Research article

CineFolio: Cinematography-guided camera planning for immersive narrative visualization

Zhan Wang [a], Qian Zhu [b], David Yip [a], Fugee Tsung [a,c], Wei Zeng [a,c,*]

[a] *The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, 511455, China*
[b] *Renmin University of China, Beijing, 100872, China*
[c] *The Hong Kong University of Science and Technology, 999077, Hong Kong, China*

ARTICLE INFO

ABSTRACT

Narrative visualization facilitates data presentation and communicates insights, while virtual reality can further enhance immersive and engaging experiences. The combination of these two research interests shows the potential to revolutionize the way data is presented and understood. Within the realm of narrative visualization, empirical evidence has particularly highlighted the importance of camera planning. However, existing works primarily rely on user-intensive manipulation of the camera, with little effort put into automating the process. To fill the gap, this paper proposes *CineFolio*, a semi-automated camera planning method to reduce manual effort and enhance user experience in immersive narrative visualization. *CineFolio* combines cinematic theories with graphics criteria, considering both information delivery and aesthetic enjoyment to ensure a comfortable and engaging experience. Specifically, we parametrize the considerations into optimizable camera properties and solve it as a constraint satisfaction problem (CSP) to realize common camera types for narrative visualization, namely *overview camera* for absorbing the scale, *focus camera* for detailed views, *moving camera* for animated transitions, and *user-controlled camera* allowing users to provide inputs to camera planning. We demonstrate the feasibility of our approach with cases of various data and chart types. To further evaluate our approach, we conducted a within-subject user study, comparing our automated method with manual camera control, and the results confirm both effectiveness of the guided navigation and expressiveness of the cinematic design for narrative visualization.

1. Introduction

Narrative visualization, as a popular form of storytelling, integrates storytelling techniques, data visualization, and other visual elements to present data-driven insights (Segel and Heer, 2010). With the emergence of virtual reality (VR), immersive narrative visualization presents a gradually increasing potential to tell data stories in a more immersive and engaging way (Yang et al., 2023; Zhu et al., 2024). Within the immersive environment, data visualization could be presented in a three-dimensional space and take on a scalable and intuitive form to promote data understanding (Lee et al., 2021).

To guide data exploration and engage audiences with narrative visualization in virtual reality, researchers have explored design guidelines and principles for immersive data stories (Conlen et al., 2023; Yang et al., 2023). In their empirical studies, the camera is considerably employed to walk the audience through different data items and thus enhance their experiences. First, an appropriate viewing camera angle helps gain a better perspective of data items from both emotional (Shi et al., 2021) and analytic (Li et al., 2023) aspects. Second, sequencing various camera perspectives together provides both focus plus overview to the viewer (Card, 1999; Cao et al., 2020), which is beneficial for supporting different narrative goals and driving the narrative smoothly (Amini et al., 2015). Third, camera movements naturally guide the audiences' attention, help them easily follow data changes, and increase engagement with data visualization (Borkiewicz et al., 2019; Yu et al., 2024). Finally, various cinematic designs of camera movement (*i.e.*, cinematography) facilitate the expressiveness of data visualization (Xu et al., 2022; Yu and Lo, 2023). Overall, well-designed cinematic camera motion can guide viewers' attention in ways that emphasize key insights, convey temporal changes, and reflect the author's intent, thereby forming coherent narrative visualizations.

However, it remains challenging for novice people unfamiliar with data visualization and film making to create camera movements for immersive visualization. Existing toolkits for creating

* Corresponding author.
 E-mail addresses: zwang834@connect.hkust-gz.edu.cn (Z. Wang), qian.z@ruc.edu.cn (Q. Zhu), daveyip@hkust-gz.edu.cn (D. Yip), season@ust.hk (F. Tsung), weizeng@hkust-gz.edu.cn (W. Zeng).

immersive data visualization (Cordeil et al., 2019; Sicat et al., 2019) are mainly limited to static visualization rather than time-varying data stories or animations. Therefore, viewers often need to explore and understand the data on their own without guidance, which is time-consuming and labor-intensive for users who are not familiar with 3D visualization. Besides, novices may have difficulty identifying representative characteristics and become disoriented when faced with multiple data points without guidance. Furthermore, inappropriate usage of cinematography may result in a confusing rather than engaging data story, particularly if they fail to consider issues such as motion sickness in VR (Hettinger and Riccio, 1992). To fill this gap, we leverage design guidelines of cinematography techniques (Conlen et al., 2023; Yang et al., 2023) to develop a semi-automatic camera planning approach that guides users to explore and understand immersive narrative visualization.

Camera planning is a classic problem in computer graphics and both interactive and automatic techniques are widely explored in various fields including data visualization (Christie et al., 2008). Previous work (Zheng et al., 2011; Hsu et al., 2013) has focused on guiding informative views or tours for data-intensive 3D visualizations to assist visual analytics tasks. In this work, we frame immersive camera movement more than an optimization task, but also as a narrative technique that supports narrative visualization. To enhance expressiveness for camera planning in immersive narrative visualization, we first conduct a literature review across disciplines including computer graphics and cinematic storytelling. We extract common cinematic keys and distill a set of high-level considerations and requirements for camera planning design in immersive narrative visualization.

On this basis, we develop a semi-automatic camera planning approach *CineFolio* to support both novice users and experts in conveying their storytelling goals through four common cinematic cameras, *i.e.*, *overview camera* and *focus camera* for distinct views, *moving camera* for dynamic transitions, and *user-controlled camera* for interactive exploration. The method unfolds in two primary stages centered around *glyphs*, *i.e.*, *viewpoint selection* and *path planning*. In the *viewpoint selection* stage, we first decompose the immersive visualizations into a set of static visualizations of different keyframes through narrative variation. For each static visualization, we filter multiple candidate viewpoints based on the different visual importance of included *glyphs*. In the *path planning* stage, we use these candidate viewpoints to construct an initial camera path graph model. We generate an optimal camera moving path with the least transition cost and interpolate a continuous transition function to obtain the final immersive data video. Compared to the traditional speed considerations for regular human motion (Assa et al., 2008; Yeh et al., 2012), we take a further step to make the camera trace irregular dynamic narrations while avoiding motion sickness. While we leverage automated planning to reduce manual effort, our approach enables authorship by incorporating user-defined narrative focus, camera pacing, and interaction control.

In summary, the major contributions of this work include:

- A set of distilled design considerations of camera planning for immersive narrative visualization based on the knowledge from cinematic and graphic domains.
- A glyph-based approach *CineFolio* to automatically generate a camera path for time-varying immersive visualization based on user preference.
- Three example cases to demonstrate our method and a user study to investigate its effectiveness for data understanding, immersion, and engagement.

2. Related work

Narrative Visualization. Narrative visualizations, also known as data-driven storytelling, integrate data visualizations into narratives to tell stories with data (Segel and Heer, 2010). It has huge potential to facilitate data presentation and communicate insights provided by the data (Riche et al., 2018; Tong et al., 2018). Segel and Heer (Segel and Heer, 2010) identified three visual narrative tactics, *i.e.*, visual structuring (*e.g.*, progress bar), highlighting, and transition guidance (*e.g.*, animated transition and camera motion). Among various types of narrative visualization techniques, animation is a widely used technique to enhance expressiveness and facilitate users' comprehension, given its intuitive and engaging nature (Chalbi et al., 2019; Chevalier et al., 2014, 2016). In animated data stories, camera motion has been validated as an effective element of transition in terms of narration guidance (Tang et al., 2020), aesthetic enjoyment (Shi et al., 2021), and emotion delivery (Lan et al., 2022).

Previous studies have summarized salient characteristics of visual transition in narrative visualization. Hullman et al. (2013) analyzed 42 narrative visualizations and identified six between-visualization transition types. They also proposed a graph-driven approach to help build a smooth and meaningful story sequence by minimizing transition costs. Some researchers got inspiration from films and adapted cinematic-related theories into transitions. Yang et al. (2021) applied Freytag's Pyramid structure, which is widely used in films and narratives, to propose a design space about story creation. Xu et al. were inspired by films and derived cinematic opening and ending styles for designing data videos (Xu et al., 2022, 2023). To put these design guidelines and principles into practice, Li et al. (2023) developed an authoring system for camera movements design for geospatial data stories. However, they focused on the data stories presented on screens. Our work similarly applied cinematic styles into animated transition but for immersive visualization.

Immersive visualization has the inherent advantage of displaying the 3D structure and spatial information of data and therefore has great potential in spatial navigation and dynamic presentation of 3D data (Zhang et al., 2023; Zhao et al., 2022; Lu et al., 2025). However, for non-expert users, independent exploration in an immersive environment is time-consuming and easily disorienting, requiring correct spatial guidance to facilitate effective data understanding and analysis. Camera movement is a very effective technique that has been shown to drive the narrative (Conlen et al., 2023; Wang et al., 2024), avoid disturbing the viewer's sense of position (Isenberg et al., 2018), and increase immersion (Borkiewicz et al., 2019). For example, David et al. present an automatic method for producing a documentary especially for molecular visualizations through producing a story graph and generating virtual tours by automated camera and visualization transitions (Kouřil et al., 2023). Yang et al. analyzed 100 3D data videos and summarized cinematography techniques from shot framing and camera movements, calling for more practical camera planning methods applied in immersive spaces (Yang et al., 2023). Our work takes a further step to automatically generate these cinematography techniques for immersive visualization.

Automatic Camera Control. Camera control is a classic problem during computer graphics as searching the camera configuration for information capture and communication (Christie et al., 2008). Many methodologies have been proposed for different scenarios and purposes related to various fields like robotics (Kavraki et al., 1996), virtual cinematography (He L.w et al., 1996), computer animation (Assa et al., 2008; Yeh et al., 2012), 3D games (Halper et al., 2001), and data visualization (Hsu et al., 2013). Christie and Olivier (Christie et al., 2008) have conducted a comprehensive survey on interactive and automatic camera control,
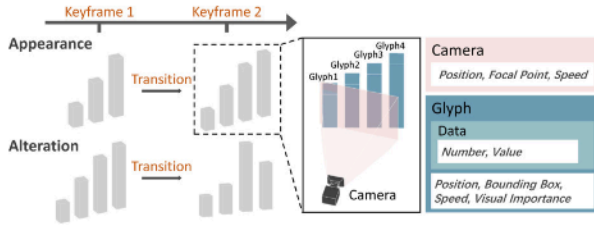
Fig. 1. This work considers properties of immersive narrative visualization, data, and camera to derive the optimal camera motion that balances data communication and expressive presentation.

covering fundamental models and approaches. Recent approaches can be divided into two main categories. The first category is related to viewpoint selection, where the system evaluates and recommends a number of views for 3D objects. Another category consists of approaches related to camera path planning. In this case, the system considers multiple constraints to achieve a globally optimal camera path for complex scenes.

For viewpoint selection techniques, researchers have proposed many measurements and criteria to serve different scenarios and purposes, which can be classified into three types: entropy-based (Vázquez et al., 2001, 2003; Feixas et al., 2009), feature-based (Ranon and Urli, 2014), and learning-based (Tao et al., 2016). Learning-based methods refer to selecting representative viewpoints based on expert opinions, mainly targeted toward subjective criteria. Entropy-based viewpoint selection techniques have been widely used in volume visualizations due to complex high-dimensional data (Zheng et al., 2011; Alharbi et al., 2023). For example, Serin et al. (2012) applied viewpoint entropy to automatically explore representative views of terrain datasets. iView (Zheng et al., 2011) reconsiders important features for volume visualizations and proposes an entropy map to display viewpoint information gained around the object onto a sphere for interactive exploration. However, entropy-based methods generally do not consider the semantic information of features and thus may not select the optimal viewpoints for features with semantic meaning. In contrast, our work focuses on storytelling using abstract data and employs feature-based methods to deliver both geometric and narrative information. To better describe narrative variations, we derive a semantic feature generated from data attributes and parametrize this feature into the visual importance of 3D objects. We integrate it into a constraint satisfaction problem along with other commonly used camera properties to select viewpoint candidates for immersive visualization.

For globally optimal pathfinding, the roadmap technique (Kavraki et al., 1996) is commonly used in robotics, computer games, and many camera motion planning techniques. Drucker and Zeltzer (Drucker and Zeltzer, 1994) applied a constraint solver to find a collision-free path for navigation through virtual environments. Andújar et al. (2004) constructed a cell and portal graph with an entropy-based measure of the relevance of a viewpoint over a 3D grid. Then the graph was used to automatically generate exploration paths for complex walkthrough models. This graph-based technique is also utilized in data visualization to present three-dimensional information. For example, Hsu et al. (2013) provided the best overall camera paths for volume data visualizations based on a collision-free roadmap. Compared to their work mainly for static visualization, our work focuses on dynamic immersive visualization. Usually, for dynamic scenes over an entire sequence of frames, the viewpoint is calculated only on a uniform subset of frames (Christie et al., 2008) and the remaining frames are interpolated continuously (Barr et al., 1992). In contrast, our work derives important keyframes during

the entire animation procedure based on the narrative variation and then utilizes possible viewpoints for each keyframe to construct the roadmap. For the smooth transition between keyframes, Assa et al. (2008) and Yeh et al. (2012) both suggested camera speed should be as constant as possible or at least with minimal acceleration. While their studies focused on capturing regular human behaviors, we take further consideration for camera speed of tracing irregular data actions (Borkiewicz et al., 2019) to create an immersive narrative experience. However, when referring to immersive visualization, simply setting a constant camera speed from data changing is unsuitable due to a common issue for immersive environments, *e.g.*, motion sickness (Wang et al., 2019, 2021). Our method provides an intuitive interpolation function for camera speed, making the balance between motion sickness and narrative variation.

3. Background and design considerations

To facilitate discussions, this section first introduces key concepts and terminologies related to camera planning for immersive narrative visualization (Section 3.1). Next, we provide an overview of different cinematic camera types (Section 3.2) and outline the necessary design requirements (Section 3.3) to be met in order to enhance the user experience.
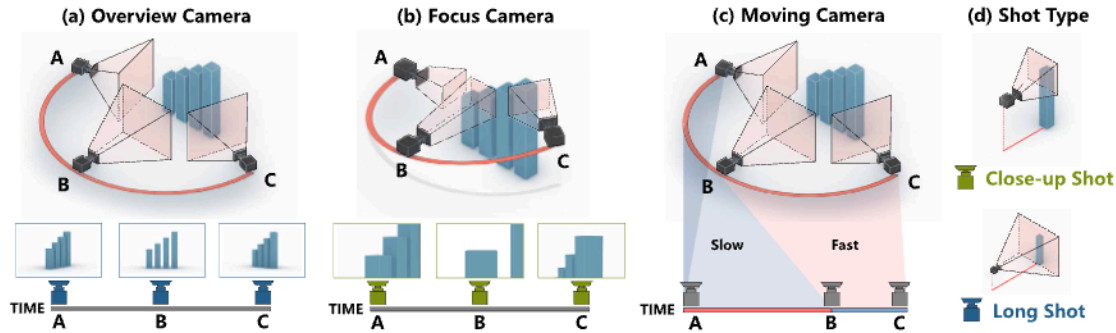
3.1. Concept and terminology

This work aims to optimize camera movements for narrative immersive visualization, specifically for 3D abstract visualization (such as 3D bar charts and heatmaps) in VR environments. The input for this optimization process is a sequence of data points that change over time. In order to synchronize these changes in data with camera motion, we classify the variations in data points into two distinct groups (Fig. 1).

- *Appearance.* Data appearance refers to the addition of a new data point or the removal of an existing one at each time stamp. This can be observed in the example of a narrative visualization for GDP growth, as shown in Case 1 (see top row of Fig. 5).
- *Alteration.* Data alteration refers to instances where the number of input data points remains constant, but their values change over time. An example can be seen in the narrative visualization of traffic volume fluctuations over one day, as depicted in Case 2 (see the middle row of Fig. 5). In the visual appearance, the visual attribute of the corresponding glyph changes as well.

At each time stamp, the input data is represented by a 3D abstract visualization that consists of a set of glyphs for each data point. Animations are used to depict data variation across different time stamps. Two distinct types of animations are employed, namely the appearance of visual forms (*e.g.*, a bar that grows in height) for *data appearance*, and the variation of visual forms (*e.g.*, a 3D heatmap that rises and falls) for *data alteration*. In accordance with Thompson et al.'s framework for animated data graphics (Thompson et al., 2020), we define the following terms:

- *Scene.* A *scene* is the immersive environment including both the visualization and background context.
- *Glyph.* A *glyph* is a graphic symbol used to represent a data point. The visual characteristics of a glyph are used to encode the value of a data point. This study focuses on immersive visualization, where glyphs are represented by 3D geometric properties, such as position within the immersive environment, and shape that is encapsulated by a geometric bounding box.
- *Keyframe.* A *keyframe* is defined as a visualization at rest for a specific time stamp.

VISINF: 100259

ARTICLE IN PRESS

Z. Wang, Q. Zhu, D. Yip et al.   Visual Informatics xxx (xxxx) xxx

Fig. 2. **CineFolio** uses (a) the close-up shot sequence for **overview camera** and (b) the long shot sequence for **focus camera**, and (c) adjusts the camera speed for data variations in **moving camera**.

- *Transition.* A *transition* is the uninterrupted change that occurs between two consecutive *keyframes*.

Immersive visualization involves many data glyphs and it is difficult to use current keyframe-based methods to identify and keep track of them. To prioritize the most important data glyphs as the narrative focus, we introduce two new quantifiable properties for a *glyph*.

- *Glyph speed* refers to the degree to which a glyph changes over time when visualized.
- *Visual importance* refers to the relative significance of a glyph in the visualization. In general, glyphs that undergo variations are considered more important than static glyphs.

The visual appearance of glyphs at each keyframe and the animation effects during transitions are achieved through the movement of the camera. In particular, we focus on three fundamental properties of the camera: *position*, *focal point*, and *speed*. We omit other properties, such as field-of-view (FOV), that are fixed from the outset.

3.2. Cinematic camera types

This work aims to automate camera movements to convey a narrative of 3D abstract data in an immersive environment. Our inspiration and guidelines come from cinematic theories that dictate how shots should be composed and organized in films. We adopt a basic distance-based classification of camera shots for *keyframe*: *close-up shot*, *medium shot*, and *long shot*. We attempt to combine these shot types into a continuous video to develop four cinematography techniques for narrative visualization summarized by Conlen et al. (2023), illustrated in Fig. 2.

- *Overview camera.* An overview camera can assist the viewer in absorbing the scale of the data rather than focusing on details. This type is commonly achieved by combining multiple long shots, where the camera is positioned far away from the visual objects (Cutting et al., 2011).
- *Focus camera.* A focus camera is preferable when presenting detailed information about specific visual objects of interest. In contrast to the *overview camera*, a focus camera is typically achieved using multiple close-up shots (Katz, 1991), which brings the object of interest closer to the focal point of view.
- *Moving camera.* In certain cases involving a moving object, especially when its speed is too fast for viewers to keep up with, a moving camera can track the object's motion and create a sense of transportation (Conlen et al., 2023). In this work, this camera is used together with *overview camera* or *focus camera* to control the camera speed between different shots.

- *User-controlled camera.* Certain cinematic visualizations offer users some control over the camera, such as predefining the next keyframe. In this work, we empower users to determine the content they wish to explore and generate a semi-automatic camera that reflects their preferences.

We develop algorithms to facilitate camera movement control for narrative visualization. Our method enables automatic *overview*, *focus*, and *moving cameras*, while also incorporating user inputs for *user-controlled camera*.

3.3. Design requirements

In addition to incorporating cinematic theories to convey a narrative, we also strive to optimize camera movement in order to effectively communicate information to viewers. To achieve this, we reviewed the literature and identified a set of design requirements that specifically focus on camera control for data visualization. These criteria are condensed into two levels: *keyframe* and *transition*. At the *keyframe* level, we strive to achieve the following goals:

K.1 Occlusion minimization. Numerous efforts have been devoted to mitigating occlusion problems using camera control in 3D visualization (Christie et al., 2008). This work also seeks to reduce occlusion for *glyphs* in the scene, regardless of the camera type being used. An essential factor to consider is to maximize the information of *glyphs* to be communicated with the viewer.

K.2 Insight maximization. Narrative visualization involves considerations more than just geometry for minimizing occlusion; it also aims to maximize the communication of data insights to the viewer (Vázquez et al., 2001). To achieve this, data objects encoding the most significant data characteristics are expected to be positioned near the focal point and made visible to the viewer.

K.3 Visual balance. Since most visualizations are asymmetric, we focus on the two balance types from the taxonomy in the visual arts (Martinez and Block, 1995) to achieve visual balance. 1) *Radial balance*: A radially balanced view places greater emphasis on a central point than on other areas. This draws the audience's attention to a focal point, effectively achieving the desired effect for the *focus camera*. 2) *Crystallographic balance*: This type aims to evenly distribute visual weight across all data objects in the scene. The goal is to draw the viewer's attention to the entire view, similar to how an *overview camera* captures a broad view.

At the *transition* level, we address two related concerns to engage viewers in stories and prompt them to react to data insights (Borkiewicz et al., 2019):

T.1 Transition smoothness. Two factors are considered here. First, sudden changes in the viewing direction should be avoided as they may trigger motion sickness, especially in VR environments (Wang et al., 2019). Second, camera speed is expected to change smoothly when transitioning between consecutive
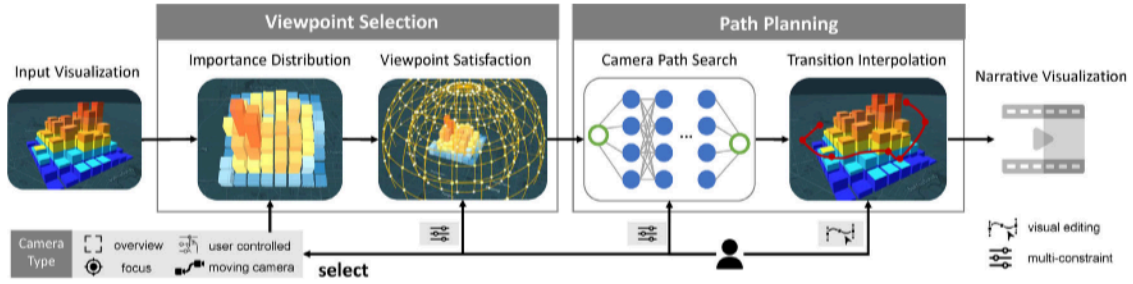
Fig. 3. Overview of our approach to camera movement planning for immersive narrative visualization, which can be divided into two main stages: (1) the **viewpoint selection** stage filters candidate viewpoints at each keyframe, and (2) the **path planning** stage searches for the optimal camera movement path and interpolates smooth transitions between keyframes. At each individual step, user control is enabled to allow for customization based on their preferences and knowledge.

keyframes. Abruptly jumping from one keyframe to another may cause the viewer to miss important insights or anticipate actions prematurely. Additionally, camera transitions are expected to align with the intended storyline to support narrative coherence. Viewpoint changes between keyframes should not only preserve spatial orientation but also emphasize key narrative beats. There-fore, we consider the relationship between the camera speed and *data variation*. When a data point changes significantly, the camera speed of the *moving camera* should be increased smoothly to keep pace with the data change.

T.2 Overall continuity. To create a more cohesive storytelling experience, the camera movement should be consistently ori-ented throughout the entire story. The keyframe-based camera planning method only computes the locally optimal viewpoint for an individual keyframe. If viewpoints for two or more consecutive keyframes have huge differences in camera position and orienta-tion, this can cause disorientation and disruption to the viewer's sense of space and scale, making them feel lost in the 3D zone.

4. CineFolio

Our semi-automated camera approach *CineFolio* enables effi-cient planning of camera movements to support various cine-matography camera types, including *overview*, *focus*, and *moving cameras*, while allowing users to manipulate camera movements in *user-controlled camera* mode. To achieve this, we employ a two-stage procedure as illustrated in Fig. 3: we first perform *viewpoint selection* (Section 4.1) to filter a set of candidate viewpoints in each keyframe and then generate an optimal camera moving path through *path planning* (Section 4.2). This results in a video of narrative visualization that meets the desired requirements. To facilitate the discussion, we list down commonly used notations adopted in this work as follows:

- *Keyframe.* A keyframe is denoted as k, with K indicating the total number of keyframes in a narrative visualization. The timestamp of the keyframe is denoted by $t(k)$. It is selected based on prominent data variations.
- *Transition* refers to the animation that takes place between two consecutive keyframes k and $k+1$. Throughout this work, the transition time. represented as $t(k+1)-t(k)$, remains constant for all transitions within a narrative visualization.
- *Data* refers to a sequence of data points denoted as $\{D_k | k \in \{1, \ldots, K\}\}$. At each keyframe, D_k includes a list of data points represented by $D_k := \{d_k^i\}_{i=1}^{|D_k|}$. The number of data points in D_k differs based on whether there is an *appearance* of new data points, in which case $|D_{k+1}|$ is greater than $|D_k|$, or if there is a *data alteration*, in which case $|D_{k+1}| = |D_k|$.
- *Immersive visualization* refers to a collection of abstract visual-izations presented in an immersive environment and denoted as $\{G_k | k \in \{1, \ldots, K\}\}$. Each visualization G_k consists of a
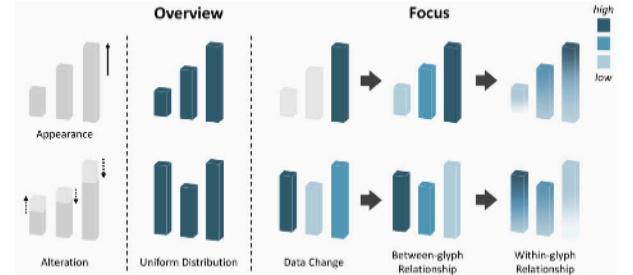
Fig. 4. Different visual importance settings for **overview camera** and **focus camera** camera. In the **overview camera**, every region of the glyph is equally important. In a **focus camera**, we provide three assumptions for computation based on **data change**, **between-glyph**, and **within-glyph relationships**.

set of glyphs represented by $\{g_k^i\}_{i=1}^{|G_k|}$, where g_k^i corresponds to the data point d_k^i. Each glyph g includes multiple attributes, such as bounding box $bbox(g)$, glyph speed $speed(g)$, and visual importance $vi(g)$.

4.1. Viewpoint selection

The objective of this stage is to generate a list of viewpoints that meet the requirements **K.1–K.3** in the *keyframe* level. While previous studies on viewpoint optimization have focused on the geometric properties of visual objects, this work goes a step further by taking into account the data attributes defined by the visual importance distribution.

4.1.1. Visual importance distribution

In order to assess the level of visually compelling regions in visualization, we introduce the concept of visual importance. The computation of visual importance, with a range from 0 to 1, varies depending on the type of camera being used, as depicted in Fig. 4. Specifically, for the *overview camera*, which displays the data scale in the scene, we assume that all glyphs are equally important and assign a uniform visual importance distribution to all glyphs. In the *user-controlled camera* mode, users can manually assign the importance distribution based on their preferences. The *fo-cus camera* mode presents a much more complex scenario with glyphs that have significantly varied levels of interest. Here, we consider the visual importance distribution from three different aspects:

- *Data change*: Glyphs that experience larger changes in the encoded data are deemed more interesting.
- *Between-glyph relationship*: If a glyph is deemed important, its adjacent glyphs will also be affected and considered important to some extent. The degree of influence is determined by their spatial relationship to the focal glyph.

Algorithm 1 Compute visual importance

Require: A list of glyphs $\{g_k^i\}$ and data changes $\{\triangle d_k^i\}$
Ensure: Visual importance distribution $\{vi(g_k^i)\}$

1: **for each** $g_k^i \in \{g_k^i\}$ **do**
2:     Initialize $vi(g_k^i) = 0$;
3:     $vi(g_k^i) = vi(g_k^i) +$ MappingDataChange($\triangle d_k^i$);
4: Normalize $\{vi(g_k^i)\} = \{vi(g_k^i)\}/\sum_{g_k^j \in \{g_k^j\}} vi(g_k^j)$;
5: **for each** $g_k^i \in \{g_k^i\}$ **do**
6:     **for each** $g_k^j \in \{g_k^j\}$ && $i \neq j$ **do**
7:         $vi(g_k^i) = vi(g_k^i) + Decay(\text{Dist}(g_k^i, g_k^j))$;
8: Normalize $\{vi(g_k^i)\} = \{vi(g_k^i)\}/\sum_{g_k^j \in \{g_k^j\}} vi(g_k^j)$;
9: **return** $\{vi(g_k^i)\}$

- *Within-glyph relationship*: Different regions on a glyph are not equally important. For example, if a bar is increasing in height, we consider the top region of the bar to be the most important, with other regions becoming less important as the distance from the top increases. The importance of an area on this glyph is distributed based on the importance of this single glyph and the distance from the varying area.

To compute the visual importance of each glyph g_k^i at keyframe k, we first measure the corresponding data change from the previous keyframe as $\triangle d_k^i = |d_k^i - d_{k-1}^i|$. Next, we pass the glyphs $\{g_k^i\}$ and their data changes $\{\triangle d_k^i\}$ into Algorithm 1, which calculates the visual importance for each glyph based on *data change* and *between-glyph relationship*. Here, MappingDataChange($\triangle d_k^i$) refers to a continuously increasing nonlinear function, empirically tuned based on trial and error, to scale the magnitude of data change into a value between 0 and 1. To account for the *within-glyph relationship*, we further divide the bounding box of a glyph into stacked boxes and assign different levels of importance to each individual box, using the same decay function when modeling *between-glyph relationship*. The resulting spatio-temporal importance distribution is then utilized to determine the camera's focal point in each keyframe. This information, along with user-selected cinematography types and predefined constraints, is then passed to the next stage for calculating viewpoint satisfaction.

4.1.2. Viewpoint satisfaction

Viewpoint selection is formulated as a constraint satisfaction problem (CSP), which has been widely adopted in camera control (Christie et al., 2008). For a 3D virtual scene, searching for the globally optimal viewpoint involves exploring the entire possible space of camera positions relative to a given focal point. However, this process can be computationally expensive, due to the huge search space. In the context of this work, immersive visualizations involve only a small number of polygons, so we can simplify the search by sampling points evenly from bounding spheres with varying radii centered around the focal point. This approach helps to avoid an overly exhaustive search.

We define a viewpoint satisfaction function as a linear combination of the constrained satisfaction of parameters for a particular viewpoint vp, which are derived from the properties of a glyph g and a viewpoint vp (Table 1). Among these parameters, *projection area* and *framing size* are primarily affected by the distance between the viewpoint and the fixed focal point determined by the weighted visual importance distribution. These two metrics, along with *horizontal* and *vertical orientation*, refine the viewing angle. The *occlusion* parameter affects both the viewpoint

Table 1
Parameters $\{p_j\}$ used in CSP for viewpoint satisfaction, derived from a glyph g and a viewpoint vp.

PARAMETER	SYMBOL	SEMANTICS
Projection Area	$area(g, vp)$	Area of g projected on vp
Framing Size	$frame(g, vp)$	Fraction of g projected on vp
Horizontal Orientation	$ori_h(g, vp)$	Angle between the front vector of g and the vector from the focal point to vp
Vertical Orientation	$ori_v(g, vp)$	Angle between the up vector of g and the vector from the focal point to vp
Occlusion	$occ(g, vp)$	Fraction of g occluded by other glyphs on vp

distance and the viewing angle. For each parameter, we defined its individual satisfaction function using a B-spline with an arbitrary number of control points. After trial and error, the chosen control points were decided based on the opinions of two experts, one in data visualization and the other in cinema, regarding the video's last performance. To determine a possible viewpoint vp at keyframe k, we compute the overall satisfaction based on each glyph's satisfaction for each parameter $p_j \in \{p_j\}$, weighted by the glyph's visual importance $vi(g_k^i)$ and the normalized parameter weight w_{p_j}:

$$Sat(vp) = \sum_{g_k^i \in \{g_k^i\}} vi(g_k^i) \left(\sum_{p_j \in \{p_j\}} w_{p_j} Sat(p_j, p_j(g_k^i, vp)) \right),$$

We then select the top N candidate viewpoints for each keyframe by ranking all sampled viewpoints based on their satisfaction scores. Formally, the selected set of candidate viewpoints V_k^* for keyframe k is computed as:

$$V_k^* = \operatorname*{argmax}_{V_k'} \sum_{vp \in V_k'} Sat(vp), \quad \text{subject to } |V_k'| = N$$

where V_k' is a subset of the candidate viewpoints at keyframe k, and N is the number of best viewpoints that satisfy keyframe-level constraints. To improve computational efficiency, we employ geometric methods to evaluate properties based on bounding boxes and ray casting, as described in Ranon and Urli (2014). The final optimal path is computed by solving the globally constrained camera path optimization described in Section 4.2.1, which integrates this viewpoint satisfaction with transition cost minimization.

4.2. Camera path planning

In the *path planning* stage, the main objective is to meet the requirements **T.1–T.2** by utilizing the viewpoint candidates generated in the first stage. Taking a global perspective (**T.2**), our method aims to search for the optimal camera path with the least transition cost. Specifically, our approach considers the speed of data variation to produce a camera path that smoothly tracks the changes in the data (**T.1**).

4.2.1. Camera path search

In order to select the optimal viewpoints for each keyframe, we consider both local satisfaction and global continuity. To this end, we construct a camera path graph that connects the top N viewpoint candidates. We use this graph to determine the shortest path that minimizes the transition cost of both camera movement distance and orientation shifts. To begin the camera path search, we first define two imaginary viewpoints as the source and destination. These viewpoints are not included in the final path. Then we construct a camera path graph from all
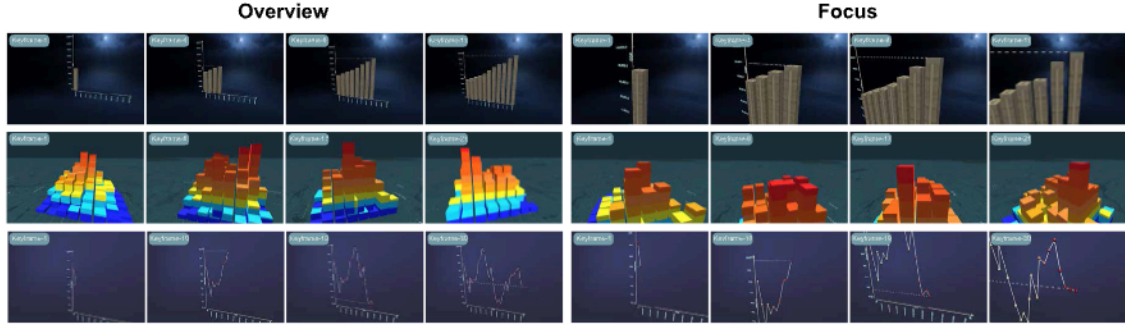
Overview   Focus

Fig. 5. Video snapshots from example cases of **GDP growth** (top), **traffic volume** (middle), and **S&P500 stock price** (bottom) generated with **CineFolio**. Each case contains four snapshots for **overview camera** and **focus camera** respectively.

viewpoint candidates of all keyframes. In the camera graph, the node represents a possible viewpoint with its satisfaction, and the edge connects two viewpoint candidates from two consecutive keyframes weighted by the transition cost. To reduce the computational complexity, we first prune those nodes with too low satisfaction or having too large transition costs with the last keyframe. To satisfy the overall continuity (*Overall continuity*) requirement, we define the transition cost between two consecutive keyframes as the difference between two fields of view. We denote the camera position of viewpoint vp as pos_{vp} and the orientation as the vector from the focal point to the camera position, $ori_{vp} = pos_{vp} - focal_{vp}$. The cost evaluation function to compute the weight for the edge between two viewpoints vp_i, vp_j can be described as the combination of distance between two viewpoints' positions and angle between two viewpoints' orientations:

$$C(vp_i, vp_j) = c_1 Angle(ori_{vp_i}, ori_{vp_j}) + c_2 Dist(pos_{vp_i}, pos_{vp_j}),$$

where c_1 and c_2 are the coefficients of the different terms. We then use Dijkstra's algorithm to compute the optimal viewpoints for all keyframes with the shortest transition cost with minimal orientation and position shifts, from the source to the destination in the graph. The full camera path is constructed by linearly interpolating the positions and orientations between these selected viewpoints.

4.2.2. Smooth transition interpolation

In the case of *moving* camera, the camera speed should be as constant as possible or at least with minimal acceleration to minimize dizziness and confusing views (Assa et al., 2008; Yeh et al., 2012). However, for the animated graphics, the visual appearance of *glyphs* might change at a more irregular speed according to the data variation. If the camera speed is still limited in a low range, it may happen that the changing *glyph* goes out of view. Taking this difference between camera motion and data variation into consideration, we define a camera speed function to catch up with data variation while maintaining continuous speed changes.

We have obtained the optimal viewpoint $\dot{v}p_k$ for each keyframe k in the *viewpoint selection* stage. In the normal case, the camera speed just maintains a constant during two keyframes k and $k + 1$ as $Dist(pos_{\dot{v}p_k}, pos_{\dot{v}p_{k+1}})/\Delta t$, where $\Delta t = t(k + 1) - t(k)$ denotes the timestep between two keyframes. For another case of the too-fast speed of the changing glyph, the speed function can be modeled as a polynomial function of at least second order $g_s(t)$ to catch up with data variation during two keyframes. Requirements are considered from two aspects. First, both the speed and acceleration should change continuously, and thus acceleration at all keyframes is zero, $g_s'(t(k)) = 0$. Second, the moving distance between every two keyframes is given since the

optimal position for each keyframe is determined in the *viewpoint selection* stage. This means the integral results of the polynomial function between every two keyframes should be equal to the normal speed function, $\int_{t(k)}^{t(k+1)} g_s(t) = Dist(pos_{\dot{v}p_k}, pos_{\dot{v}p_{k+1}})$. This polynomial function has multiple possible solutions to meet these requirements. Hence, we require that the optimal speed function also minimizes a deviation with the normal constant speed at each value of time. When the camera speed increases above the limited value during two keyframes, it should continuously decrease during the next normal transition rather than directly jump to a constant. The deviation is measured between three keyframes as:

$$deviation(g_s(t)) = \int_{t_i}^{t_{i+2}} (g_s(t) - f_s(t))^2.$$

4.3. User control

We provide users with the flexibility to manipulate the camera at each stage of the process (as shown in Fig. 3). *CineFolio* is designed to support both novice users who lack expertise in cinematography or data visualization, and expert users seeking fine-grained control. Novice users can select from predefined cinematic camera types, while more experienced authors can manually adjust importance scores for glyphs in the Unity interface to control the narrative focus. First, users can choose from three camera types as input: *overview camera*, *focus camera*, and *user-controlled camera*. For the first two camera types, *CineFolio* automatically computes the visual importance distribution. For the *user-controlled camera*, users can set the visual importance for each glyph based on their own preferences through the Unity editing panel. Users can also choose which constraints to use during the automatic design process. Lastly, users can manually adjust the camera positions for each keyframe on the optimal camera path generated by our method.

5. Evaluation

We first created three example cases of immersive narrative visualizations using *CineFolio* implemented in Unity3D (Section 5.1) to demonstrate the validity in various types of visualization. We also conducted a within-subject study to evaluate the usefulness and user engagement with the immersive narrative visualization by *CineFolio* (Section 5.2).

5.1. Example cases

We created three types of immersive visualizations (Fig. 5), which are most commonly used in existing 3D data videos (Yang et al., 2023).

Case 1: GDP Growth. This story illustrates China's economic growth between 2012 and 2022 through an animated bar chart.
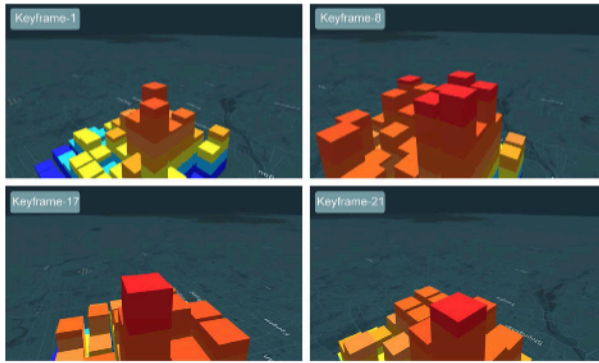
Fig. 6. Snapshots for traffic volume in *user-controlled camera*. The user expected to focus on the region with extreme values.

The intended narrative communicates a steady and positive progression in national GDP over time. Each bar corresponds to the GDP of a particular year, and the chart demonstrates a consistently increasing trend in GDP on a global scale. We used *CineFolio* to create a dynamic visualization that showcases the bars growing steadily with a constant speed one by one and the texture of bills to facilitate the visceral understanding of data (Lee et al., 2021). The *overview camera* tells the story by showing the entire timeline of bars steadily rising in height. This gives viewers a broad impression of continuous growth. In contrast, *focus camera* emphasizes key narrative beats by zooming in on the most recent bars, underscoring the acceleration of GDP near the end of the timeline.

Case 2: Traffic Volume. This story presents daily fluctuations in Beijing's urban traffic across different regions and hours through a 3D heatmap. The narrative focuses on the emergence of hotspots and transitions in congestion throughout the day. The geospatial data is aggregated and encoded into 64 bars that represent the traffic volume of each smaller region within the selected area. The traffic volume of each region was encoded by colors and the height of bars for intuitive data perception from different angles. Since the data alteration appears regularly according to data, we extract timestamps of these variations as keyframes. The *overview camera* camera narrates this spatial variation by orbiting across the cityscape, mimicking a cinematic drone shot to expose the overall patterns. The *focus camera* highlights individual regions with sudden spikes in volume, supporting a story about anomalies or critical time windows in congestion. For example, in keyframes 8 and 17, the camera is attracted by those very high bars, which means the largest traffic volumes. However, in keyframe 21, the focal point is obviously changed to the left region next to those very high bars since there appear huge data variations with the last keyframe.

Case 3: S&P500 Stock Price. This story conveys the volatile fluctuations of the S&P 500 during the second half of 2022 through a line chart. The narrative is about local highs and lows within a globally uncertain market. Similar to Case 1, *overview camera* provides the full trendline for context, showing the rise and fall across time. In contrast, *focus camera* traces the growing line with irregular up and down changes and emphasizes the local trend more. Compared to *overview camera*, focal points of view are much closer to the highest and lowest points in keyframes 10 and 19. Especially in keyframe 19, this close shot provides a more accurate comparison between two values with subtle differences.

For these cases, we also enlisted two users to test *user-controlled camera* for the same visualizations. The *user-controlled camera* allows the author or viewer to define their own story, for instance by highlighting a region of interest or anomaly. In the traffic case (Fig. 6), a user specified a focus on extreme values, resulting in a personalized narrative emphasizing that anomaly throughout the camera movement. Initially, the user assigned a value of 1 to the specific glyph and 0 to the others, denoting his preference. At keyframe 1 of the resulting video, the view encompasses all the glyphs, and the camera-to-glyph distance is between configurations in the views captured by two automated cameras. Between keyframes 1 and 8, the camera moves toward the crucial glyph. Finally, as depicted in keyframes 17 and 21, the camera focuses solely on the designated glyph while the automated cameras attempt to capture a broader range of glyphs (middle row of Fig. 5).

5.2. User study

5.2.1. Participants

We recruited 12 participants (M = 5, F = 7; Age: 20–27) through the university mailing list and collected their prior experience in VR, visualization, and computer games. They all had normal or corrected-to-normal visions. Their VR experience varied: 3 participants had no experience; 3 participants had less than 10 h experience; 3 participants had 10-20 h of experience, and 3 participants had more than 20 h of experience. Most participants have experience using and creating data visualization and their self-reported experience ranged from 1 to 3 years. Most participants (8/12) played computer games frequently: only 4 reported they played less than 2 h of games per week, and the other 2 participants played 20 h per week. We provided a $10 gift card as compensation for each participant.

5.2.2. Design procedure

The experiment was within-subject with three conditions that were divided according to the degree of automatic guidance based on our method, including (1) a fully automated overview mode, (2) a fully automated focus mode, and (3) a user-controlled exploration mode without any navigation guidance. For per participant, we obtained 3 camera modes × (2 Bar Tasks + 3 Heatmap Tasks + 2 Line Tasks) × 2 answer ranges (1 for accuracy and 1 for confidence) = 42 responses. We used a Latin square design to partition the participants into 3 groups to balance the order of the three tested camera modes.

Participants were first given a brief introduction to the experimental procedure. VR headset (Oculus Quest 2) was adjusted to ensure the sample text could be shown in front of them and seen clearly. We then provided a short VR training about how to control the camera in the VR headset with controllers. We allowed them to try a self-controlled camera when the user-controlled mode was presented in the experiment for the first time. For each condition (camera mode × visualization), participants first experienced the immersive narrative visualization without known tasks. Then we asked them to finish the tasks one by one. We combined several low-level tasks referring to Brehmer and Munzner's taxonomy (Brehmer and Munzner, 2013; Munzner, 2014) and generated three or four detailed questions for each case to encourage users to interpret data from different perspectives. Participants were allowed to explore the visualization unlimited times. After completing all tasks, participants were asked to fill in a 7-point Likert scale questionnaire on each mode's overall engagement of the visualization by collectively considering four aspects (i.e., focused attention, cognitive involvement, aesthetics, enjoyment) (Amini et al., 2018), usability, immersion, and workload of each mode (i.e., the mental load and physical load). We also conduct a short interview to collect their feedback about each camera mode, their expected automated camera design, and other expected guided tools in immersive visualizations. The experiment lasted 1 h on average.

VISINF: 100259

ARTICLE IN PRESS

Z. Wang, Q. Zhu, D. Yip et al.   Visual Informatics xxx (xxxx) xxx
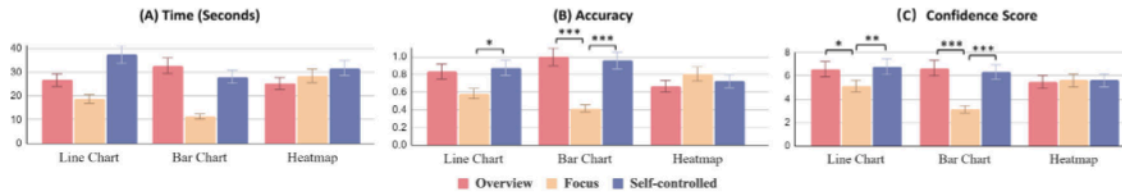
Fig. 7. Results for the (A) average time, (B) accuracy, and (C) confidence score by case. Confidence intervals indicate 95% confidence for mean values. An asterisk indicates statistical significance for p <.05.
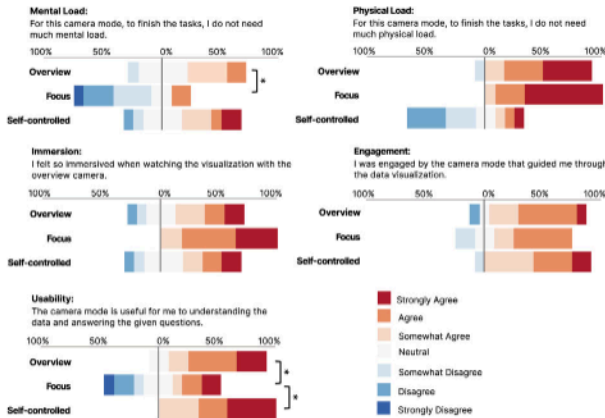
Fig. 8. Results for the participants' subjective ratings. An asterisk indicates statistical significance for p <.05.

5.2.3. Results

Quantitative Result. As shown in Fig. 7(A), we observed a trend that the average time with our methods (i.e., *overview camera* and *focus camera* modes) is less than the self-controlled mode without significance ($p > .05$). For the average task accuracy in Fig. 7(B), we found that participants got significantly higher scores with the self-controlled mode than the *focus camera* mode in the line ($p < .05$) and bar chart cases ($p < .001$). Moreover, participants performed better under the *overview camera* mode than the *focus camera* mode with the bar chart case ($p < .001$). There is no significance between these modes in the heatmap case. Additionally, participants reported significantly higher confidence scores with the self-controlled mode and the *overview camera* mode than the *focus camera* in the bar chart and the line chart cases (Fig. 7(C)).

In Fig. 8, the subjective ratings of all cases show that participants perceived the *overview camera* and the self-controlled mode have higher usability scores for finishing the tasks than the *focus camera* mode ($p > .05$). They reported the *focus camera* has a higher mental load than the *overview camera* ($p < .05$). However, their responses indicate that the *overview camera* and *focus camera* modes have lower physical load than the self-controlled mode ($p < .001$). The result of the overall engagement shows that participants were more engaged with the self-controlled mode than the *focus camera* mode ($p < .05$). The results of other aspects exhibit no significant variation in the three modes.

Qualitative Feedback. We present participants' reasons for their subjective ratings (Fig. 8) and how they engage with the immersive visualization under different experimental modes.

Immersion. Generally, participants reported the novel experience with visualization in VR. Although they did not show a significant difference in different modes (Fig. 7(D)), more than half of participants (7/12) mentioned that the *focus camera* provided a more *"immersive and shocked experience with data (P8)"* because they could observe the visualization from a relative close

view with *"smooth guidance that follows the animation of data (P11)"*.

Usability. Participants rated lower scores for the usability of the *focus camera* (Fig. 7(D)) than the other modes as the *focus camera* mode is not helpful in understanding data. They preferred to use *overview camera* and the self-controlled mode because both of them can *"provide a brief and steady perception of the entire data (P3)"*. This echoes the task accuracy result in Fig. 7(B). However, some participants reported that *focus camera* is beneficial for perceiving the data changes, such as *"comparing similar data with slight differences (P9)"*, *"tracing important changes (P10)"*, and *"judging the extreme values (P7)"*. Most of the participants (9/12) said that they would prefer to use the automated camera rather than self-controlled mode to understand the data when they may *"not know where to watch without the guidance at the beginning (P4)"* or *"have difficulty in manually catching up with the dynamic data (P5)"*

Workload. Echoing the feedback on the usability, participants reported a higher mental load with *focus camera* than *overview camera* as they could not always get the whole picture of data. The other reason is that sometimes they *"may not trust the* focus camera *that could always deliver important information (P10)"*. For the physical load, they rated significantly higher scores with the self-controlled mode than the automated cameras as they *"do not need to move my body, even my head (P1)"*, which can be easy to view the data animation. In addition, they expressed the difficulty in adjusting the camera when viewing the dynamic visualization as they *"could be concentrated on the task with the automated view (P5)"* and *"always forget to use the controllers to follow the data changes (P1)"*.

Engagement with Visualization. Participants were more engaged with the self-controlled control mode than the *focus camera* as they prefer *"the sense of free control (P1)"* in VR, and most of them (8/12) may connect the engagement with the freedom based on their feedback. We found that participants show different altitudes toward *focus camera*. Some participants (5/12) reported shocked viewing experience with *"obvious visual impact"* and explicitly expressed the preference for this mode because it brings *"a deepen the impression of data variations (P7, P9)"* and *"a drastic vibration that makes me feel I am hitting the visualization (P12)"*. In contrast, others (P4, P10) said *focus camera* emphasizes much on the current view of data which makes them easier to *"lose the memory of previous information"*. Participants gave higher cognitive involvement scores of *overview camera* and self-controlled mode than *focus camera* as they could *"obtain the overview perspective, especially for the bar chart and line chart (P4, P7)"*. This allows them to finish the tasks more easily than using *focus camera* based on their cognition.

6. Discussion

We discuss the design implications and research opportunities for future camera motion planning in immersive narrative visualizations based on our findings.

Understand the best practice of cinematography between immersive narrative and data exploration. We used cinematography as the narrative mechanism in immersive visualization, as we believe, with the convenience of teleportation (Isenberg et al., 2018) and aesthetic potential (Conlen et al., 2023; Borkiewicz et al., 2019), it can be a promising technique to enhance the narrative experience as well as facilitate the data exploration. As a probe of extending cinematic effects to virtual reality (Yip, 2020), we initially considered the most commonly used camera shots and parameterized the automatic generation as a constraint-based optimization problem. Compared to traditional 2D story-telling, virtual reality enables embodied experiences that spatially and temporally situate the viewer inside data stories. This allows for unique narrative effects, such as surrounding a user in a data hotspot, or guiding attention through physical proximity, which would be impossible on flat screens. However, constraints are predefined based on the general consensus, and thus final performance is not sensitive to user preference. Future work could leverage large language models to design a more flexible and personalized approach for more narrative visualizations.

Enhance viewers' trust of automatic camera design through data-driven approaches. Our study revealed that a few participants were confused about the reasons for automatic camera motion. As a result, they kept a close eye on the visualization in order not to lose any information. This conflict highlights a need for making automatic logic more transparent, potentially through visual overlays or annotations that explain why the camera is moving where it does, helping novice users build trust in the system's storytelling capabilities. Besides, the design considerations of automatic camera planning should consider users' prior experience with VR and visualization. For example, professionals may find their expected exploring logic inconsistent with the normal one generated by automatic cameras. Future research could also focus on learning camera techniques from existing 3D data video examples through data-driven approaches such as neural networks and applying the learned rules to create new narrative immersive visualizations.

Enrich interactive layers for guided immersive narratives. Our study shows that many people prefer to explore complex visualization freely with first guidance. The initial goal of our method is to generate the optimal camera motion for viewing the animated immersive visualization. Therefore, users were limited in the provided perspectives when viewing the visualization. A few participants emphasized the sense of control in the immersive environment and expected the combination of flexible interaction and automatic guidance. For instance, when audiences view an immersive visualization following an automatic camera, they should be allowed to control the camera manually through body actions at any time. Thus, guided camera design responsive to real-time human actions remains promising. Future work should investigate hybrid approaches in which the system suggests an initial camera path based on the storyline, while allowing users to dynamically adjust or modify segments of the path.

7. Conclusion

In summary, this paper presents *CineFolio*, a semi-automatic camera planning approach that guides data exploration and enhances the engaging narrative experience for immersive visualizations. Based on a set of design considerations that summarize commonly used cinematic camera types and graphics criteria, *CineFolio* facilitates automatic creation for three common camera types, *i.e.*, *overview camera*, *focus camera*, and *user-controlled camera*, by allowing users to input their limited preferences and edit the generated camera path. *CineFolio* has been evaluated with three cases of various abstract data visualization types and user studies, showing its potential for aiding viewers to effortlessly and engagingly explore immersive visualizations.

CRediT authorship contribution statement

Zhan Wang: Writing – original draft, Software, Methodology. **Qian Zhu:** Writing – original draft, Visualization, Formal analysis. **David Yip:** Writing – review & editing, Validation. **Fugee Tsung:** Supervision, Funding acquisition. **Wei Zeng:** Writing – review & editing, Validation, Supervision, Funding acquisition, Conceptualization.

Ethical approval

This study was reviewed and approved by The Human and Artefacts Research Ethics Committee at HKUST(GZ). The protocol number is HKUST(GZ)-HSP-2024-0005. The study was conducted in accordance with general ethical research standards for human participants, ensuring participant privacy, confidentiality, and voluntary participation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

A.1. The list of experimental tasks

Case 1: Bar Chart
Q1: From which year does GDP exceed 1 million?
Q2: Which year has the largest growth rate?
Case 2: Heatmap
Q1: Which area has largest overall traffic volume?
Q2: Which time has the largest volume?
Q3: Which area has the largest variation in traffic volume?
Case 3: Line Chart
Q1: Which week has the highest stock price?
Q2: Which week has the lowest stock price?

Appendix B. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.visinf.2025.100259.

References

Alharbi, R., Strnad, O., Luidolt, L.R., Waldner, M., Kouřil, D., Bohak, C., Klein, T., Gröller, E., Viola, I., 2023. Nanotilus: Generator of immersive guided-tours in crowded 3D environments. IEEE Trans. Vis. Comput. Graphics 29, 1860–1875. http://dx.doi.org/10.1109/TVCG.2021.3133592.

Amini, F., Riche, N.Henry., Lee, B., Hurter, C., Irani, P., 2015. Understanding data videos: Looking at narrative visualization through the cinematography lens. In: Proc. ACM CHI. pp. 1459–1468. http://dx.doi.org/10.1145/2702123.2702431.

Amini, F., Riche, N.H., Lee, B., Leboe-McGowan, J., Irani, P., 2018. Hooked on data videos: Assessing the effect of animation and pictographs on viewer engagement. In: Proc. ACM AVI. pp. 1–9. http://dx.doi.org/10.1145/3206505.3206552.

Andújar, C., Vázquez, P., Fairén, M., 2004. Way-finder: Guided tours through complex walkthrough models. Comput. Graph. Forum 499–508. http://dx.doi.org/10.1111/j.1467-8659.2004.00781.x.

Assa, J., Cohen-Or, D., Yeh, I.C., Lee, T.Y., 2008. Motion overview of human actions. In: ACM SIGGRAPH Asia. pp. 1–10. http://dx.doi.org/10.1145/1457515.1409068.

Barr, A.H., Currin, B., Gabriel, S., Hughes, J.F., 1992. Smooth interpolation of orientations with angular velocity constraints using quaternions. ACM SIGGRAPH Comput. Graph. 26, 313–320. http://dx.doi.org/10.1145/142920.134086.

Borkiewicz, K., Christensen, A., Kostis, H.N., Shirah, G., Wyatt, R., 2019. Cinematic scientific visualization: The art of communicating science. In: ACM SIGGRAPH 2019 Courses. pp. 1–273. http://dx.doi.org/10.1145/3305366.3328056.

Brehmer, M., Munzner, T., 2013. A multi-level typology of abstract visualization tasks. IEEE Trans. Vis. Comput. Graphics 19, 2376–2385. http://dx.doi.org/10.1109/TVCG.2013.124.

Cao, R., Dey, S., Cunningham, A., Walsh, J., Smith, R.T., Zucco, J.E., Thomas, B.H., 2020. Examining the use of narrative constructs in data videos. Vis. Informatics 4, 8–22. http://dx.doi.org/10.1016/j.visinf.2019.12.002.

Card, M., 1999. Readings in Information Visualization: Using Vision To Think. Morgan Kaufmann.

Chalbi, A., Ritchie, J., Park, D., Choi, J., Roussel, N., Elmqvist, N., Chevalier, F., 2019. Common fate for animated transitions in visualization. IEEE Trans. Vis. Comput. Graphics 26, 386–396. http://dx.doi.org/10.1109/TVCG.2019.2934288.

Chevalier, F., Dragicevic, P., Franconeri, S., 2014. The not-so-staggering effect of staggered animated transitions on visual tracking. IEEE Trans. Vis. Comput. Graphics 20, 2241–2250. http://dx.doi.org/10.1109/TVCG.2014.2346424.

Chevalier, F., Riche, N.H., Plaisant, C., Chalbi, A., Hurter, C., 2016. Animations 25 years later: New roles and opportunities. In: Proc. ACM AVI. pp. 280–287. http://dx.doi.org/10.1145/2909132.2909255.

Christie, M., Olivier, P., Normand, J.M., 2008. Camera control in computer graphics. In: Comput. Graph. Forum. pp. 2197–2218. http://dx.doi.org/10.1111/j.1467-8659.2008.01181.x.

Conlen, M., Heer, J., Mushkin, H., Davidoff, S., 2023. Cinematic techniques in narrative visualization. arXiv preprint arXiv:2301.03109.

Cordeil, M., Cunningham, A., Bach, B., Hurter, C., Thomas, B.H., Marriott, K., Dwyer, T., 2019. IATK: An immersive analytics toolkit. In: Proc. IEEE VR. pp. 200–209. http://dx.doi.org/10.1109/VR.2019.8797978.

Cutting, J.E., Brunick, K.L., DeLong, J.E., 2011. How act structure sculpts shot lengths and shot transitions in hollywood film. Projections 5, 1–16. http://dx.doi.org/10.3167/proj.2011.050102.

Drucker, S.M., Zeltzer, D., 1994. Intelligent camera control in a virtual environment. In: Graphics Interface, Canadian Information Processing Society. pp. 1–10.

Feixas, M., Sbert, M., González, F., 2009. A unified information-theoretic framework for viewpoint selection and mesh saliency. ACM Trans. Appl. Percept. 6, 1–23. http://dx.doi.org/10.1145/1462055.1462056.

Halper, N., Helbing, R., Strothotte, T., 2001. A camera engine for computer games: Managing the trade-off between constraint satisfaction and frame coherence. In: Comput. Graph. Forum. pp. 174–183. http://dx.doi.org/10.1111/1467-8659.00510.

He, L.w., Cohen, M.F., Salesin, D.H., 1996. The virtual cinematographer: A paradigm for automatic real-time camera control and directing. In: Proc. ACM SIGGRAPH. pp. 217–224. http://dx.doi.org/10.1145/237170.237259.

Hettinger, L.J., Riccio, G.E., 1992. Visually induced motion sickness in virtual environments. Presence: Teleoperators & Virtual Environ. 1, 306–310.

Hsu, W.H., Zhang, Y., Ma, K.L., 2013. A multi-criteria approach to camera motion design for volume data animation. IEEE Trans. Vis. Comput. Graphics 19, 2279–2801. http://dx.doi.org/10.1109/TVCG.2013.123.

Hullman, J., Drucker, S., Riche, N.Henry., Lee, B., Fisher, D., Adar, E., 2013. A deeper understanding of sequence in narrative visualization. IEEE Trans. Vis. Comput. Graphics 19, 2406–2415. http://dx.doi.org/10.1109/TVCG.2013.119.

Isenberg, P., Lee, B., Qu, H., Cordeil, M., 2018. Immersive Visual Data Stories. Springer, Cham, http://dx.doi.org/10.1007/978-3-030-01388-2_6.

Katz, S.D., 1991. Film Directing Shot By Shot: Visualizing from Concept To Screen. Gulf Professional Publishing.

Kavraki, L.E., Svestka, P., Latombe, J.C., Overmars, M.H., 1996. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. IEEE Trans. Robot. Autom. 12, 566–580. http://dx.doi.org/10.1109/70.508439.

Kouřil, D., Strnad, O., Mindek, P., Halladjian, S., Isenberg, T., Gröller, M.E., Viola, I., 2023. Molecumentary: Adaptable narrated documentaries using molecular visualization. IEEE Trans. Vis. Comput. Graphics 29, 1733–1747. http://dx.doi.org/10.1109/TVCG.2021.3130670.

Lan, X., Shi, Y., Wu, Y., Jiao, X., Cao, N., 2022. Kineticharts: Augmenting affective expressiveness of charts in data stories with animation design. IEEE Trans. Vis. Comput. Graphics 933–943. http://dx.doi.org/10.1109/TVCG.2021.3114775.

Lee, B., Brown, D., Lee, B., Hurter, C., Drucker, S., Dwyer, T., 2021. Data visceralization: Enabling deeper understanding of data using virtual reality. IEEE Trans. Vis. Comput. Graphics 27, 1095–1105. http://dx.doi.org/10.1109/TVCG.2020.3030435.

Li, W., Wang, Z., Wang, Y., Weng, D., Xie, L., Chen, S., Zhang, H., Qu, H., 2023. GeoCamera: Telling stories in geographic visualizations with camera movements. In: Proc. ACM CHI. http://dx.doi.org/10.1145/3544548.3581470.

Lu, Tao, Zhu, Qian, Ma, Tiffany, Kam-Kwai, Wong, Xie, Anlan, Endert, Alex, Yang, Yalong, 2025. Ego vs. exo and active vs. passive: Investigating the individual and combined effects of viewpoint and navigation on spatial immersion and understanding in immersive storytelling. In: Proc. ACM CHI. pp. 1–19. http://dx.doi.org/10.1145/3706598.3713849.

Martinez, B., Block, J., 1995. Visual Forces: An Introduction To Design. Prentice Hall.

Munzner, T., 2014. Visualization Analysis and Design. CRC Press.

Ranon, R., Urli, T., 2014. Improving the efficiency of viewpoint composition. IEEE Trans. Vis. Comput. Graphics 20, 795–807. http://dx.doi.org/10.1109/TVCG.2013.2297932.

Riche, N.H., Hurter, C., Diakopoulos, N., Carpendale, S., 2018. Data-Driven Storytelling. CRC Press.

Segel, E., Heer, J., 2010. Narrative visualization: Telling stories with data. IEEE Trans. Vis. Comput. Graphics 16, 1139–1148. http://dx.doi.org/10.1109/TVCG.2010.179.

Serin, E., Adali, S.Hasan., Balcisoy, S., 2012. VEA 2012: Automatic path generation for terrain navigation. Comput. Graph. 36, 1013–1024. http://dx.doi.org/10.1016/j.cag.2012.08.006.

Shi, Y., Lan, X., Li, J., Li, Z., Cao, N., 2021. Communicating with motion: A design space for animated visual narratives in data videos. In: Proc. ACM CHI. pp. 1–13. http://dx.doi.org/10.1145/3411764.3445337.

Sicat, R., Li, J., Choi, J., Cordeil, M., Jeong, W.K., Bach, B., Pfister, H., 2019. DXR: A toolkit for building immersive data visualizations. IEEE Trans. Vis. Comput. Graphics 25, 715–725. http://dx.doi.org/10.1109/TVCG.2018.2865152.

Tang, T., Tang, J., Hong, J., Yu, L., Ren, P., Wu, Y., 2020. Design guidelines for augmenting short-form videos using animated data visualizations. J. Vis. 23, 707–720. http://dx.doi.org/10.1007/s12650-020-00644-z.

Tao, Y., Wang, Q., Chen, W., Wu, Y., Lin, H., 2016. Similarity voting based viewpoint selection for volumes. Comput. Graph. Forum 35, 391–400. http://dx.doi.org/10.1111/cgf.12915.

Thompson, J., Liu, Z., Li, W., Stasko, J., 2020. Understanding the design space and authoring paradigms for animated data graphics. In: Comput. Graph. Forum. pp. 207–218. http://dx.doi.org/10.1111/cgf.13974.

Tong, C., Roberts, R., Borgo, R., Walton, S., Laramee, R.S., Wegba, K., Lu, A., Wang, Y., Qu, H., Luo, Q., et al., 2018. Storytelling and visualization: An extended survey. Information 9, 65. http://dx.doi.org/10.3390/info9030065.

Vázquez, P.P., Feixas, M., Sbert, M., Heidrich, W., 2001. Viewpoint selection using viewpoint entropy. In: Proc. ACM VMV. pp. 273–280. http://dx.doi.org/10.5555/647260.718491.

Vázquez, P.P., Feixas, M., Sbert, M., Heidrich, W., 2003. Automatic view selection using viewpoint entropy and its application to image-based modelling. In: Comput. Graph. Forum. pp. 689–700. http://dx.doi.org/10.1111/j.1467-8659.2003.00717.x.

Wang, Y., Chardonnet, J.R., Merienne, F., 2019. VR sickness prediction for navigation in immersive virtual environments using a deep long short term memory model. In: Proc. IEEE VR. pp. 1874–1881. http://dx.doi.org/10.1109/VR.2019.8798213.

Wang, Y., Chardonnet, J.R., Merienne, F., Ovtcharova, J., 2021. Using fuzzy logic to involve individual differences for predicting cybersickness during VR navigation. In: Proc. IEEE VR. pp. 373–381. http://dx.doi.org/10.1109/VR50410.2021.00060.

Wang, Z., Yuan, L.P., Wang, L., Jiang, B., Zeng, W., 2024. VirtuWander: Enhancing multi-modal interaction for virtual tour guidance through large language models. In: Proc. ACM CHI. pp. 1–20. http://dx.doi.org/10.1145/3613904.3642235.

Xu, X., Wu, A., Yang, L., Wei, Z., Huang, R., Yip, D., Qu, H., 2023. Is it the end? Guidelines for cinematic endings in data videos. In: Proc. ACM CHI. pp. 1–16. http://dx.doi.org/10.1145/3544548.3580701.

Xu, X., Yang, L., Yip, D., Fan, M., Wei, Z., Qu, H., 2022. From 'wow' to 'why': Guidelines for creating the opening of a data video with cinematic styles. In: Proc. ACM CHI. pp. 1–20. http://dx.doi.org/10.1145/3491102.3501896.

Yang, L., Wu, A., Tong, W., Xu, X., Wei, Z., Qu, H., 2023. Understanding 3D data videos: From screens to virtual reality. In: Proc. IEEE PacificVis. pp. 197–206. http://dx.doi.org/10.1109/PacificVis56936.2023.00029.

Yang, L., Xu, X., Lan, X., Liu, Z., Guo, S., Shi, Y., Qu, H., Cao, N., 2021. A design space for applying the freytag's pyramid structure to data stories. IEEE Trans. Vis. Comput. Graphics 28, 922–932. http://dx.doi.org/10.1109/TVCG.2021.3114774.

Yeh, I.C., Lin, W.C., Lee, T.Y., Han, H.J., Lee, J., Kim, M., 2012. Social-event-driven camera control for multicharacter animations. IEEE Trans. Vis. Comput. Graphics 18, 1496–1510. http://dx.doi.org/10.1109/TVCG.2011.273.

Yip, D.K.M., 2020. Cinematic Surrealism of the Interactive Virtual Space. Springer, Singapore, pp. 53–71. http://dx.doi.org/10.1007/978-981-15-4642-6_4.

Yu, Z., Lo, C.H., 2023. Mapping the viewer experience in cinematic virtual reality: A systematic review. Presence: Virtual Augment. Real. 32, 205–229. http://dx.doi.org/10.1162/pres_a_00409.

Yu, Z., Wang, H., Un, K., 2024. Automatic cinematography for body movement involved virtual communication. IET Commun. 18, 344–352. http://dx.doi.org/10.1049/cmu2.12748.

Zhang, Y., Wang, Z., Zhang, J., Shan, G., Tian, D., 2023. A survey of immersive visualization: Focus on perception and interaction. Vis. Informatics 7, 22–35. http://dx.doi.org/10.1016/j.visinf.2023.10.003.

Zhao, Y., Jiang, J., Chen, Y., Liu, R., Yang, Y., Xue, X., Chen, S., 2022. Metaverse: Perspectives from graphics, interactions and visualization. Vis. Informatics 6, 56–67. http://dx.doi.org/10.1016/j.visinf.2022.03.002.

Zheng, Z., Ahmed, N., Mueller, K., 2011. Iview: A feature clustering framework for suggesting informative views in volume visualization. IEEE Trans. Vis. Comput. Graphics 17, 1959–1968. http://dx.doi.org/10.1109/TVCG.2011.218.

Zhu, Q., Yuan, L., Xu, Z., Yang, L., Xia, M., Wang, Z., Liang, H.N., Ma, X., 2024. From reader to experiencer: Design and evaluation of a VR data story for promoting the situation awareness of public health threats. Int. J. Hum.-Comput. Stud. 181, 1–13. http://dx.doi.org/10.1016/j.ijhcs.2023.103137.