use it to magnify the landmarks while keeping the context buildings with minimum distortion. We introduce a boundary scheme to guarantee the smooth transition from the enlarged block to neighboring blocks.

Assuming we want to enlarge a block B with vertices $V = \{v_1, v_2, \dots, v_n\}$, we first initialize the bounding space B' to the area of B plus the adjacent empty space M (see Fig. 4(e)). For each vertex v_i in B and vertex v'_i in B' , we have

$$|v'_i - v_i| = m_i, \quad (2)$$

where m_i is the empty space around v_i (see Fig. 4(e)). We can also set the shape of the bounding space to be consistent with the perspective view by adjusting the value of m_i (i.e., the closer v_i to the viewpoint, the larger the m_i) (see Fig. 4(e)). The area of the bounding space B' is maintained during the optimization process.

The bounding space B' is then partitioned using a regular grid space. Each quad defined by the grid is categorized as *empty quad*, *landmark quad*, or *non-landmark building quad*. A weight is assigned to each quad to control the distortion. The building and landmark quads are usually assigned a high value to avoid severe distortion, while empty

quads are given a value close to 0. When users zoom into the selected route, the landmarks in the blocks will be magnified by expanding the landmark grids, and the expanding scale is determined by the zooming parameter controlled by users. The remaining grids are automatically deformed to squeeze the empty space or shrink the other non-landmark buildings. For more details about the grid-based scaling algorithms, please refer to [19, 20].

Translation and Scaling of Buildings After the grid-based scaling, we can compute the affine transformation for each building in the block. Usually, the buildings are just translated and scaled. Given the original bounding box and the new bounding box of each building before and after the grid-based scaling, an affine transformation can be easily established. Please note that we do not modify the vertex coordinates of buildings. We only compute the transformation matrix for each building and then let the graphics hardware handle the real transformation and rendering.

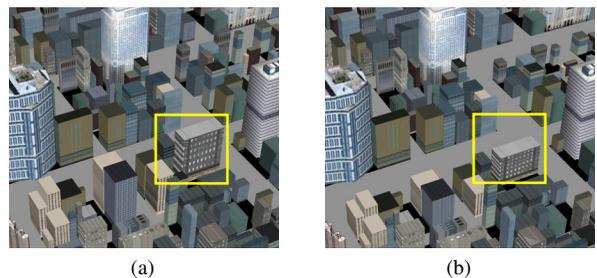


Fig. 5: Occlusion-free building adjustment: (a) The building in the yellow rectangle occludes the selected route; (b) The result after scaling the building.

3.3 Occlusion-Free Route Visualization

Although the visibility of the selected route and landmark buildings has been dramatically improved after the carving and scaling processes, some buildings in front of the road may still block the route (see Fig. 5(a)). We have experimented with several solutions to achieve an occlusion-free route visualization. One simple solution is to remove all the non-landmark buildings occluding the selected route. However, the context will be lost and users may be misled. Another solution is to make these buildings transparent. This result is acceptable. We can also artificially lower these buildings until they no longer block the route (see Fig. 5(b)). A similar approach is adopted by Takahashi *et al.* [16] to achieve occlusion-free route visualization in a rural environment. We find that there are no perfect solutions to this problem. Fortunately, users can change viewpoints to observe the route from different angles to alleviate this problem.

3.4 Interactive Route Zooming

Our system allows interactive zooming into a selected route. Users can specify two locations on the map, and a route can be automatically computed. A focus+context view of the 3D virtual city will be generated based on the route. Important landmarks pre-defined or selected by users are magnified in the zooming process for exploring the details. One drawback of the typical zooming operations is that context may be clipped from the view. In our system, users can choose a zoom factor, which determines the scaling factor λ of the landmarks and route, to enlarge the details. The contexts are only squeezed in to spare more space for the enlarged landmarks and route. A smooth animation on the zooming is generated by gradually increasing the scaling value λ of the landmark and route. Occlusion is eliminated by dynamic building adjustment. Re-targeting operation is also supported to allow users to interactively refine the route query. Routes are broken down into segments, and users can re-target the different parts of the route for a detailed view. By specifying multiple locations on the map, the routes between them can be selected interactively, and views are regenerated for comparison.

4 APPLICATION

Our route zooming technique can broaden the route and enlarge the buildings of interest with minimum distortion, which leads to some interesting applications such as information overlay and clutter reduction for annotations.

4.1 Information Overlay

Information overlay is a widely used visualization technique. It is now possible to overlay labels, images, and videos to Google Earth via KML files. In some applications, users may want to investigate the correlation of certain data such as air pollution, light pollution, traffic, resident activity, and wireless signal strength with a 3D urban environment. Overlaying such information onto the 3D urban environment can then greatly facilitate the visual analysis. However, this is not a trivial task. The main challenge is how to avoid the occlusion caused by dense tall buildings and to convey the correlation between the data and the surrounding 3D environment. For example, if we directly overlay the roadside air pollution information onto the ground, it will be occluded by the surrounding buildings (see Fig. 6(a)). If the information is placed over the buildings, it will block the buildings, and the correlation cannot be easily detected by the users. Our route zooming technique can effectively solve this problem. We can first widen the selected route through our seam carving algorithm to gain enough empty space, and then overlay the information onto the broadened road. Using our technique, the overlaid information becomes visible to the users, and the correlation with the 3D environment can be conveniently analyzed (see Fig. 6(b)).

4.2 Clutter Reduction for Annotations

Annotations in the form of text labels or iconic symbols can provide some essential information about an object in a 3D urban environment. However, adding annotations to buildings in a 3D urban environment may cause severe visual clutter if these buildings are too close (see Fig. 6(c)). Many solutions have been proposed. Our method can also facilitate clutter reduction for annotations. We can create some extra space around the buildings so we can have more room to layout these annotations. For example, we can increase the distance between buildings or broaden the roads surrounding these buildings. In Fig. 6(d), we apply our route zooming to a cluttered area, allowing annotations to be easily placed onto the roads without overlapping. Compared with other methods, our approach does not cover other buildings with annotations; thus, no building details (e.g., facades and shapes) are lost.

5 EXPERIMENTS AND EVALUATION

We have conducted the experiments on an Intel(R) Core(TM)2 2.13GHz PC with 1GB RAM and an NVIDIA Geforce 7900 GS GPU with 256MB RAM. The routes, landmarks, and viewpoints were manually selected to illustrate our algorithm. The major computational

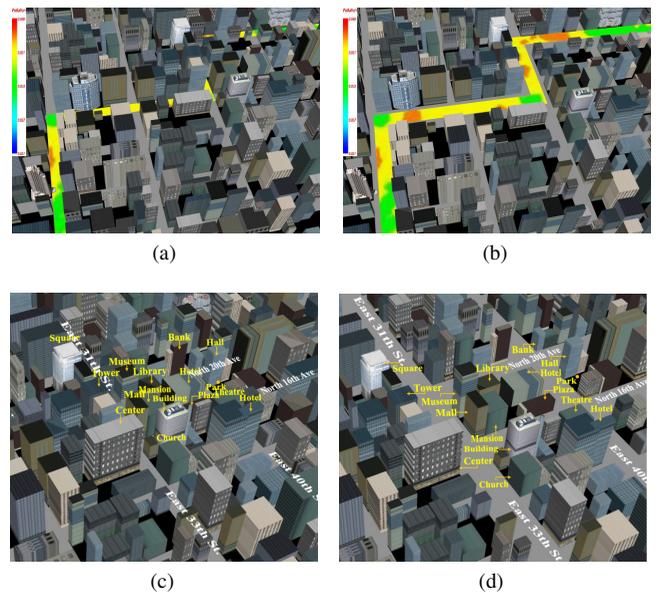


Fig. 6: Applications of our methods: (a)-(b) Overlaying a roadside pollution map in a 3D urban environment before and after applying our route zooming; (c)-(d) Placing some annotations in a 3D urban environment before and after applying our route zooming.

costs of our algorithm are using the seam carving algorithm to compute the translation parameters for each block and using the grid-based scaling to compute the affine transformation matrix for buildings in the blocks having landmarks. We only report the CPU cost of our experiments because the GPU is very powerful nowadays and can give real time rendering performance.

5.1 Results

We first tested our road broadening algorithm on two real maps. One map shows an area in Hong Kong, while another shows an area in London. We selected two quite different routes, a zigzag one and a loop one, to demonstrate the effectiveness of our method. Fig. 8 shows the results. The chosen routes are shown in red in the left column. The processes of the dynamic seam carving are shown in the middle column with the added seams marked in blue. The final results are shown in the right column. We can see that the routes have been widened without affecting the structure of the whole map. The distortions to the other parts are minimum. This experiment demonstrates that our algorithm can create extra empty space in a map and effectively broaden different routes, such as zigzag routes and loop routes, without causing much distortion to the map. The resolution of the Hong Kong map is 1014×541 , and the seam carving process with 33 iterations took 0.1 seconds. The resolution of the London map is 924×636 , and the seam carving process with 19 iterations took 0.06 seconds. The selected routes in the map were both broadened about 2-3 times.

Fig. 9 shows the results of our grid-based zooming technique. After the road segments were broadened around the landmarks, we had extra space which could be exploited to enlarge the landmarks. Fig. 9 (a) shows a block with six buildings, and the landmark building is labeled in red. Fig. 9 (b) shows the new layout after the road broadening and some extra empty space was created around the block. Fig. 9 (c) shows the result after the grid-based zooming. The block was divided into 80×150 grids, and the grid-based scaling for the block only took 0.01 seconds. Note that the boundary of the block was realigned with the road such that there was no distortion for the block boundary except scaling. In addition, the shapes, relative positions, and relative sizes of other buildings in the block were still well kept. As we can optionally choose to keep the original size of these context buildings, users will not misperceive the sizes of these buildings compared with

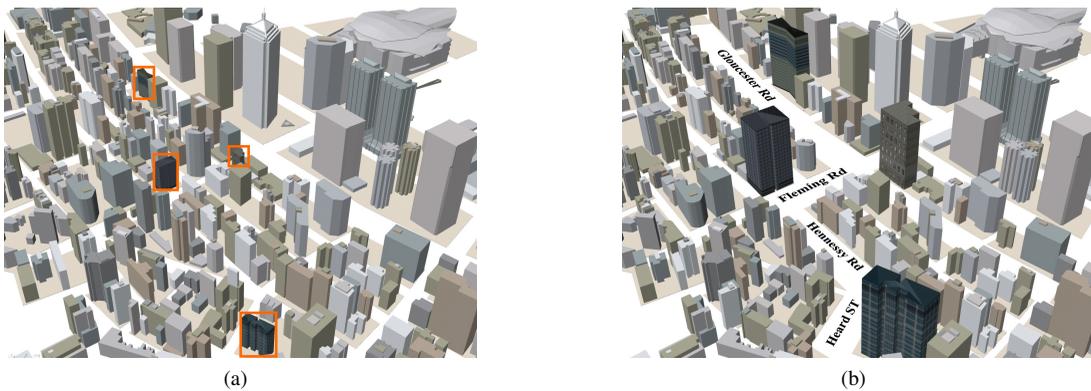


Fig. 7: The results on a 3D Hong Kong environment: (a) A route with four landmarks; (b) The result after the focus+context route zooming.

the other context buildings. From this example, we can see that the grid-based non-linear zooming technique did not distort the shapes of the buildings and best kept their relative positions.

After that, we tested our system with an artificial 3D city modeled after Manhattan, which has a regular grid road network. Fig. 1(a) shows the city model where a route and five landmarks were manually selected. The view is very cluttered. The route is barely discernible, and the landmarks are buried in the context buildings. We applied our focus+context route zooming technique to the route and landmarks. The result is shown in Fig. 1(b). The visibility of the route has been dramatically improved. We scaled a few buildings in front of the road to totally reveal the route to the viewers. The landmarks are enlarged, while the context buildings are all maintained. The distortion to the context is much smaller compared with other focus+context techniques such as fisheyes. More importantly, we can now conveniently overlay useful information on the road. For example, the road names are overlaid on the road in Fig. 1(b). Our focus+context view of the route and landmarks is more informative than the other simple zooming schemes. The map was converted to a 512×512 image during preprocessing. The seam carving process with 44 iterations for the selected route took about 0.1 second, which broadened the route four times. The grid-based scaling for 5 blocks took 0.04 seconds.

Our next experiment was conducted on a 3D urban environment modeled after London, whose 3D environment looks quite different from that of Manhattan. Fig. 10 (a) shows a region in London. We can see that the road system is quite irregular, and the blocks are no longer rectangular. We selected a long route in the map to test our algorithm. As we do not have the detailed 3D models in this area, we built a 3D virtual environment using the sketch up tool provided by Google (see Fig. 10 (b)). For comparison, Fig. 10 (c) shows the result of using a uniform scaling. Only two landmarks and part of the route are now visible. Other useful information is pushed out of the window, and the context is lost. To provide a focus+context zooming, we applied our route zooming technique to the whole route and the result is shown in Fig. 10 (d). As the route is too long, not all the contents can be shown in the display window, and some landmarks and route segments are pushed outside the window. To solve this problem, we applied our technique to only part of the route. Fig. 10 (e) shows the result after applying our technique to the top half of the route, while Fig. 10 (f) shows the result after applying our technique to the bottom half of the route. From the figures, we can see the results have become much better. The map resolution is 512×412 . The seam carving algorithm took 0.12 seconds, while the grid-based scaling for six blocks took about 0.04 second. The route was enlarged four times.

We also tested our algorithm on a real Hong Kong city model and the results are shown in Fig. 7. Fig. 7 (a) shows a cluttered view of a route and its associated four landmarks. The layout in Fig. 7 (b) (generated in 0.2 seconds) becomes clearer and more informative after applying our technique.

As the time performance for a single frame in our current system was not very fast, we adopted a keyframe-based animation technique to speed up the zooming process. Zooming is usually continuous, and users often need to pause and conduct various interactions; thus, our system immediately started the seam carving process once users selected a route. The results at various discrete zooming levels (*i.e.*, zoom factor = 2, 4, 6, *etc.*) were computed and stored when users just started the zooming or conducted other tasks. These results then served as the key frames. For other zooming levels, linear interpolation was used to generate the intermediate results (*e.g.*, the translation parameters for blocks and the transformation parameters for landmarks and other buildings). With this scheme, our system can achieve interactive speed for all the city models in the experiments. As most of our computations such as scaling different blocks can be conducted in parallel, we believe that the computation time can be dramatically reduced if the GPU acceleration is exploited.

5.2 Evaluation

In order to validate our route zooming technique, we conducted a user study and invited 25 college students to participate. Among these 25 participants, six of them frequently use Google Earth or similar systems; 14 of them sometimes use them; and five of them seldom use them. Both the Manhattan style map (see Fig. 1) and the London style map (see Fig. 10) were used in the user study. We provided three navigation styles: the 2D route map with landmark icons (see Fig. 10 (a)), the 3D urban environment equipped with our focus+context route zooming interaction, and the 3D urban environment with the traditional zooming, panning, and rotation. After a five-minute introduction, the participants spent ten minutes to become familiar with our system. Afterwards, a route from one location to another location in each map was selected for the participants, and they were required to use different navigation styles in random order to explore the route. After they finished, they were asked to rank different navigation styles based on how effectively the style could provide the overall information (*i.e.*, turning points, landmarks, and real environment) of the route. 20 subjects ranked our method as the most effective navigation style, while the remaining five ranked our method as the second.

We asked the participants to give further evaluation of our technique by answering some general questions. When asked whether the distortion in our method misleads users, 18 participants answered no, while 7 participants thought that sometimes the results were misleading. The participants were then asked to compare our method with the fish-eye view. 21 subjects felt that the distortion introduced by our technique was more acceptable and less severe than the distortion caused by the fish-eye view. However, when asked whether our technique is useful for finding the path in a real city environment, only 13 subjects thought it is helpful while the others thought it depends on the real city environment. We will discuss this limitation in the next section. Finally, 22 participants felt that our technique could help in planning

tours before visiting a real city and would like to see this feature in Google Earth or similar systems. Overall, the feedback to our technique is very positive. Some suggestions for improvement were also provided by the participants. For example, some critical information (e.g., turning/crossing points) could be highlighted when zooming into the route, and the snapshots of landmarks from different view angles could be provided for users.

6 DISCUSSION

From the experiments, we can clearly see that our method provides a new way to conduct zooming in a 3D urban environment. We broaden the route of interests and magnify the landmarks without removing other parts. According to our experiments, our algorithm can handle various routes (i.e., zigzag, loop, regular, and irregular routes) in different 3D environments. Our focus+context route zooming visualization is very useful for users to navigate in 3D urban environments. It can greatly reduce some tedious and time-consuming user interactions such as switching between different views, zooming in/out, and panning. To view a selected route in detail, users just need to move mouse and zoom into the route, and then our technique can automatically and clearly reveal the route and the landmarks with low distortions while keeping the surrounding 3D environments to provide a nice overview.

Our method has some limitations. Many 3D urban environments have clear structures and plenty of empty space, which can be exploited by our algorithm. However, for very complicated city layouts or very long routes with too many landmarks, the results may not be good if we directly apply our algorithm to the whole route as demonstrated in the London example. In this case, we can divide a route and landmarks into several sets and allow users to examine each set separately using our method. One solution is to calculate the zooming scales for different parts of the route according to the distance to the viewpoint. Another solution is to use the ruffling technique, by which the area pointed by the mouse is first zoomed in, and then returns to normal after the mouse moves away. In rare situations where a few landmarks are grouped together, our algorithm may not create enough space to magnify all these landmarks without pushing parts of the route and context buildings out of the display. Another limitation is that if a landmark lies in front of a route, magnifying it will further occlude the route. To solve this problem, users may have to change viewpoints to view the route and the landmarks from different angles. Automatic viewpoint selection may alleviate this problem, and we will leave this for future work. As our method is designed for the bird's-eye view of a city, its main purpose is for route query, tour planning, and information overlay. For orientation and wayfinding in real city environments, as the street views and the 45-degree bird's-eye views can be quite different, users are suggested to use the street views of the route together with the views provided by our system. Users can virtually walk along the chosen route at the street level to obtain a sense of the real environment for orientation and wayfinding.

7 CONCLUSION AND FUTURE WORK

In this paper, we have presented a novel zooming technique to visualize a route and its associated landmarks from 45-degree bird's-eye views. By taking advantage of some recent developments in computer graphics, especially seam carving and grid-based zooming, our method can reduce distortion and provide seamless focus + context zooming to help users perform a very common task in 3D urban environments. The novelty of our method lies in a creative utilization of the empty space in a 3D urban environment. We first applied an adapted seam carving algorithm to expand the empty space (mainly roads), which can be used as a buffer to absorb the future expansion of the route and landmarks. We then used a grid-based zooming technique tailored for 3D urban environments to enlarge landmarks with minimal distortion to buildings and blocks. Compared with other methods, we did not distort any building shapes except scaling, and the visible road segments maintained their original straightness. The distortions caused by the enlargement of the route and landmarks were absorbed globally by the empty space originally existing or later created. Most computations in our method are in 2D, which can be done efficiently.

This new zooming feature can be integrated into typical 3D urban environments such as Google Earth and Microsoft Virtual Earth to help users explore a city and plan their tours.

There are many possible avenues for future work. We want to investigate a force-based model to expand the blocks into the newly created empty space. We will study how to combine our technique with multi-perspective view to make it more powerful. We also want to use GPU to accelerate our techniques. As the seaming carving algorithm is designed for images, our current method needs to discretize the vector map into an image during pre-processing. We plan to investigate a robust road widening algorithm that directly works on vector maps.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments. The Hong Kong city model is courtesy of the CompuMaps. This work was supported in part by grant HK RGC CERG 618705 and 618706.

REFERENCES

- [1] M. Agrawala and C. Stolte. Rendering effective route maps: Improving usability through generalization. In *ACM SIGGRAPH 2001*, pages 241–249, 2001.
- [2] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. *ACM Trans. Graph.*, 26(3):10, 2007.
- [3] M. S. T. Carpendale, J. Ligh, and E. Pattison. Achieving higher magnification in context. In *Proceedings of the ACM symposium on user interface software and technology*, pages 71–80, 2004.
- [4] G. Chen, G. Esch, P. Wonka, P. Müller, and E. Zhang. Interactive procedural street modeling. *ACM Trans. Graph.*, 27(3):103, 2008.
- [5] P. Degener, R. Schnabel, C. Schwartz, and R. Klein. Effective visualization of short routes. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1452–1458, 2008.
- [6] B. Elias and V. Paelke. User-centred design of landmark visualizations. *Map-based Mobile Services*, pages 33–56, 2008.
- [7] G. W. Furnas. Generalized fish-eye views. *Proceedings of the ACM SIGCHI*, 17(4):16–23, 1986.
- [8] F. Grabler, M. Agrawala, R. W. Sumner, and M. Pauly. Automatic generation of tourist maps. *ACM Trans. Graph.*, 27(3):11, 2008.
- [9] J. Heer and M. Agrawala. Multi-scale banking to 45 degrees. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):701–708, 2006.
- [10] T. A. Keahey. The generalized detail-in-context problem. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 44–51, 1998.
- [11] Y. K. Leung and M. D. Apperley. A review and taxonomy of distortion-oriented presentation techniques. *ACM Trans. Comput.-Hum. Interact.*, 1(2):126–160, 1994.
- [12] S. Möser, P. Degener, R. Wahl, and R. Klein. Context aware terrain visualization for wayfinding and navigation. *Computer Graphics Forum*, 27(7):1853–1860, 2008.
- [13] A. Raab and M. Rüger. 3D-ZOOM: Interactive visualisation of structures and relations in complex graphics. *3D Image Analysis and Synthesis*, pages 125–132, 1996.
- [14] M. Sorrows and S. Hirtle. The nature of landmarks for real and electronic spaces. *Spatial Information Theory*, pages 37–50, 1999.
- [15] R. Spence and M. Apperley. Data base navigation: an office environment for the professional. *Behaviour & Information Technology*, 1(1):43–54, 1982.
- [16] S. Takahashi, K. Yoshida, K. Shimada, and T. Nishita. Occlusion-free animation of driving routes for car navigation systems. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1141–1148, 2006.
- [17] T. Tezuka and K. Tanaka. Landmark extraction: A web mining approach. *Spatial information theory*, pages 379–396, 2005.
- [18] M. Trapp, T. Glander, H. Buchholz, and J. Dolner. 3D generalization lenses for interactive focus + context visualization of virtual city models. In *Proceedings of the IEEE International Conference on Information Visualization*, pages 356–361, 2008.
- [19] Y.-S. Wang, T.-Y. Lee, and C.-L. Tai. Focus+context visualization with distortion minimization. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1731–1738, 2008.
- [20] Y.-S. Wang, C.-L. Tai, O. Sorkine, and T.-Y. Lee. Optimized scale-and-stretch for image resizing. *ACM Trans. Graph.*, 27(5):8, 2008.

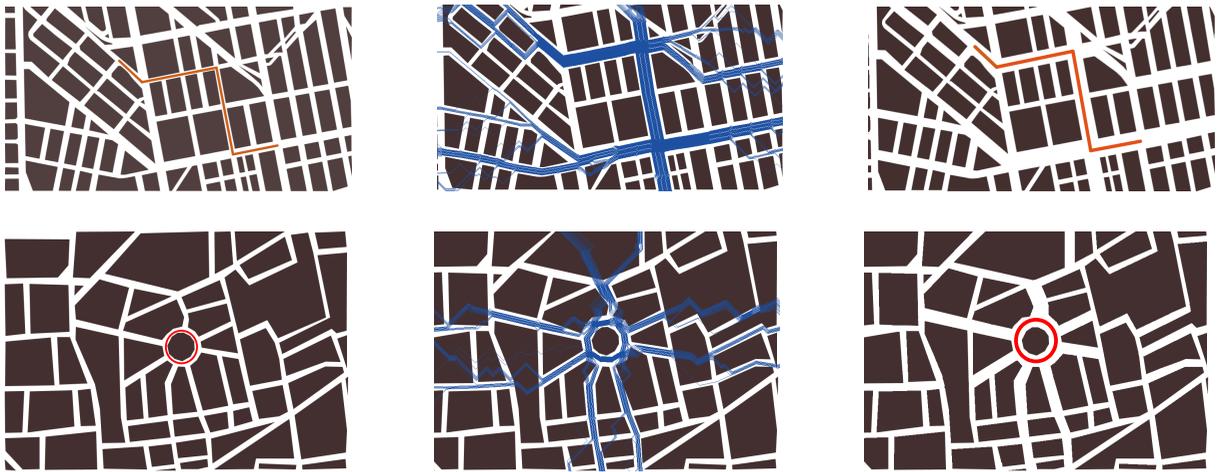


Fig. 8: Roads broadening results for a Hong Kong map (top row) and a London map (bottom row). The left column shows the original maps. The middle column shows the seam carving processes with the added seams shown in blue. The final results are put in the right column.

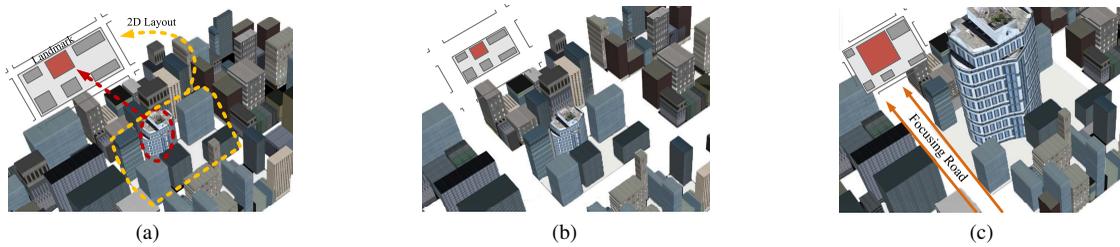


Fig. 9: Grid-based scaling results: (a) A block with one landmark and five context buildings; (b) Empty space is created around the block after road broadening; (c) Result after applying grid-based scaling.

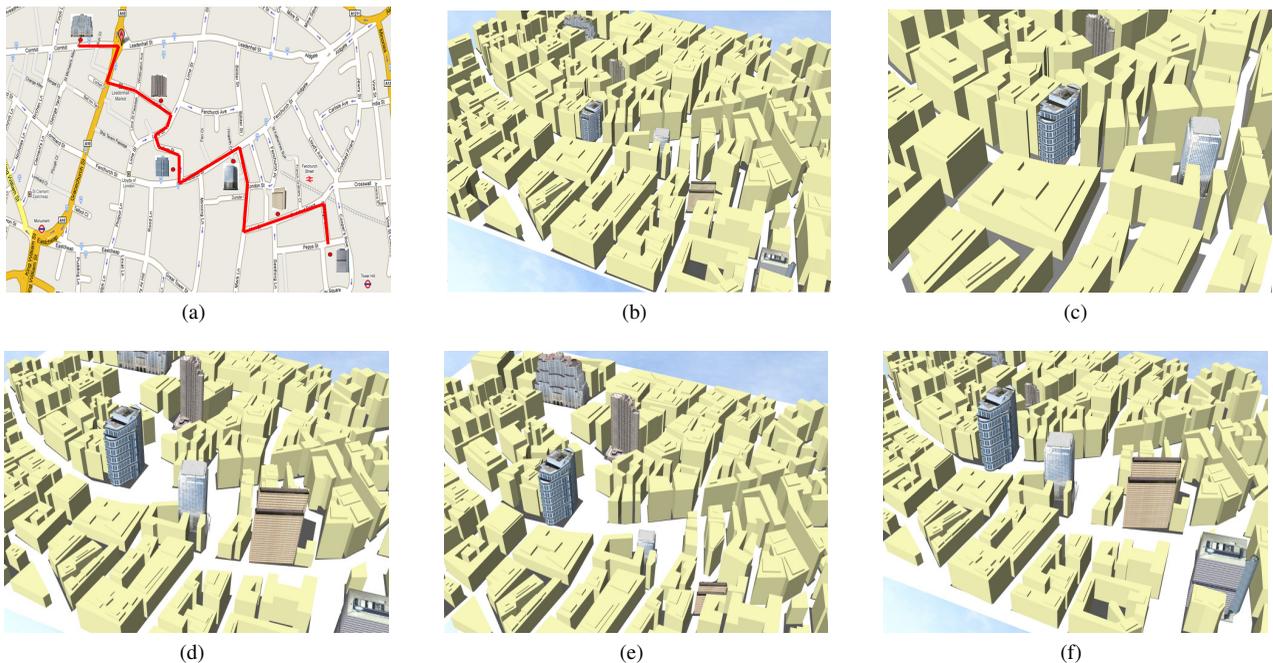


Fig. 10: Results on a 3D virtual environment modeled after London: (a) A real map of a London area and a long route; (b) The traditional 3D view; (c) The result after applying a traditional scaling; parts of the route and some landmarks are pushed out of the window; (d) The result after applying our route zooming technique; (e) The result after applying our technique only to the top half of the route; (f) The result after applying our technique to the bottom half of the route.