Direct Manipulation Video Navigation in 3D

Cuong Nguyen[1] **Yuzhen Niu**[1,2] **Feng Liu**[1]
[1]Portland State University [2]Fuzhou University
Portland, OR 97207-0751 USA Fuzhou, Fujian 350108, China
{cuong3,yuzhen,fliu}@cs.pdx.edu

ABSTRACT
Direct Manipulation Video Navigation (DMVN) systems allow a user to navigate a video by dragging an object along its motion trajectory. These systems have been shown effective for space-centric video browsing. Their performance, however, is often limited by temporal ambiguities in a video with complex motion, such as recurring motion, self-intersecting motion, and pauses. The ambiguities come from reducing the 3D spatial-temporal motion (x, y, t) to the 2D spatial motion (x, y) in visualizing the motion and dragging the object. In this paper, we present a 3D DMVN system that maps the spatial-temporal motion (x, y, t) to 3D space (x, y, z) by mapping time t to depth z, visualizes the motion and video frame in 3D, and allows to navigate the video by spatial-temporally manipulating the object in 3D. We show that since our 3D DMVN system preserves all the motion information, it resolves the temporal ambiguities and supports intuitive navigation on challenging videos with complex motion.

Author Keywords
Video navigation; direct manipulation; 3D visualization.

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation: User Interfaces

INTRODUCTION
The ubiquity of video demands convenient video navigation tools. While the traditional timeline slider and its variations are good for time-centric browsing tasks [1], they are sometimes inconvenient for space-centric browsing. For example, it is often a tedious job to find a video frame where an object of interest is at a certain position using the timeline slider. To address this problem, Direct Manipulation Video Navigation systems have been recently developed [2, 3, 4, 6, 7]. These systems allow an object of interest to be directly manipulated along its motion trajectory as a way to navigate a video.

While DMVN systems can often effectively support space-centric browsing tasks, their performance is often limited by the complex object motion in a video. Previous research identified several representative motion patterns that can fail DMVN, such as recurring motion, self-intersecting motion, and pauses [2, 3, 6]. These difficulties are caused by temporal ambiguities when objects at different times are mapped to a similar or even the same position in the video frame. These ambiguities are often inevitable when the 3D spatial-temporal object motion (x, y, t) is mapped to the 2D spatial motion (x, y), which is visualized and manipulated in 2D image space by existing DMVN systems.

(a) 2D DMVN (b) 3D DMVN

Figure 1. Direct manipulation video navigation in 3D. 3D DMVN shown in (b) visualizes each video frame and the object motion trajectory in 3D, resolves the temporal ambiguity, and enables more convenient direct manipulation-based video navigation than 2D DMVN shown in (a).

In this paper, we present a 3D DMVN system that solves the temporal ambiguity problem in a principled way. Instead of projecting the 3D spatial-temporal object motion trajectory (x, y, t) into the 2D image space (x, y), which loses motion information, our system maps the 3D motion into the 3D space (x, y, z) by mapping time t to depth z. In this way, no information is lost. Our system accordingly renders each video frame in a 3D image plane and the motion trajectory in a 3D volume. Then a user can select and drag an object along the 3D motion trajectory. As our system preserves all the motion information, no temporal ambiguity is introduced: intuitively, any two points on a motion trajectory at least differ from each other in the t (e.g. z) dimension. As our system visualizes all the motion information and resolves temporal ambiguities, it enables a user to conveniently drag the object of interest and provide more accurate input for the system to navigate to the proper video frames than 2D systems.

RELATED WORK
Direct manipulation video navigation systems allow users to browse a video by directly dragging video content in the image space and have been shown effective for space-centric video browsing tasks [2, 3, 4, 5, 6, 7]. These systems extract motion information using computer vision techniques such as optical flow [12], render the motion trajectory in the image space, and allow a user to drag an object of interest along

CHI 2013, April 27–May 2, 2013, Paris, France.

its motion trajectory. When the user drags the object along the trajectory in the image space, the system will select and navigate to a proper frame that is closest to the user pointer location according to a carefully defined distance measurement. As discussed in [6], there are two common types of distance functions: purely spatial and spatial-temporal distance.

As existing DMVN methods only render the motion trajectory in the 2D image space, the 3D spatial-temporal motion trajectory is projected onto the 2D image space and thus multiple points at different frames will be possibly mapped to the same location. This introduces temporal ambiguities and sometimes fails DMVN methods. Existing methods address the ambiguities by carefully defining the distance functions [2, 3]. However, as pointed out in [6], these distance functions can only mitigate the ambiguity problems and sometimes introduce additional problems. Karrer *et al.* specifically addressed "pauses", a specific ambiguity problem, by embedding either loop or timeline in the motion trajectory visualization [6]. This paper presents a principled solution to the ambiguity problem by utilizing 3D space for motion trajectory visualization and interaction. Our solution fundamentally resolves ambiguities by preserving and visualizing all the spatial-temporal motion information.

Our 3D DMVN system is relevant to the trajectory-based video object manipulation system [10], which also visualizes video and motion in a 3D volume, but focuses on temporal video manipulation tasks, such as video re-timing and editing. Direct manipulation video navigation is not supported as video scrubbing is performed on a separate 3D representation, not directly on the video object. Our system is also related to the Video Summagator system for 3D video summarization and navigation [8]. The Summagator system models and renders a video as a 3D volume, but it only allows a user to deform the video volume and does not support direct manipulation on video object. Our system shares a similar idea to the Tumble technology that displays occluded 2D objects in a layered 2D drawing using an in-place 3D stacked view [9].

3D DIRECT MANIPULATION VIDEO NAVIGATION

Our 3D DMVN system models a motion trajectory as a spatial-temporal sequence, $\{(x_t, y_t, t)|t_a \leq t \leq t_e\}$, where (x_t, y_t) is the object location at frame t, and $[t_a, t_e]$ is the object motion duration. Our system maps the motion trajectory (x_t, y_t, t) to the 3D curve (x_t, y_t, z_t), where z_t is the third dimension of the 3D space and is a function of t. Our system visualizes the current video frame on an image plane in a 3D volume and renders the 3D trajectory in the same volume, as shown in Figure 1 (b). With our system, a user can first rotate the video frame and motion trajectory as a whole by rotating the 3D volume to avoid temporal ambiguities and then drag an object along its trajectory. The most important 3D aspect of our system is its capability to rotate the visualization in 3D to resolve the ambiguities. Meanwhile, 3D rotation will make the video frame be shown from a skewed angle. But we find that this normally does not create problems as the rotation needed to resolve the ambiguities is typically small. This is because the trajectory moves monotonically in the z direction. We now describe our system in detail.

(a) 3D DMVN without correction (b) 3D DMVN with correction

Figure 2. Perspective correction. The number of frames between every two neighboring red points is the same. Due to perspective distortion, the distance between two neighboring red points increases as the girl moves closer to the camera in (a). This problem is corrected in (b).

3D Trajectory

Like previous methods [2, 6], our method first pre-processes an input video to create the motion trajectory (x_t, y_t, t) using optical flow [12] and feature tracking [11]. The motion trajectory is then mapped to a 3D curve (x_t, y_t, z_t). By default, z_t is linearly dependent on t as $z_t = at$, where a is a constant to control the space between points on the trajectory along the z dimension. The default value for a is 2 in our system.

Perspective Correction

While the linear mapping from t to z_t works well for most of the videos we tested, a better mapping is required to handle the perspective problem in the original video. Figure 2 shows a video where a girl is walking toward the camera from the far end of the hallway and a boy is moving away from the camera. Assume that the girl is moving toward the camera at a constant speed. Due to the perspective projection of the camera imaging system, the projected walking speed of the girl in the image space increases as she approaches the camera, as shown in Figure 2 (a). This sometimes is problematic: the same amount of the dragging length results in a different amount of frame changes, thus compromising the navigation experience. For the video in Figure 2, when a user drags the girl who is faraway from the camera, the boy jumps forward quickly. When the girl is close to the camera, the boy then moves very slowly. We handle this problem by non-linearly mapping t to z_t. Specifically, we compute z_t so that the 3D distance between any two points on the trajectory remains the same. In this way, we compensate the 2D spatial displacement with that in the z dimension. A user can optionally select this perspective correction mode.

Trajectory and Video Frame Visualization in 3D

We visualize the current video frame and the motion trajectory in a 3D volume using the Visualization Toolkit (VTK)[1]. There are two common projection transformations in rendering 3D scene using a standard 3D Computer Graphics pipeline: perspective projection and orthographic projection. To avoid the distortion to the video frame and motion trajectory, we use the orthographic projection in rendering.

There are two options to position and render the current video frame and motion trajectory. The first is to render the trajectory according to its coordinate $\{(x_t, y_t, z_t)\}$ and the current video frame at a 3D plane $z = z_t$ in order to make the object follow the user pointer. We render the video frame using texture mapping in VTK. This straightforward solution has one

[1] `http://www.vtk.org`

(a) 2D DMVN (b) 3D DMVN

Figure 3. Video with recurring motion. 3D DMVN visualizes the recurring motion in 3D and clearly conveys all the motion cycles for a user to drag the object to the desired location within an appropriate cycle.

problem: as z_t changes, the image plane moves along the z axis. We find that the movement of the image plane is sometimes disturbing when a user manipulates an object. Therefore, we use a second option that we always render the current frame at $z = 0$ and when a user moves the object, we shift the trajectory toward $z = 0$ along the z axis to follow the dragging movement. Note, the trajectory is only shifted along the z dimension. No motion in the x-y image plane where the object moves in a video is changed. Our experiment shows that this method provides a natural dragging experience.

Some part of the trajectory will be occluded by the video frame, when it is shifted behind the frame. We address this problem by making the video frame semi-transparent to reveal the occluded part, as shown in Figure 5 (c) and (d). A small opacity value for the video makes the video less clear than the input but reveals the trajectory well. A big value shows the video well but reveals the trajectory less. By default, our system makes the video frame a bit semi-transparent with an opacity value of 0.85. While users can use a slider to adjust this value, our experiment shows that the default value works well and users typically do not need to change it.

Next Frame Estimation

When a user drags an object to a new location, our system finds the next frame where the object moves to the new pointer location. We use the spatial-temporal distance from [4] to find the next frame. Given a new pointer location (x_p, y_p) in the screen coordinate system, we aim to find a frame $\hat{t}$ such that $(\hat{x}, \hat{y}, \hat{t})$ minimizes the following distance.

$$\hat{t} = \arg\min_{t_v} \sqrt{(x_p - \bar{x}_v)^2 + (y_p - \bar{y}_v)^2 + \kappa(t_v - t_c)^2} \quad (1)$$

where (x_v, y_v, t_v) is a point in the motion trajectory and $(\bar{x}_v, \bar{y}_v)$ is the coordinate of (x_v, y_v) in the screen coordinates computed using the coordinate system transformation matrices from VTK. t_c is the current selected frame and κ is a scaling factor to balance the impacts of the spatial and temporal distance. We use the 2D screen coordinates instead of 3D coordinates of the pointer position because 3D coordinates of a point in a 3D volume is more tedious to specify using input devices like a mouse than its 2D coordinates.

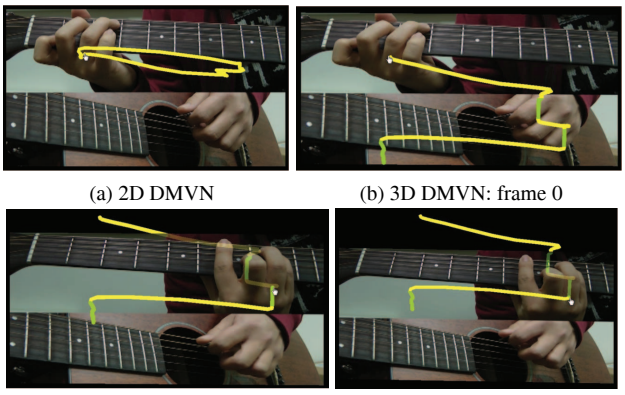
EVALUATION

We first show how our 3D DMVN system resolves temporal ambiguities that are challenging for 2D DMVN systems and then report the result from our user study. Please refer to the video demo for a better assessment[2].

[2] `http://graphics.cs.pdx.edu/project/3DDMVN`

(a) 2D DMVN (b) 3D DMVN

Figure 4. Video with self-intersecting motion. 3D DMVN resolves the trajectory ambiguity that is exhibited in 2D DMVN.



(a) 2D DMVN (b) 3D DMVN: frame 0

(c) 3D DMVN: frame 115 (d) 3D DMVN: frame 155

Figure 5. Video with "pause" motion. 3D DMVN naturally extends the motion trajectory along the t (e.g. z) dimension even though there is a pause in the motion in the x-y plane.

Figure 3 shows a video where the knob is rotated periodically. It is difficult for a user to manipulate the knob motion with 2D DMVN systems as multiple cycles of the motion will be projected onto similar locations in the 2D image space, as shown in Figure 3 (a). It is especially difficult to navigate to a specific cycle and the user experience is also compromised by sudden objectionable temporal jumps. With our 3D DMVN system, a user can solve this problem by rotating the 3D volume, as shown in Figure 3 (b). After rotation, a user can clearly observe the whole motion trajectory and drag the knob to the desired position.

Similarly, our 3D DMVN system can resolve the ambiguity from self-intersecting trajectories. Figure 4 shows a video with painting motion. The brush motion is too complicated to follow in 2D. By rotating the 3D volume, this complicated trajectory can be more easily followed in 3D than in 2D.

Our 3D DMVN system can naturally handle another special type of temporal ambiguity, pause. The video in Figure 5 shows the hand movement when playing guitar. The hand on the top pauses frequently. When this hand pauses, the hand at the bottom still moves. With 2D systems, it is difficult to navigate inside the pause period to appreciate the motion of the hand at the bottom. Karrer *et al.* handles this problem by embedding an extra loop or timeline in the motion trajectory [6]. In contrast, our system solves this pause problem without any special handling. As shown in Figure 5, the 3D trajectory naturally visualizes the "motion" in the t (e.g. z) dimension in the 3D space: while there is no motion in the x-y plane, the motion curve still extends along the t dimension. We mark the trajectory segments of pauses in green for

	Time: mean	Time: std	Accuracy: mean	Accuracy: std
2D	32.24	6.19	74.65	25.52
3D	17.87	8.08	2.83	2.58

Table 1. The user performance in locating frames of interest in videos with the 2D and 3D DMVN system. The navigation time is measured in seconds and the navigation accuracy is measured in frames.

illustration in Figure 5 (b). The user can then drag the hand along these segments to navigate inside the pause periods.

Besides these challenging motions, our 3D DMVN system can handle well with other challenging examples, such as the example shown in Figure 2 where an object moves toward camera and the actual motion has perspective effect.

User Study

We conducted a user study to evaluate the user experience with our 3D DMVN system in handling temporal ambiguities. In our study, we asked a user to locate a frame with some particular content using our implementation of a 2D DMVN system [4] and our 3D system, respectively. Both systems use the spatial-temporal distance function in [4] to locate the next frame. We tested these systems in four experiments. In each experiment, we tested on a video with each of the temporal ambiguity problems mentioned before. Specifically, the tested videos exhibit the recurring motion, self-intersecting motion, pause, and perspective problem, respectively.

Our study recruited 10 participants from our campus. Before the study, we introduced the two DMVN systems to participants and made them familiar with these systems. The participants were asked to perform the tasks on a Windows 7 desktop machine with a mouse as an input device. In order to minimize the learning effect, each participant watched each input video twice before performing each task. For each experiment, we measured the navigation time which starts when a participant grabs an object and stops when the participant stops dragging. We also recorded the frame number that the participant stops at. At the end of each experiment, we also asked the participant to select which system is easier for them to accomplish the task. We used the within-participant study design. Each participant performed all the experiments using both systems. We used a balanced 2×2 Latin square to randomize the order of systems for each participant.

Results

We report the statistical analysis on the navigation time that a participant needs to find a target frame and the navigation accuracy using the 2D and 3D DMVN system in Table 1. These results show that participants were faster when using the 3D system (M = 17.87s, STD = 8.08s) to locate the frames of interest in videos than the 2D system (M = 32.24s, STD = 6.19s). They were also more accurate with the 3D system (M = 2.83 frames, STD = 2.58 frames) than with the 2D system (M = 74.65 frames, STD= 25.52 frames). The p-values of the paired two-sample t-tests between the 3D and 2D DMVN system for the navigation time and accuracy are both smaller than 0.001. The final interviews show that all the participants unanimously consider the 3D DMVN system is easier than the 2D system to navigate to the frames of interest in the four challenging videos with significant temporal ambiguities.

CONCLUSION

This paper described a 3D DMVN system that supports the visualization and interaction of a motion trajectory in 3D. We showed that this 3D DMVN system resolves the temporal ambiguities that are typically challenging for 2D systems. As the user experience with DMVN systems depends on motion trajectory estimation, we will incorporate better motion estimation algorithms to make our system more robust in the future. Our system currently can only handle small camera motion. We will incorporate video stabilization to improve the user experience on shaky videos [2].

Acknowledgments. The videos in this paper are used from Youtube user willkempartschool, depro9, and SwingStepTv under a Creative Commons license. This work was supported by NSF CNS-1205746 and CNS-1218589.

REFERENCES

1. Cheng, K.-Y., Luo, S.-J., Chen, B.-Y., and Chu, H.-H. Smartplayer: user-centric video fast-forwarding. In *ACM CHI* (2009), 789–798.
2. Dragicevic, P., Ramos, G., Bibliowitcz, J., Nowrouzezahrai, D., Balakrishnan, R., and Singh, K. Video browsing by direct manipulation. In *ACM CHI* (2008), 237–246.
3. Goldman, D. B., Gonterman, C., Curless, B., Salesin, D., and Seitz, S. M. Video object annotation, navigation, and composition. In *ACM UIST* (2008), 3–12.
4. Karrer, T., Weiss, M., Lee, E., and Borchers, J. Dragon: a direct manipulation interface for frame-accurate in-scene video navigation. In *ACM CHI* (2008), 247–250.
5. Karrer, T., Wittenhagen, M., and Borchers, J. Pocketdragon: a direct manipulation video navigation interface for mobile devices. In *MobileHCI* (2009), 47:1–47:3.
6. Karrer, T., Wittenhagen, M., and Borchers, J. Draglocks: handling temporal ambiguities in direct manipulation video navigation. In *ACM CHI* (2012), 623–626.
7. Kimber, D., Dunnigan, T., Girgensohn, A., III, F. M. S., Turner, T., and Yang, T. Trailblazing: Video playback control by direct object manipulation. In *IEEE ICME* (2007), 1015–1018.
8. Nguyen, C., Niu, Y., and Liu, F. Video summagator: an interface for video summarization and navigation. In *ACM CHI* (2012), 647–650.
9. Ramos, G., Robertson, G., Czerwinski, M., Tan, D., Baudisch, P., Hinckley, K., and Agrawala, M. Tumble! splat! helping users access and manipulate occluded content in 2d drawings. In *the Working Conference on Advanced Visual Interfaces* (2006), 428–435.
10. Shah, R., and Narayanan, P. Trajectory based video object manipulation. In *IEEE ICME* (2011), 1–4.
11. Shi, J., and Tomasi, C. Good features to track. In *IEEE CVPR* (1994), 593–600.
12. Sun, D., Roth, S., and Black, M. Secrets of optical flow estimation and their principles. In *IEEE CVPR* (2010).