

3D Route Planning Within a Stereoscopic Environment Based on 2D Mouse Interaction

Leonhard Vogelmeier^(✉), Christoph Vernaleken, and Peter Sandl

Airbus Defence and Space, 85077 Manching, Germany
leonhard.vogelmeier@airbus.com

Abstract. Stereoscopic displays have reached a level of maturity sufficient for professional applications. At Airbus, an evolutionary approach is fostered to integrate such displays in existing mission planning systems for airborne use. One necessary building brick is to provide interaction with the stereoscopic content in a way users are familiar with from common (two-dimensional) human-computer interaction.

This paper describes an approach to provide three-dimensional interaction within a stereoscopic environment based on standard mouse and keyboard using interaction metaphors.

Keywords: Mission planning · Stereoscopic visualization · Virtual reality · Mixed reality · Interaction

1 Introduction

Today's mission planning systems rely on a two dimensional map for the primary mission planning task. Consequentially, the route planning process is divided into two steps, the first of which, dealing with the lateral component of the route, is carried out on a traditional map, assuming constant altitude. Afterwards, the vertical component is elaborated, refining the altitudes, aided by a profile view. This is an effective method for planning high altitude routes of low complexity. However, when topography has a major impact on mission success, e.g. for air to ground attacks, this planning method becomes inefficient, since the user has to mentally integrate both the lateral and vertical components, a very demanding task.

Planning tools based on stereoscopic, three-dimensional visualization can help to solve this issue because they relieve the user from the mental integration task. But they can only be efficient if user interaction with the 3D scene is adequately implemented.

It is intended to use the stereoscopic route planning application in conjunction with a common 2D screen, as the mission planning process requires a lot of text based and two-dimensional information to be entered and processed. To enable seamless interaction in the entire mission planning system, a basic requirement is to rely exclusively on standard mouse and keyboard, also for interaction with the stereoscopic environment. This necessitates that the handling of the mouse and the keyboard in the stereoscopic environment shall follow the expectations of users arising from their experience with the default 2D environment.

In the following, a corresponding interaction method is developed and realized inside a stereoscopic route planning demonstrator. The implementation is subsequently tested and further developed until acceptable interaction results are achieved. The improvement of the method was supported by informal user tests.

2 Problem Statement

One basic metaphor for working with a mouse and a two-dimensional mouse cursor is moving the cursor over an object to interact with it; this requires the cursor to be always the topmost object. This metaphor can easily be transferred to non-stereoscopic 3D-visualizations where the three-dimensional content is projected onto an arbitrary image plane, as visualized in Fig. 1, the content of which is then visualized on a screen. This projection reduces the *cursor over object* problem to a 2D problem that can easily be solved. By contrast, in a stereoscopic 3D-visualisation, two projections are applied, one for each eye, cf. Fig. 2. As a consequence, for all virtual 3D-objects which are not placed directly on the screen plane, two pictures are displayed on the screen, one for each eye. Trying to move the 2D cursor over a 3D symbol not in the screen plane may therefore result in placing it between the two projected pictures. This could only be solved by the user by closing one eye which would reduce the picking problem again to a 2D problem. This not an acceptable solution. First tests also showed that cursors moving only in the screen plane cause unacceptable eye strain in many cases. Moving it in the line of sight of a virtual 3D object located in front of or behind the screen plane will result in two objects on the same spot with a different depth impression for the user. Only one of the two objects can be fused to a depth image and the other one generates a hard to interpret double picture which implies unacceptable stress to the human visual perception system. This problem will be called *depth perception problem* in the following.

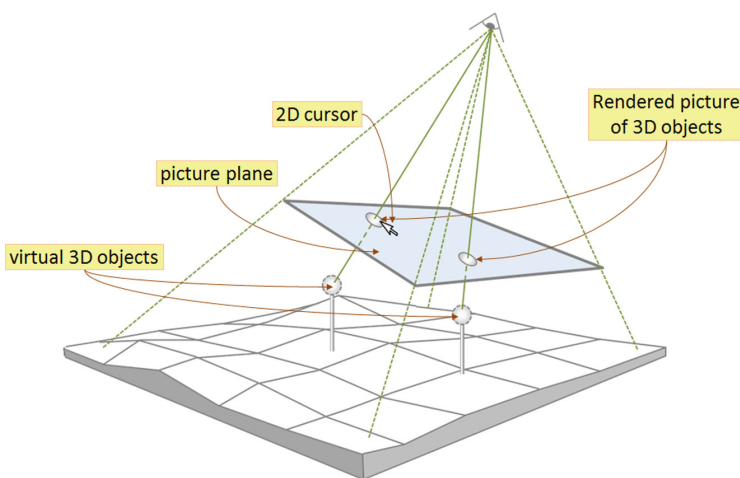


Fig. 1. Standard non-stereoscopic projection of 3D content to a 2D picture plane

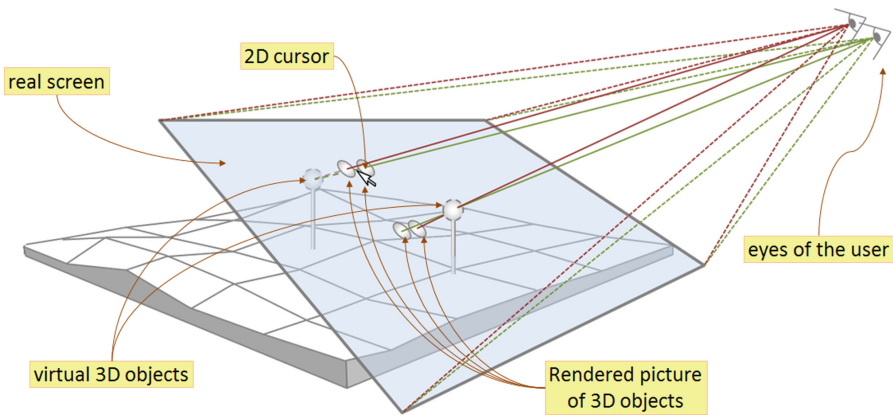


Fig. 2. Stereoscopic projection of 3D content to the screen

To provide an adequate interaction, a solution for the *cursor over object* problem must be found for use inside the stereoscopic environment. With an adequate solution, other two-dimensional interaction metaphors like *left click on object* for selection, or *mouse over* for additional information can be easily transferred.

3 Proposed Solution

As solution for this problem we introduced a virtual 3D cursor as displayed in Fig. 3. The 2D cursor is replaced by a 3D geometry which starts its movement in the screen plane at the position of the 2D cursor, the visual representation of which is hidden (see number one in Fig. 3). An intersection line is calculated, starting from a position in the middle between the user's eyes and heading through the 2D mouse position of the

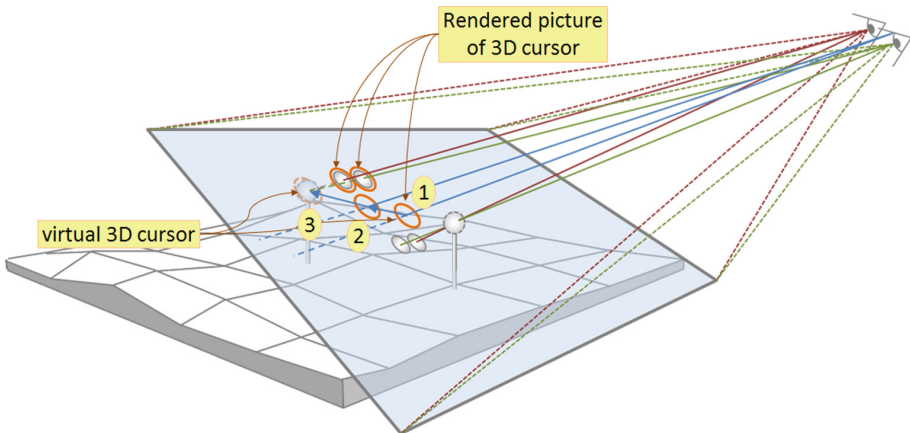


Fig. 3. Behavior of virtual 3D cursor for picking virtual 3D objects

screen. For each movement of the mouse, it is calculated if the intersection line hits virtual objects. If this is not the case, the 3D cursor moves attached to the mouse position in the screen plane (see 2 in Fig. 3). If an intersection occurs, the 3D cursor jumps to the position of the related 3D object (see 3 in Fig. 3). Tests showed that this technique works well for objects which are positioned behind the screen in relation to the user. For objects which are positioned in front of the screen the cursor has to be moved behind the object and then jumps from behind the object towards it. This violates the user expectation to move the cursor over an object for interaction, and has therefore been rated as unacceptable. Additionally, it has been detected that for objects which are located far away from the screen plane, the *depth perception problem* appears again when moving the cursor close to the objects.

To overcome these problems, the authors invented a virtual plane and transferred the cursor movement from the screen plane to the virtual plane. The virtual plane is oriented parallel to the 3D map, and the distance to the map is defined by the highest 3D object in the scene as shown in Fig. 4. The 3D cursor now always moves atop all 3D objects. In Fig. 4, it starts moving behind the screen (number one), moves out of the screen (number two) and jumps to the virtual 3D-object (number three). First tests showed that while this method was acceptable in most cases, it fails when some 3D objects are located very high above the others. This moves the virtual 3D cursor plane very high above any 3D objects placed lower, which again leads to the depth impression problem described above.

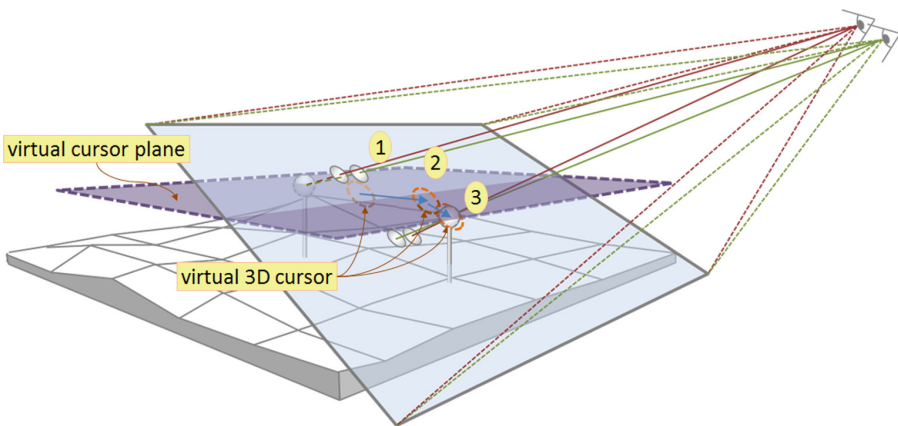


Fig. 4. Visualization Virtual cursor plane and related 3D cursor movement

To attenuate this effect, the height calculation for the cursor plane was changed to be located between the highest and the lowest object. In order to prevent a recurrence of the problem of moving the cursor behind objects for picking them, virtual intersection areas as displayed in Fig. 5 has been invented. They start at the intersection point of the plumb line of an object with the cursor plane, and reach up to object itself. They are always oriented towards the user, and their width equals the width of the object. With

every movement of the cursor, it is checked if the intersection line intersects with an intersection area. If no (see Fig. 5 number one), the cursor is placed on the cursor plane. If yes, the cursor is placed on the intersection position between intersection line and intersection area (see Fig. 5 number two). If an intersection with an object occurs, the cursor is placed on the object (see Fig. 5 number three).

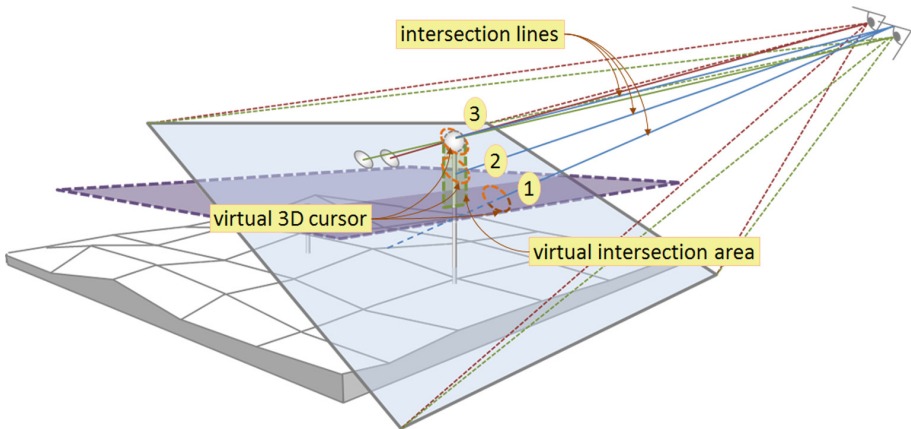


Fig. 5. Visualization of virtual intersection area for 3D cursor control

In some cases, this picking method produces the sensation of the cursor ‘crawling up’ to the 3D object, which is an unexpected behavior at the start, but it has been rated as acceptable after a short familiarization phase by the test users.

With this last adaption an acceptable *move cursor over object* method has been found and implemented in an route planning demonstrator described in the following.

4 Demonstrator Setup

A photograph of the realized demonstrator is shown in Fig. 6. It consists of a screen capable of providing an active stereo¹ picture, a tracking system, tracked active stereo shutter glasses, a standard keyboard and a standard mouse. The content, an interactive three-dimensional map and a three-dimensional route, is rendered in a way which gives the user the impression that the map is just lying on the table. The screen acts in this case like a window providing a view into a showcase.

To achieve this effect, the eye-position of the user is continuously measured by the tracking system to render the correct perspective regarding the position of the users eye, the position of the screen and the position of the table. To keep the negative effects of the de-coupling of convergence and accommodation² implied by the stereoscopic

¹ For an in detail description of *active stereo* see [1].

² For an in detail description of *de-coupling of convergence and accommodation* see [2].

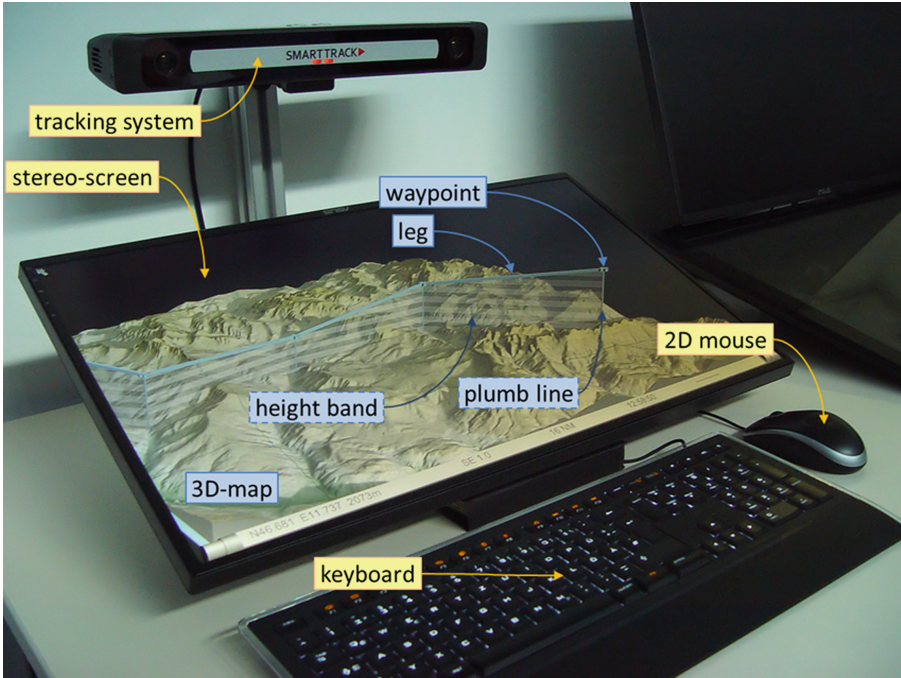


Fig. 6. Photograph of demonstrator with declaration of hardware (orange) and content (blue) (Color figure online)

visualization low, the screen has been tilted to an angle of 30° relative to the table, to keep the to be visualized content near to the screen.

The intractable objects of the demonstrator are the 3D-map and the waypoints and legs of the 3D-route (see Fig. 6). The plumb line and the height band are only for visualization. The defined needs for interaction are listed in Table 1.

Table 1. Defined needs for interaction for 3D-map and 3D-route

Interaction with 3D-map	Interaction with 3D-route
Move map	Select waypoint
Scale map	Add waypoint at end of route
Rotate map	Insert waypoint between two existing waypoints
Tilt map	Move waypoint in all three dimensions
	Delete waypoint

Using the developed *cursor over object* method, the means for interaction listed Table 2 have been implemented. The discrimination between map and object interaction is done by analyzing if a *mouse over object* occurs or not. If not map interaction is applied if yes object interaction. During map interaction e.g. moving the map the

Table 2. Defined means for interaction for route manipulation

Interaction mean	Interaction with 3D-route
Left mouse button click on waypoint	Select waypoint
Right mouse button click on 3D-map	Add waypoint at end of route
Left mouse button drag on leg	Insert waypoint between two existing waypoints
Left mouse button drag	Move waypoint parallel to ground
mouse wheel while cursor on object	Change height of waypoint
Delete-key while cursor on object	Delete waypoint

search for intersection with objects is deactivated. A special case is the interaction *add waypoint at end of route*. In this case the intersection of the actual intersection line with the 3D-map is calculated and the waypoint is placed at the intersection point with the height of the preceding waypoint.

First user tests proved the method to be usable for 3D route manipulation; further quantitative test are planned.

5 Summary and Way Ahead

The Authors propose an interaction method for a stereoscopic environment which utilizes standard mouse and keyboard and interaction metaphors derived from the well-established two-dimensional interaction.

The method has been implemented inside a stereoscopic route planning demonstrator and improved to provide suitable interaction. Qualitative, informal user tests have been applied to assist the improvement of the method. In a next step qualitative measurements shall be applied, to evaluate interaction performance and failure rate to compare it to standard two-dimensional interaction.

References

1. Wikipedia. https://en.wikipedia.org/wiki/Active_shutter_3D_system
2. Patterson, E.P.: Human Factors of Stereoscopic 3D Displays. Springer, London, Heidelberg, New York, Dordrecht (2015). doi:[10.1007/978-1-4471-6651-1](https://doi.org/10.1007/978-1-4471-6651-1)